UAB
Universitat Autònoma de Barcelona

From Pixels to Patterns: Learning the Visual Grammar of Document Layouts

Sanket Biswas

UAB

Universitat Autònoma de Barcelona

From Pixels to Patterns: Learning the Visual Grammar of Document Layouts

A dissertation submitted by **Sanket Biswas** at Universitat Autònoma de Barcelona to fulfil the degree of **Doctor of Philosophy**.

Bellaterra, August 19, 2025

Director	**Dr. Josep Lladós** Universitat Autònoma de Barcelona Dept. Ciències de la Computació Centre de Visió per Computador
Co-Director	**Dr. Umapada Pal** Indian Statistical Institute Computer Vision and Pattern Recognition Unit
Thesis committee	**Dr. Joost van de Weijer** Universitat Autònoma de Barcelona Dept. Ciències de la Computació Centre de Visió per Computador **Dr. Silvia Cascianelli** Dipartimento di Ingegneria "Enzo Ferrari" Università degli Studi di Modena e Reggio Emilia, Italy **Dr. Anjan Dutta** Surrey Institute for People-Centred Artificial Intelligence (PAI) University of Surrey, United Kingdom (UK)

This document was typeset by the author using LaTeX 2ε.

The research described in this book was carried out at the Computer Vision Center, Universitat Autònoma de Barcelona.

ISBN 978-84-121011-8-8

Printed by Ediciones Gráficas Rey, S.L.

To my Mom, Missi, my friends and the loving memory of my father and grandmother...

Agradecimientos

At the age of 22, I arrived in Barcelona with curiosity in my heart and uncertainty in my steps. The **Computer Vision Center (CVC)** soon became more than just a workplace — it became a home. A place where ideas found form, friendships took root, and the language of research became a shared rhythm. I will always be grateful to the CVC and its vibrant community for shaping both my academic path and my personal growth. Thank you for giving me belonging, purpose, and a place to dream. I would like to thank the FI grant from the UAB for providing me a four years doctoral study scholarship without that for sure it would have not been possible.

This thesis is written in loving memory of my late father, **Subhasis Biswas**. Though life did not allow him to stand beside me at this milestone, I have felt his presence in every quiet moment of perseverance and every small victory along the way. His love, resilience, and unwavering belief in me have been my compass. In truth, every page of this work carries a part of him—this achievement is ours, not mine alone.

Among those I owe the most heartfelt thanks is my advisor **Prof. Josep Lladós**, who has been a steady presence throughout this journey. From the very beginning, he gave me the rare gift of freedom — the freedom to explore, to question, to contribute independently, and to collaborate across domains and ideas. He trusted me to find my own voice in the research world, even when my paths were unconventional or messy. He has been my rock — not only intellectually, but also emotionally — through moments of clarity and confusion, through success and self-doubt. With patience, empathy, and calm, he navigated my chaotic self and allowed me to fully immerse in this beautiful journey. His quiet confidence in me gave me the courage to take risks, be creative, and keep moving forward. For his mentorship, generosity, and unshakable belief in my potential, I am profoundly grateful. This thesis would not have been possible without his guidance, and it is an honor to have grown under his wing. I would also like to thank my co-supervisor **Prof. Umapada Pal**, from whom I got the inspirations for pursuing Ph.D. Thank you for your constant encouragement, for introducing me to invaluable opportunities, and for opening the door to the CVC and to Prof. Lladós, which ultimately shaped the course of my doctoral journey.

I would like to express my sincere gratitude to **Adobe Research** for hosting me during my international research stay, where I had the opportunity to develop *DocSynthv2*. I am especially thankful to my mentor, **Dr. Rajiv Jain**, for his guidance, insightful feedback, and constant support throughout the project, and to **Dr. Tong Sun**, Director of the team, for entrusting me with this opportunity and fostering an environment that

encouraged creativity and exploration. This collaboration not only strengthened the technical contributions of my thesis but also expanded my perspective on the practical impact of layout-aware document generation. Also, a big thanks to my colleagues and companions at Adobe (Curtis, Vlad, Chris, Varun, Joe, Puneet, Ani, Jiuxiang) for helping me grow during this incredible journey. I am also grateful to **Ennova Research** for the opportunity to see my work applied in a real-world setting through the deployment of *SwinDocSegmenter* in their agentic AI application, *K-Lens*. My sincere thanks go to the company's CEO, **Raffaele Andreace**, for inviting me to collaborate with the entire Ennova team and for fostering an environment where research could be translated into impactful, production-ready solutions. This experience not only broadened my understanding of industrial deployment but also helped me grow as a practitioner, learning how to bridge the gap between academic innovation and operational AI systems.

To the colleagues who became my companions, my extended family — thank you for your support through late nights, paper deadlines, and shared laughter. To **Sounak** — thank you for being like a brother to me when I had just arrived. Your guidance, warmth, and quiet strength helped me find my feet in a new country and in a research world that often felt overwhelming. You've been my unofficial mentor, and I will always cherish the way you looked out for me with both kindness and clarity. To **Pau Riba** — thank you for being there at the very start, both as a formal mentor and as a friend. You helped bridge the transition into the CVC community, and your early support gave me the grounding I needed to begin this journey. To **Andrés** — my best friend at CVC, thank you for being the one I could share everything with — from research frustrations to personal joys. Your presence made even the hardest days easier, and the good ones unforgettable. To **Ali** — thank you for being my big brother, my fiercest critic, and the most loving person all at once. You challenged me to grow, to reflect, and to never settle. Your honesty and affection have shaped me deeply, and I carry that with immense gratitude. To **Rubén** — your calming presence throughout the years was a gift. In moments when I became overly excited, impatient, or overwhelmed, you reminded me to breathe, to observe, to think. Thank you for being that quiet anchor in the storm. And to my CVC companions — David, Hector, Ramzy, Klara, Papa, Stan, Diego, Alex, Manu, Armin, Federico, Francesco, Marco, Enrico, Pietro, and Giuseppe — thank you for sharing the workspace, the conversations, the lunches, and the laughter. The energy we built together gave this place life and made CVC not just an institution, but a home. To **Ayan** and **Moha** — thank you for being my longest-standing collaborators during this thesis and for resisting me through every intense brainstorm, chaotic deadline, and wild idea; and to **Andrea Gemelli** (the Bro), thank you for always having my back no matter what. **Emanuele**, you were always like a little brother to me at CVC — full of curiosity, kindness, and light. To **Sergi** — my partner-in-crime and my longest flatmate over three incredible years — sharing this journey with you was an absolute pleasure, and your presence made both work and life feel lighter. And to **Beatriz** — your smile could always brighten the toughest days, and you brought a breath of fresh air whenever I needed it most. A special thank you to **Khanh**, my Vietnamese "machine" friend, whose relentless work ethic and dedication never ceased to inspire me. Sharing the lab with you — watching you work day and night with such focus — reminded me what

true perseverance looks like, and I am grateful for the motivation and friendship you brought into my journey. A big shoutout to my buddy **Albin**, with whom I shared not only countless moments at CVC but also many adventures and conversations beyond it. From research discussions to spontaneous outings, you were a constant presence — a reminder that friendship can make both the work and the life around it richer and more meaningful. To **Richard** and **Ramon**, with whom I shared many moments at CVC, yet whose friendship has continued to thrive well beyond those walls — thank you for keeping our bond alive and as strong as ever. To Laura, Nuria, Mireia, Natalia, Pedro, Marc, Joan Masoliver, Raquel, Ana Maria, Montse, Gigi, and Kevin — thank you for being so incredibly caring and for always making sure I had everything I needed at CVC. Your support, warmth, and attention made all the difference — truly the best thing one could ask for in a place like this. To **Silvia** and **Claire** — you were always like big sisters to me at CVC, offering support, laughter, and wisdom when I needed it most. The memories we shared, especially with the unforgettable Summer Party crew — Dimos, Guillem, Andrey and Adri — are among the dearest of this journey.

And now to my special girlies — starting with **Andrea Millán**, the flower to my soul if I were the butterfly. You are the most joyfully bubbly person one could ever meet, my partner-in-crime through it all. Thank you for filling this journey with light, laughter, and endless warmth. And **Nora Graichen** — the person who became my pillar of support during my hardest times. You were always there, no matter what, with strength, clarity, and care. As clever and sharp as you are kind, I've learned so much from you — and continue to. Thank you for holding space for me when I needed it the most. To my best friend **Missi** — the sunshine of my life, my Miss Surprise — thank you for being the unwavering light through it all. I truly wouldn't have survived this journey without you. Your love, your magic, and your constant presence made everything possible. You took care of me in every way, big and small, and never once let me feel the distance from my family. In moments of stress, joy, or uncertainty, you were always there with open arms, a listening ear, or a much-needed laugh. You are, without a doubt, the best gift Barcelona has given me, and I will cherish that forever.

To my amazing professors — Maria, Ramon, Fernando, Ernest, Lluís, Oriol, Bogdan, Javier, and Joost — thank you for always inspiring me. I've learned so much from each of you over these years, and your teachings have left a lasting mark on both my academic and personal growth. A very special thanks to **Dimos** and **Alicia** — you've been more than just professors. With both of you, I've shared not only meaningful conversations about research, but also deep reflections about life. Your presence has been a true support system, and I'm endlessly grateful for your wisdom and care. To my lab family — a group of people who came and went during my time here, but who each left a lasting mark on me. To Yi, Yixiong, Danna, Guillem, David, Matias, Yeray, Miruna, Shaolin, Sandesh, Ramzy, Artemis, Carlos, Pau Torras, Adri, with whom I shared countless moments — from intense weekends meeting deadlines to enjoying meals together during breaks — thank you for the camaraderie, the laughter, and the shared determination. A special nod to Kai, Fei, Shiqi and Yaxing, whose brilliance, curiosity, and dedication have been a constant source of inspiration throughout my Ph.D. journey. To **Guillermo**, who has always inspired me and been like an elder brother, thank you for your guidance, encouragement, and the steady presence you have been through-

out this journey. To my present flatmate, **Nello**, thank you for being a steady source of support during the hardest moments of the final stretch of my Ph.D., reminding me daily that I was not facing the last mile alone.

I am also deeply grateful to **Jaime**, who quickly became like a brother to me, and to his partner, **Laura**, who welcomed me as part of their family in Barcelona. Their kindness, warmth, and companionship made this city feel like home. It has been a joy to share in their happiness as they welcomed little Martín into the world during this Ph.D. journey—a reminder that while this thesis marks one chapter of my life, it has also been intertwined with the beginnings of new ones. To my young bees in the Uni — starting with Adarsh, Neil, Dani and Joan, and my special kiddos **Nil** and **Maria** — thank you for bringing your unique spark into my life. To Nil and Maria in particular, you became my siblings I've never had, and we shared so many moments, conversations, and little adventures together that I will always carry with me. And to my Bengali bros — **Dipam**, **Alloy**, **Subhajit** and **Soumitri** — your presence has been a gift. You're all such special people, and I truly couldn't imagine this journey without you. A special word for Alloy, Subhajit and Soumitri, with whom I collaborated extensively throughout this research journey. I would also like to thank **Jordy Van Landeghem**, my leading collaborator on the DUDE project and a true big brother throughout this journey. Jordy not only showed me the way in navigating research challenges but also generously shared his own Ph.D. experiences with me — offering guidance, encouragement, and the kind of honest advice that only comes from someone who has walked the same path. I am equally thankful to **Silvia** and **Konstantina**, fellow researchers and friends with whom I shared countless conference memories; their companionship and shared experiences made every academic gathering far more meaningful and memorable. Among these moments, one that will forever remain special is *ICDAR 2024 in Athens*, where the joy of collaboration and friendship reached its peak — and where I had the honor of receiving the *Best Student Paper Award*, shared with my colleague and friend **Ayan**.

I also wish to express my heartfelt gratitude to my family in India — my uncle, my aunty, my beloved cousin **Avrajit** and my brother from another mother **Aman** — for being there for my mother throughout these years. Knowing that she was surrounded by your care, love, and support gave me the peace of mind to focus on my work while being far away. To my closest friends from India with a special mention to Aman, Shanku, Suraj, Suvojit, Somnath, Shreya and Riyanka, thank you for always being there to cheer me up, to listen, and to stand by me through every high and low. Your moral support meant more than words could ever capture — especially in the most difficult chapter of this journey, when I lost my father. In those moments, your presence, calls, and messages became lifelines that reminded me I was not alone, even from across the distance. I also hold a special place in my heart for my late grandmother, **Tandra Ghosh Roy**, who was like a second mother to me and whose love and guidance shaped so much of who I am. Losing her during this journey was another reminder of how deeply our lives are intertwined with those we love. Though she is no longer here, her blessings and strength continue to guide me, just as yours do, Maa. And finally, to my beloved Maa (mother), **Jonaki Biswas** — the "love of my life", my constant, my strength. You are not just the last name in this list, but the very reason it exists at all — the purpose behind every step I take and every milestone I reach. Your boundless love, quiet sacri-

ICDAR 2024, Athens — a highlight of my Ph.D. journey, marked by collaboration, community, and the Best Student Paper Award shared with my colleague Ayan.

fices, and unwavering belief in me have been the foundation upon which this journey was built. Every achievement here carries your reflection, every page bears the imprint of your strength. For everything — thank you, from the depths of my heart.
This thesis is not only the story of my work, but also of many hands, hearts, and histories that carried me forward. I am humbled to have walked this path with all of you — but most of all, I am grateful to have walked it for you. And if every journey tells a story, then this one, in the truest sense, is OURS.

"Alone we can do so little; together we can do so much."
— Helen Keller

.
.

Abstract

Understanding the visual and structural language of documents is central to Document AI. This thesis explores the hypothesis that **layout acts as a latent language**—a structured grammar that governs how information is arranged and interpreted in visually rich documents. Departing from traditional OCR-centric pipelines, we investigate layout-aware approaches across three interlinked axes: *Interpretation*, *Representation*, and *Generation*. In the **Interpretation** axis, we introduce transformer-based segmentation frameworks, including SWINDOCSEGMENTER and its semi-supervised extension SEMIDOCSEG, enabling precise instance-level parsing in both high-resource and low-resource settings. For **Representation**, we develop self-supervised and graph-based models such as SELFDOCSEG and DOC2GRAPHFORMER, learning robust, task-agnostic embeddings that capture visual, spatial, and relational cues without reliance on annotated data. In the **Generation** axis, we propose layout-conditioned generative frameworks—DOCSYNTH, DOCSYNTHV2, and SKETCHGPT—that model documents as sequences of layout primitives and enable controllable synthesis, sketch completion, and document design.

The collective contributions of this thesis establish a unified perspective of layout as both signal and structure, enabling end-to-end systems that not only read but reason and generate with layout awareness. We demonstrate the practical value of these contributions through deployments in real-world document intelligence systems and by proposing new benchmarks for multimodal document reasoning. This work opens new frontiers in treating layout not as noise to be removed, but as a language to be learned.

Keywords – Computer Vision, Pattern Recognition, Document AI, Document Understanding, Layout Understanding, Document Layout Analysis, Vision-Language Models, Instance Segmentation, Self-Supervised Learning, Semi-Supervised Learning, Graph Neural Networks, Document Generation, Layout as Language, Multimodal Reasoning, Structured Document Synthesis

Resum

Comprendre el llenguatge visual i estructural dels documents és essencial en la Intel·ligència Artificial Documental. Aquesta tesi explora la hipòtesi que el **disseny de l'estructura de pàgina actua com un llenguatge latent**, una gramàtica estructurada que regeix com s'organitza i interpreta la informació en documents visualment rics. Allunyant-nos dels enfocaments tradicionals centrats en OCR, investiguem mètodes conscients de l'estructura al llarg de tres eixos interconnectats: *Interpretació*, *Representació* i *Generació*. En l'eix d'**Interpretació**, presentem arquitectures basades en *Transformers* com ara SWINDOCSEGMENTER i la seva extensió semi-supervisada SEMIDOCSEG, que permeten una segmentació precisa a nivell d'instància tant en escenaris amb recursos com amb pocs recursos etiquetats. A l'eix **Representació**, desenvolupem models auto-supervisats i basats en grafs, com SELFDOCSEG i DOC2GRAPHFORMER, que aprenen representacions robustes, agnòstiques a la tasca, integrant senyals visuals, espacials i relacionals sense necessitat d'anotacions. A l'eix de **Generació**, proposem marcs generatius condicionats a l'estructura com DOCSYNTH, DOCSYNTHV2 i SKETCHGPT, que modelen els documents com seqüències de primitives estructurals, permetent síntesi controlada, auto-completat d'esbossos i generació estructurada de documents.
Les contribucions d'aquesta tesi estableixen una visió unificada del disseny de pàgina com a senyal i estructura, permetent sistemes que no només llegeixen sinó que també raonen i generen amb consciència del disseny. Demostrem el valor pràctic d'aquests avenços mitjançant aplicacions reals d'intel·ligència documental i proposem nous bancs de proves per al raonament multimodal. Aquest treball obre noves fronteres per tractar el disseny no com a soroll, sinó com un llenguatge que cal aprendre.
Paraules Clau – Visió per Computador, Reconeixement de Patrons, Intel·ligència Artificial Documental, Comprensió de Documents, Comprensió del Disseny de Pàgina, Anàlisi del Disseny de Documents, Models Visió-Llenguatge, Segmentació per Instàncies, Aprenentatge Auto-supervisat, Aprenentatge Semi-supervisat, Xarxes Neuronals de Grafs, Generació de Documents, Disseny com a Llenguatge, Raonament Multimodal, Síntesi Estructurada de Documents

Resumen

Comprender el lenguaje visual y estructural de los documentos es fundamental en la Inteligencia Artificial Documental. Esta tesis explora la hipótesis de que el **diseño de la estructura de página actúa como un lenguaje latente**, una gramática estructurada que rige cómo se organiza e interpreta la información en documentos visualmente complejos. Alejándonos de los enfoques tradicionales centrados en OCR, investigamos métodos basados en la estructura a lo largo de tres ejes interrelacionados: *Interpretación*, *Representación* y *Generación*. En el eje de **Interpretación**, introducimos arquitecturas basadas en *Transformers* como SWINDOCSEGMENTER y su extensión semi-supervisada SEMIDOCSEG, que permiten una segmentación precisa a nivel de instancia en contextos tanto con abundancia como escasez de datos etiquetados. En el eje de **Representación**, desarrollamos modelos auto-supervisados y basados en grafos, como SELFDOCSEG y DOC2GRAPHFORMER, que aprenden representaciones robustas, agnósticas a la tarea, integrando señales visuales, espaciales y relacionales sin necesidad de anotaciones. En **Generación**, proponemos marcos generativos condicionados al diseño de página como DOCSYNTH, DOCSYNTHV2 y SKETCHGPT, que modelan los documentos como secuencias de primitivas de diseño, habilitando la síntesis controlada, la auto-completación de bocetos y el diseño estructurado de documentos.
Las contribuciones de esta tesis establecen una perspectiva unificada del diseño de página como señal y estructura, permitiendo sistemas que no solo lean, sino que también razonen y generen con conciencia del diseño. Mostramos el valor práctico de estos aportes mediante despliegues en sistemas reales de inteligencia documental y proponiendo nuevos benchmarks para el razonamiento multimodal. Este trabajo abre nuevas fronteras para tratar el diseño no como ruido a eliminar, sino como un lenguaje que debe aprenderse.
Palabras Clave – Visión por Computador, Reconocimiento de Patrones, Inteligencia Artificial Documental, Comprensión de Documentos, Comprensión del Diseño de Página, Análisis de Diseño de Documentos, Modelos Visión-Lenguaje, Segmentación por Instancias, Aprendizaje Auto-supervisado, Aprendizaje Semi-supervisado, Redes Neuronales de Grafos, Generación de Documentos, Diseño como Lenguaje, Razonamiento Multimodal, Síntesis Estructurada de Documentos

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Design is the silent ambassador of your brand.
– Paul Rand

Documents are more than containers of text—they are structured canvases where language, layout, and logic converge. *In this thesis, we explore the hypothesis that layout itself functions as a visual language—one that can be parsed, represented, and generated by intelligent systems. From fine-grained layout segmentation to structure-aware representation learning and layout-conditioned document synthesis, this work examines the cognitive scaffolding that enables machines to read and compose documents like humans. The journey begins by revisiting the fundamental role of documents in our daily lives and builds toward a unified view of Document AI where interpretation, representation, and generation converge under the lens of layout understanding.*

1.1 Reading Systems in our Daily Life

Documents are deeply embedded in nearly every aspect of our daily lives—ranging from administrative paperwork and financial records to handwritten notes and cultural archives [141]. The modern world generates a vast and diverse ecosystem of documents that vary not only in content but also in layout, visual structure, modality, and purpose [77]. While early optical character recognition (OCR) systems primarily focused on extracting textual content [36, 279], the complexity of contemporary documents necessitates a deeper, more holistic understanding that transcends mere character recognition [63, 189, 125]. This evolution from simple text extraction to intelligent

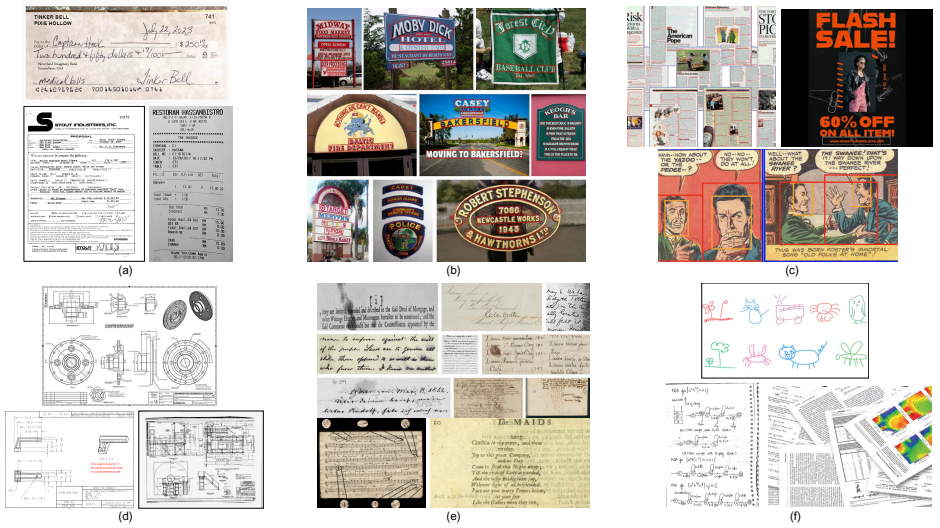


Figure 1.1: A diverse set of document types showcasing the visual and structural complexity of reading systems encountered in daily life.

document processing (IDP) marks a critical shift in the field of Document Artificial Intelligence (AI) or Visually-rich Document Understanding (VrDU), which *combines computer vision and Natural Language Processing (NLP) to interpret a document's textual content and its structural and visual cues* [237, 46, 29].

The pervasive role of documents across diverse real-world domains underscores the need for intelligent reading systems that go beyond basic text recognition to achieve human-like understanding. As illustrated in Figure 1.1, the types of documents we engage with daily—from financial reports and engineering blueprints to handwritten notes, cultural media, and environmental signage—span a broad spectrum of layouts, languages, and formats. This diversity presents a two-fold challenge: automated document understanding systems must not only extract explicit textual content but also interpret the implicit meaning encoded in spatial arrangements, typographic cues, and hierarchical structures. Bridging this perceptual-semantic gap is central to advancing machine reading in complex visual documents.

1.2 The Visual Grammar of Documents

Just as grammar governs the structure and meaning of written language, document layouts encode a **visual grammar** that guides human comprehension and interaction. Across domains—from legal contracts and architectural blueprints to historical manuscripts and comic books—this grammar manifests in the spatial organization of text, images, symbols, and whitespace. These visual structures carry semantic weight,

Table 1.1: Reading systems across document domains: motivations and cognitive insights.

Domain	Practical Motivation	Cognitive / Perceptual Insight
Fintech / Legaltech (forms, invoices, cheques)	Automated workflows require reliable extraction of key–value pairs, signatures, and stamps.	Humans group totals, dates, and parties using Gestalt proximity [266]; models replicate such grouping strategies.
Technical Diagrams (mechanical, electrical)	Accurate retrieval of specifications improves CAD/CAM productivity.	Experts follow spatial continuity, aligned with Gestalt's principle of good continuation [67].
Scene Text (signage, storefronts)	Mobile OCR aids real-time navigation and accessibility in dynamic viewpoints.	Eye-tracking studies reveal rapid fixations on high-contrast lettering [239].
Cultural Media (infographics, comics, newspapers)	Digitisation supports archiving, semantic indexing, and media analytics.	Reading order is layout-dependent; comprehension is tied to spatial organization [44].
Historical Archives (manuscripts, registries)	Layout-aware transcription helps preserve fragile and non-standard formats.	Split-attention occurs when marginalia overlaps with main body text [234].
Handwriting & Sketching (notes, calligraphy)	Real-time parsing enhances pen-based interaction and annotation systems.	Visual working memory chunks strokes using proximity and enclosure cues [267].
Scientific Papers (journals, preprints)	Structured extraction supports summarization, citation indexing, and semantic retrieval.	Readers rely on layout cues (titles, references) for hierarchical comprehension [57].

directing reading order, clarifying relationships, and shaping the user's understanding of content.

Building on the diverse use cases outlined in Figure 1.1, we now delve deeper into specific domains where the *interplay between layout, semantics, and visual structure defines the document's meaning*. Each of these domains presents unique computational challenges that demand tailored modeling approaches for reliable machine understanding. From the spatial logic of financial forms to the narrative flow of cultural media and the variability of scene text in natural environments, these examples highlight the spectrum of document modalities our systems must accommodate.

Table 1.1 captures the interplay between practical demands and human perceptual strategies across a variety of document domains. Each domain presents unique challenges that extend beyond textual recognition, requiring systems to understand the layout semantics that humans process almost subconsciously. The practical motivations span automation, accessibility, and archival needs, while the cognitive insights

underscore how humans extract meaning through visual grouping, spatial reasoning, and attention mechanisms. What this comparison reveals is that human readers consistently rely on visual heuristics—such as proximity, continuity, and enclosure—to impose structure on visual information. These perceptual rules function like an invisible scaffold that organizes our attention and supports comprehension. For instance, readers of scientific literature don't just decode words; they navigate through sections, tables, and references using spatial landmarks that encode hierarchy and emphasis. Similarly, when interpreting comics or technical diagrams, readers infer sequences or relationships not through syntax, but through spatial design and visual alignment.

From a machine learning perspective, these cognitive cues offer powerful guidance for building document understanding models. *Rather than treating layout as an auxiliary signal, it should be recognized as a primary modality*—a carrier of semantics just as crucial as text or image features. By grounding AI systems in these human-inspired perceptual principles, we can build more robust models that generalize better across domains, adapt to visual variability, and align more naturally with how people read and reason about documents. The following subsections present representative domains that shape the design of modern document AI systems, illustrating the semantic role of layout across both structured and creative formats.

Financial and Legal Documents: Checks, invoices, contracts, and legal agreements are characterized by highly structured layouts where the spatial arrangement of elements carries significant semantic knowledge. Structured understanding of these documents is paramount for enabling automation in critical sectors such as fintech, legaltech, and insurancetech, facilitating tasks like automated data entry, compliance checks, and fraud detection [45, 3].

Technical Diagrams: Electrical schematics, mechanical blueprints, and network diagrams convey complex information primarily through visual and spatial relationships. Here, the layout dictates semantic connections—such as component linkages or process flows—that are not explicitly captured by text alone, demanding specialized interpretation methods [171, 221].

Scene Text and Signage: Text embedded within natural environments, such as street signs, product labels, or building facades, presents unique challenges. Reading systems must robustly detect and interpret text under demanding real-world conditions, including varying lighting, occlusions, diverse fonts, and significant skew, requiring sophisticated computer vision techniques [124, 185].

Cultural Media: Newspapers, comic books, and traditional literary works integrate text, images, and sophisticated layouts to present structured narratives or informational content. Understanding these documents requires systems that can discern reading order, panel segmentation, and the interplay between visual and textual elements to reconstruct the intended story or information flow [110, 250, 5].

Historical Archives: Fragile and often handwritten manuscripts from historical collections necessitate advanced document analysis for preservation and accessibility. These documents often feature unique scripts, degraded paper, and complex, non-

standard layouts, demanding specialized restoration techniques and layout-aware transcription for accurate digitization and scholarly analysis [186, 200, 230].

Handwriting and Sketches: Informal documents, ranging from personal notes to design sketches, exhibit highly creative and often unstructured layouts. Flexible document representations are required to capture the nuanced visual information and contextual relationships inherent in such free-form content [218, 177, 198].

Scientific Papers: Scientific articles exhibit unique multimodal layouts that encode both content and meta-information such as affiliations, references, equations, and figures—each serving distinct semantic roles. Beyond their hierarchical structure, these documents often contain cross-references (e.g., "see Fig. 2" or "as discussed in Section 3.1"), which require the reader to resolve long-range dependencies across sections [291]. Human readers effortlessly navigate these links using layout cues like caption placement, typographic variation, and figure alignment. For machine understanding, modeling such document-wide relationships remains a challenge.

1.3 Quantifying Document Layout Through Human Interaction

A **document layout** is the spatial and visual organisation of all communicative elements on a page or screen—text blocks, images, rules, whitespace, ornaments—together with the reading paths they implicitly prescribe. In other words, layout is the "syntax" that governs where content appears, how readers' eyes are steered, and which relationships they infer among neighbouring elements. When people skim a newspaper, follow a flowchart, or sign a cheque, they implicitly parse this syntax: they group nearby items (Gestalt proximity), follow aligned baselines (good continuation), and prioritise salient headers before subsidiary details. Document AI systems must learn to exploit the same cues. To make this notion operational, we treat layout as a set of quantifiable features derived from patterns observed in human document interaction as illustrated in Table 1.2. These measurable cues allow us to cast *layout understanding* as structured prediction over: (i) *Zones* - coherent regions such as paragraphs, tables, or panels. (ii) *Relations* - adjacency, containment, ordering, caption-of, etc. (iii) *Reading Graph* - a directed graph whose edges approximate typical human scan-paths.

Understanding how humans interact with documents offers a grounded framework for modeling layout as a structured and perceptual language. Document layouts are not arbitrary. Humans rely on consistent visual cues to extract meaning, follow reading flow, and associate semantically linked elements. These cues manifest in observable behaviours such as eye fixations, saccadic jumps, attention to prominent visual elements, and multimodal alignment between text and images. We conceptualize these behaviours into four layout-level attributes: *grouping*, *reading hierarchy*, *salience*, and *cross-modal anchoring*, each of which can be quantified and modeled computationally. As illustrated in Figure 1.2, these layout cues form the bridge between perceptual

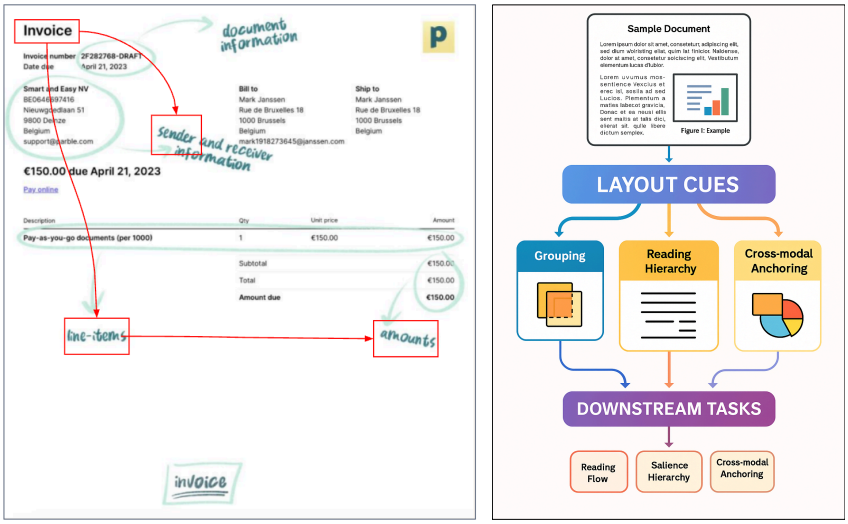


Figure 1.2: **Visual cues in document layouts support downstream understanding.** *Left*: Human annotations reveal perceptual grouping and reading flow in an invoice. *Right*: Diagram illustrating how layout cues—grouping, reading hierarchy, and cross-modal anchoring—serve as structured signals for document intelligence tasks.

Table 1.2: Mapping observable reading behaviours to the layout cues for Document AI

Human behaviour	**Layout attribute**	**Quantifiable signal**
Eye–fixation clusters on logically related items	**Grouping**	Euclidean proximity, shared bounding boxes, connected components
Saccades that follow columns or diagram edges	**Reading hierarchy**	Ordered zone graph; path-length and orientation statistics
Rapid detection of titles, captions, signatures	**Salience**	Font-size / weight distribution, colour contrast, recursion depth in zone-tree
Integration of text with nearby figures	**Cross-modal anchoring**	Alignment offsets and relative overlap between text and image regions

behaviour and downstream document understanding tasks. On the left, we present a real-world invoice annotated with human attention and interaction flows, evidencing how people naturally group related fields like invoice metadata, sender–receiver details, and line-item amounts. On the right, we abstract these insights into a processing pipeline: a document first emits layout cues which are then interpreted through cue-specific detectors. These detectors responsible for identifying visual grouping, reading order, and alignment between modalities feed into broader downstream tasks such as

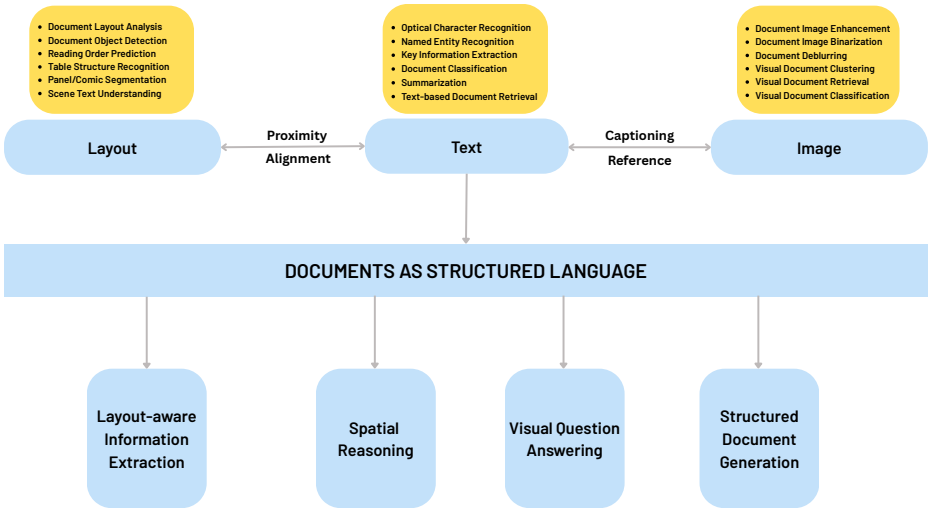


Figure 1.3: The **"Layout as Language" paradigm in Document AI**, illustrating how layout, text, and image modalities interact to support core tasks such as information extraction, spatial reasoning, question answering, and document generation.

reading flow estimation, salience prediction, and cross-modal understanding.

For example, *grouping* emerges through the proximity and alignment of invoice fields, which humans perceive as logically connected blocks. *Reading hierarchy* is informed by typography and spacing, allowing the user to follow a predictable flow from titles to details. *Salience* drives rapid attention to prominent entities such as the amount due, typically through bold text or isolated positioning. Finally, *cross-modal anchoring* connects figures or tables with their surrounding descriptive text, a process humans perform seamlessly but machines must learn to replicate. By grounding layout understanding in these cognitive principles and modeling them with quantifiable signals (as outlined in Table 1.2), we enable document AI systems towards intelligent layout-aware perception that do more than detecting bounding boxes or parsing raw text.

1.4 Modeling Documents as Structured Language

Understanding documents is inherently a multimodal and spatial task. In the preceding sections, we analyzed how humans engage with documents by relying on visual grammar—a set of perceptual cues that guide reading behavior, interpretation, and comprehension. These layout cues, such as grouping, reading flow, salience, and cross-modal anchoring, shape how meaning is inferred from spatial structure. The question now becomes: **how can machines learn to interpret documents with similar fluency?** In this thesis, we embrace the paradigm of **"Layout as Language"**, wherein

the spatial organization of document elements functions analogously to linguistic syntax. Just as words in a sentence derive meaning from their position and order, document elements such as headers, tables, figures, and paragraphs derive communicative power from how they are arranged, aligned, and related within a page. This view enables a shift from isolated text recognition toward a structured, multimodal understanding of documents. We no longer treat layout, text, and image as disjoint channels but rather as interacting components of a compositional system. For example, proximity between a figure and caption suggests anchoring; larger and bolded text signals hierarchy; and the vertical alignment of form fields implies grouping and semantic equivalence. These are not just visual artifacts—they are meaningful signals for information extraction, question answering, and document generation.

Figure 1.3 illustrates this conceptual leap from perception to modeling. It positions layout, text, and image as cooperative modalities, each contributing to the structural language of documents. Their interactions give rise to downstream applications that require layout-aware reasoning. By modeling these interactions computationally, we aim to move closer to *human-like document understanding*. This involves creating machine learning systems that learn to parse, reason, and generate over structured documents in ways that reflect human visual processing. In the following section, we present a taxonomy of such modeling strategies, ranging from layout-only models to fully multimodal architectures.

1.5 Modeling Document Layouts in Document AI

Building on the conceptual framing of documents as structured languages—where layout, text, and images function in a compositional interplay—we now focus on the computational strategies that allow machines to model such structures effectively. The goal of layout-aware modeling in Document AI is to capture not just the semantics of textual content, but also the spatial, visual, and contextual cues embedded in the document's layout. These cues often encode essential information about logical grouping, reading order, semantic hierarchy, and cross-modal referencing that are indispensable for downstream understanding tasks.

In the context of this thesis, we define modeling document layouts as the design and implementation of learning algorithms that explicitly account for layout signals, either independently or in combination with other modalities (text and image), to perform structured reasoning over document content. This modeling process involves both the representation of the input modalities (i.e., how layout features are encoded) and the fusion strategies that align these features across modalities.

1.5.1 A Taxonomy of Modeling Approaches

We classify the modeling space into three core streams based on the modalities involved and the nature of their integration:

Layout-only Models: These approaches rely purely on spatial or geometric structures of documents. Methods in this class include object detection over layout zones [26, 24], graph-based models over bounding boxes [211, 212, 78], or clustering/grouping heuristics [264, 163]. They are especially effective for tasks like document layout analysis and reading order prediction, where visual or structural arrangement is the primary.

Layout+Text Fusion Models: Here, layout information is paired with textual features, often using encodings such as bounding-box coordinates, 2D positional embeddings, or graph-based adjacency. Popular models like LayoutLM [274], StrucTexT [157], BROS [101] or XYLayoutLM [84] fall under this category. These models are particularly suited for tasks like Key Information Extraction (KIE) [113, 106], where context is jointly defined by spatial and semantic features.

Multimodal Layout-Text-Image Models: These models aim to harness the full richness of document data by simultaneously incorporating layout, text, and image features. They use fusion mechanisms typically based on transformer architectures [249] to model the interplay between modalities. Examples include DocFormer [6, 7], SelfDoc [152], UniDoc [82], UDOP [238], and Donut [129]. Such models enable performance on complex tasks such as Visual Question Answering (DocVQA) [180, 240], document editing [182], and captioned generation [83].

1.5.2 Modeling Objectives and Challenges

The primary goal of layout modeling is to learn representations that mirror the structural semantics perceived by humans. From this perspective, the modeling strategy should aim to:

- Preserve document structure through explicit spatial encoding.
- Enable reasoning across zones, sections, or entities.
- Adapt across domains (e.g., invoices, scientific PDFs, handwritten forms).

However, this modeling paradigm introduces several key challenges:

- Spatial alignment noise due to OCR inaccuracies or image artifacts.
- Cross-modal grounding difficulties when layout and text regions do not align neatly with visual elements.
- Data sparsity for layout-rich annotations compared to plain text datasets.

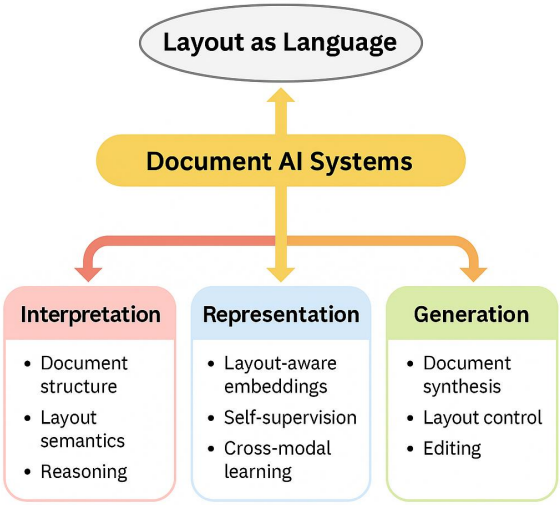


Figure 1.4: **Core Axes of the Thesis—Interpretation, Representation, and Generation** This figure illustrates the three foundational axes guiding the thesis: *Interpretation*, which investigates how layout cues guide human-like understanding of documents; *Representation*, which focuses on learning unified multimodal embeddings that capture the structure and semantics of layout-text-image compositions; and *Generation*, which explores layout-conditioned document synthesis to enable controllable and semantically grounded outputs.

- Modality imbalance, where the model over-relies on text despite layout or image cues being decisive.
- Understanding how models leverage layout remains opaque; probing and explainability methods are still emerging.

In summary, modeling document layouts requires moving beyond surface-level extraction toward a deeper, structural understanding of how humans interact with visual information. By treating layout as a compositional language, we acknowledge its fundamental role in shaping meaning across diverse document types. Addressing the aforementioned challenges demands not only new architectural frameworks for Document AI models but also a rethinking of representation, grounding, and evaluation strategies. The remainder of this thesis is organized around three core axes (interpretation, representation and generation) as illustrated in Figure 1.4 which together form a comprehensive approach to treating layout as a first-class citizen in Document AI. This perspective sets the stage for a new generation of models that more faithfully emulate human document understanding and production.

1.6 Scope and Research Questions

This thesis investigates how documents, as structured visual languages, can be understood, represented, and generated by intelligent systems. At the heart of this inquiry is the idea that **layout is not just a carrier of content but a language in itself**—one that orchestrates the interplay between text, image, and spatial organization to convey meaning. Building on the previous sections where we explored the visual grammar of documents and introduced the modeling taxonomy in Document AI, this thesis narrows its focus to how layout-driven document modeling enables machines to perform complex reasoning and generation tasks. The central hypothesis is that integrating layout signals into modeling pipelines can significantly enhance the machine's capacity to mimic human-like document understanding.

Research Question 1: How can we move beyond bounding-box detection and reliably segment every document element at pixel level—even when objects overlap or are nested?

Axis : Interpretation

Objective: Build document layout parsing systems that go beyond basic object detection and instead achieve instance-level segmentation, providing finer granularity. The goal is to preserve complex visual boundaries, manage overlaps between logical elements such as tables, images, and text, and generalize across varied document types like scientific PDFs, invoices, magazines, and archival scans.

Contribution: We introduce a robust baseline for instance-level document segmentation [26] by adapting the Mask R-CNN framework [95] specifically to the unique challenges posed by document images. Unlike traditional document layout analysis methods that rely on coarse bounding boxes or heuristic-driven segmentation, our approach emphasized pixel-level precision, enabling the delineation of overlapping, nested, and visually subtle regions such as stamps, tables, annotations, and logos. This represents one of the first systematic efforts to benchmark fine-grained document segmentation, addressing the growing need for high-resolution structure understanding in Document AI. Experimental evaluation for the benchmark suite was carried on 2 diverse data domains - scientific articles [291] and historical Japanese documents [224] with complex layouts.

Research Question 2: Can transformers capture the "layout grammar" of documents so that models reason over long-range spatial dependencies as naturally as humans?

Axis : Interpretation

Objective: To explore how spatial layout, visual features, and text as a visual language can be jointly modeled in a transformer framework, enabling the network to infer relationships between distant parts of a document—such as columnar reading order, grouped elements, or referencing structures (e.g., a figure caption far from the figure).

Contribution: We introduce DocSegTr [24], a novel and the first transformer-based ar-

chitecture purpose-built for document segmentation. Unlike traditional models that often struggle to encode fine-grained layout structures, DocSegTr incorporates a *twin-attention mechanism* that jointly captures both local and global layout cues. This enables the model to learn visual grouping, reading flow, and hierarchical region relationships—elements critical to interpreting the semantics of a document's structure. A major innovation in this work is the introduction of an *inverse focal loss*. Traditional loss functions often bias learning toward frequently occurring and large-sized layout entities, which can result in poor recall for smaller component regions like signatures, headers, or logos. Our introduced focal loss formulation addresses this imbalance by explicitly up-weighting the loss contribution of such underrepresented or small instances, significantly improving recall without sacrificing precision. Beyond architecture and loss design, the chapter also provides an insightful interpretability study, where we visualize the attention maps learned by the model. These visualizations reveal that DocSegTr attends to document regions in a manner that mirrors human reading behavior—moving along columns, linking related content, and attending to salient zones like titles and captions. Just as syntactic structures in language help disambiguate meaning, layout structures in documents guide comprehension and intent. Through this lens, DocSegTr demonstrates how layout can be modeled compositionally, akin to linguistic syntax, bridging perceptual layout signals with semantic understanding.

Research Question 3: Can advanced mask-based architectures be adapted for fine-grained document layout segmentation across both high-resource and low-resource settings, while preserving layout semantics and ensuring generalization to diverse real-world domains?

Axis : Interpretation

Objective: To develop and evaluate end-to-end segmentation models that encode and preserve document layout structure at the pixel level. This involves extending strong object-centric detectors (like MaskDINO [144] and Swin Transformers [169, 168]) to support fine-grained segmentation, particularly in settings with complex hierarchies (e.g., invoices, forms, scientific papers) and in scenarios with limited annotated data (eg. magazines, posters). Special focus is given to cross-domain generalization, robustness under annotation sparsity, and semantic alignment with layout cues such as grouping, reading flow, and salience.

Contribution: In this thesis, this effort is presented with two complementary contributions: (i) *SwinDocSegmenter:* We adapt a hierarchical transformer backbone (Swin) within a document segmentation pipeline to explicitly capture multi-scale layout patterns and region relationships. By customizing the segmentation heads and incorporating layout-aware augmentations, our model achieves state-of-the-art performance on multiple document datasets. The use of transformer-based global attention mechanisms allows the model to learn implicit layout rules, offering interpretability and robustness. (ii) *SemiDocSeg:* Building on the same architecture, we explore few-shot and semi-supervised settings, simulating real-world scenarios where large-scale annotations are scarce. By introducing a hybrid training regime (pseudo-labeling + con-

trastive learning), we show that strong performance can be achieved with minimal supervision. This positions our method as both practical and scalable for deployment in low-resource environments. Together these contributions demonstrated that layout-aware segmentation can be achieved in both ideal and challenging settings. Moreover, our models are shown to implicitly learn key visual grammar cues such as grouping and reading order, thus reinforcing the key principle that layout can be modeled like a compositional language.

Research Question 4: Can documents be effectively represented as graphs to model both their structural layout and semantic relationships—across modalities, tasks, and languages?

Axis : Representation

Objective: To design graph-based representations that capture textual, visual, and geometric relationships within documents, enabling semantic entity recognition and relation extraction in a multilingual and task-agnostic manner.

Contribution: In this thesis, we propose a unified line of work through the Doc2Graph framework: In *Doc2Graph* [78], we develop a task-agnostic graph representation model for structured document understanding, using node and edge classifiers to jointly tackle entity recognition and relationship extraction. In *GeoContrastNet* [22], we further introduce a contrastive learning objective for graph neural networks (GNNs) that aligns geometric layout features with semantic structure, enhancing layout-aware reasoning. In *Doc2Graph-X* [183], we extend this paradigm to multilingual document processing. By integrating multilingual embeddings at both word and sentence-level, we build robust graph representations across languages, achieving strong performance on SER and RE tasks with minimal parameters. Finally, we also propose a transformer model called Doc2GraphFormer, which combines these modules enhanced by graph attention supervision. It demonstrates that graph priors—particularly hierarchical layout structures—can guide multimodal attention maps during training for improved structured understanding across tasks.These contributions establish graph-based reasoning as a powerful and efficient mechanism to encode “layout as language,” where nodes and edges act as words and syntax of document structure, applicable across both monolingual and multilingual contexts.

Research Question 5: Can layout-aware knowledge distillation preserve structural dependencies in lightweight document understanding models without significant performance loss?

Axis : Representation

Objective: To investigate whether knowledge from large, multimodal document models—capturing both layout and visual semantics—can be distilled effectively into compact student models by leveraging structured intermediate representations (e.g., graphs or token embeddings).

Contribution: In this thesis, we explore layout-aware knowledge distillation to build efficient yet structure-sensitive Document AI models. In *DistilDoc* [248], we propose a

multimodal distillation strategy that transfers both semantic and layout-specific cues from a large teacher to a lightweight transformer, achieving competitive performance with reduced size and latency. Complementing this, *GraphKD* [14] frames distillation as graph alignment, where a student model learns structural dependencies by mimicking the teacher's layout-informed graph representations. Finally, we investigate a spatial-aware lightweight solution to pre-trained Large Language Models (LLMs) over how the layout modality impacts learning across document understanding tasks. Together, they show that preserving the "grammar" of layout—even in compressed models—supports the thesis vision of layout as language, enabling lightweight models to retain meaningful document understanding.

Research Question 6: Can pixel-accurate document layout segmentation be learned largely from unlabeled pages through self-supervised objectives that treat "layout as language"?

Axis : Representation

Objective: To develop a pre-training method that learns robust layout-aware representations from entirely unlabeled document images. The goal is to enable downstream fine-tuning for segmentation with minimal annotated supervision by leveraging synthetic layout cues and self-distillation.

Contribution: We introduce a self-supervised framework called *SelfDocSeg* [174] based on a self-labeling paradigm called Bootstrap Your Own Latent (BYOL) [81], which employs two augmented document views processed through student and teacher branches. A novel *Layout Mask Generation (LMG) module* is proposed that creates layout masks from document-specific cues such as edge detection and whitespace projection without needing human annotations. These masks guide the model to focus on layout-relevant structures during training. By treating the spatial organization of content as a latent grammar, SelfDocSeg shows that document regions can be learned analogously to linguistic tokens, where alignment, grouping, and flow are captured through visual regularities. This positions layout not merely as an auxiliary input, but as a learnable syntax—one that models can internalize to navigate and interpret the semantics of a document, even in the absence of labels.

Research Question 7: Can document generation be conditioned on layout structure to produce realistic and controllable synthetic documents for training and benchmarking Document AI systems?

Axis : Generation

Objective: To develop a controllable document image synthesis framework that conditions generation on explicit layout cues such as bounding boxes, class labels, or spatial templates—allowing the creation of realistic and task-aligned synthetic documents.

Contribution: In DocSynth [27], we introduce the first end-to-end pipeline for synthetic document generation using layout as a guiding modality. The framework leverages a two-stage design: (i) *Layout conditioning stage:* Takes class-labeled layout templates and injects semantic structure into the generation pipeline. (ii) *Image generation*

stage: Uses a GAN-based model to synthesize high-resolution document images, maintaining fidelity to the conditioned layout. We demonstrate that layout-conditioned synthesis significantly improves the quality and utility of synthetic datasets, enabling their use for training downstream models (e.g., object detection, OCR). This work validates the hypothesis that layout can serve as a generative language, a guiding script for creating structured, plausible visual documents.

Research Question 8: Can autoregressive generation models, guided by layout constraints, learn to produce visually coherent and semantically meaningful documents from structured prompts?

Axis : Generation

Objective: To design and evaluate a layout-guided autoregressive modeling framework that enables controllable document synthesis — generating complex, diverse, and semantically grounded documents based on layout and content inputs.

Contribution: In DocSynthv2 [25] we propose a fully autoregressive modeling approach for generating structured documents in vector format. By jointly modeling layout and content as sequences, we enable high-resolution generation that preserves both syntactic (layout) and semantic (text) elements. The creation of the PubGenNet benchmark further supports evaluation in this direction. SketchGPT [243] further addressed the task of document completion in sketch-based generation, where partial layouts are filled using autoregressive decoding. This model can reason over incomplete visual structures, reinforcing the importance of hierarchical layout modeling and extending generation beyond clean templates. By treating layout as a latent language and leveraging sequence modeling, our models can "write" documents with both structural grammar and contextual relevance.

1.7 Thesis Structure

The thesis is organized along three core modeling axes—**Interpretation**, **Representation**, and **Generation**—each addressing distinct but interconnected aspects of Document AI. The structure is outlined below with a brief summary of contributions for each chapter.

Chapter 3 – Foundation and Froniers - Provides a comprehensive overview of prior work in document AI covering layout analysis, structured representation learning, and document generation. It traces the evolution from heuristic pipelines to end-to-end deep learning and self-supervised models to finally multimodal large language models.

Axis 1: Interpretation

Chapter 3 – Beyond Bounding Boxes: Fine-Grained Document Segmentation - We introduce a pixel-level segmentation pipeline adapted from Mask R-CNN for struc-

tured document layouts. This chapter lays the foundation for instance-level parsing with benchmark results on complex domains like scientific PDFs and historical documents.

Chapter 4 – DocSegTr: A Transformer Approach to Layout Segmentation - A twin-attention transformer architecture is presented for document segmentation, with interpretability analysis and an inverse focal loss tailored to improve recall of small layout entities. Attention visualizations reinforce the *Layout as Language* hypothesis.

Chapter 5 – Advancing Robustness in Document Layout Segmentation: From SwinDocSegmenter to SemiDocSeg - This chapter introduces a unified segmentation framework using Swin Transformers. It addresses both high-resource (SwinDocSegmenter) and low-resource (SemiDocSeg) settings, showing the model's generalization and label efficiency across diverse domains.

Axis 2: Representation

Chapter 6 – Encoding Structure as Language: Towards Graph-based Representation of Document Layouts - Presents a task-agnostic graph representation framework (Doc2Graph), extended to multilingual contexts and enhanced with contrastive learning (GeoContrastNet). A transformer variant (Doc2GraphFormer) is also proposed, modeling documents as structured graphs.

Chapter 7 – Self-Supervised Visual Representation Learning for Document Layouts - A self-supervised framework is presented for pixel-accurate layout segmentation without labels. The Layout Mask Generation (LMG) module synthesizes training signals from edge/whitespace cues, learning spatial grammar as latent syntax.

Axis 3: Generation

Chapter 8 – DocSynth: Layout-Guided Document Image Synthesis - Proposes a GAN-based generation framework conditioned on layout templates. The model enables realistic document synthesis aligned with spatial structure, validating the use of layout as a guidance signal.

Chapter 9 – Towards Autoregressive Vector Document and Sketch Generation - An autoregressive vector-based generation approach (DocSynthv2) is introduced, modeling layout and text as sequences. SketchGPT further explores sketch completion, demonstrating sequence modeling of document grammar.

Chapter 10 – Conclusion and Future Work In the concluding chapter, the thesis synthesizes the core findings across interpretation, representation, and generation—each guided by the central hypothesis that layout functions as a visual language, enabling machines to read, represent, and synthesize documents with human-like precision

and abstraction. To support these modeling advances and address the lack of evaluation standards in real-world scenarios, this thesis also contributes a suite of benchmark datasets that target underexplored yet critical aspects of Document AI, especially for multimodal reasoning over diagrams, equations, and non-standard layouts.

Chapter 2

Foundations and Frontiers

Design is not just what it looks like and feels like.
Design is how it works.
– Steve Jobs

Understanding the language of document layouts lies at the intersection of computer vision, natural language processing, and document image analysis. This chapter surveys the current state of the art, tracing developments across three major axes that shape the core of this thesis: ***Interpretation****,* ***Representation****, and* ***Generation*** *of document layouts. The discussion spans layout-aware pretraining strategies, multimodal reasoning, and document synthesis, offering both conceptual clarity and comparative analysis for design choices. By bridging historical context, methodological advances, and open challenges, this chapter sets the stage for developing next-generation systems capable of understanding, representing, and generating document layouts with human-like precision.*

Document Artificial Intelligence (Document AI) has emerged as a vital research domain that aims to automate the understanding, extraction, representation, and generation of structured information from unstructured or semi-structured document formats. This spans a wide spectrum of document types, including invoices, scientific publications, administrative forms, historical manuscripts, and handwritten notes. At its core, Document AI intersects the fields of computer vision, NLP, and machine learning to interpret the rich visual-linguistic layout of documents.

Early approaches in document understanding relied heavily on Optical Character Recognition (OCR) systems followed by rule-based parsing or handcrafted heuristics [227]. However, these methods often fail to capture the implicit layout semantics and spatial relationships inherent to structured documents, especially in noisy, scanned, or non-

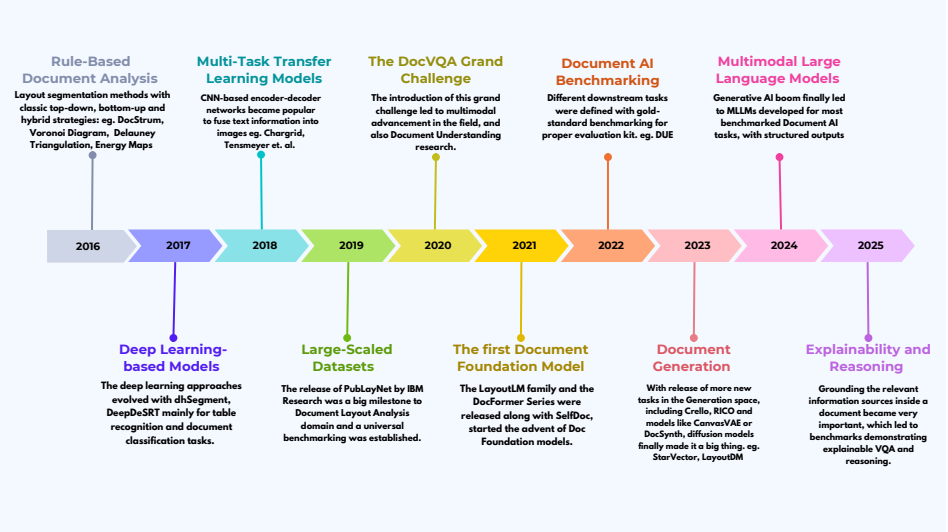


Figure 2.1: **A Timeline of Document AI Advancements**: From Rule-Based Document Layout Analysis to Multimodal Large Language Models and Explainable Reasoning

standard formats. The limitations of OCR-centric pipelines have driven a paradigm shift toward layout-aware models that leverage both visual and textual cues [274, 75]. The evolution of Document AI over the past decade has been marked by a series of paradigm shifts—from rule-based layout analysis to multimodal large language models—each expanding the boundaries of what machines can interpret, represent, and generate (Figure 2.1). This trajectory, captured in the timeline, highlights how breakthroughs in model architectures, pretraining paradigms, and large-scale benchmarking have converged to form the foundation of today's document intelligence systems.

The rise of deep learning has catalyzed rapid advancements in Document AI. Convolutional Neural Networks (CNNs) and Transformer architectures are now commonly used for layout segmentation [147], document classification [90], and key information extraction [114]. Multimodal pre-training methods, such as DocFormer [6], Donut [129] and LayoutLM [274] and integrate visual, textual, and positional embeddings to build robust document representations. Beyond understanding, recent work has explored controllable document generation [86], self-supervised learning for document representations [174], and autoregressive modeling for sketch and layout synthesis [25]. The field is also enriched by large-scale datasets such as PubLayNet [291], DocLayNet [197], and BigDocs [215], which have facilitated rigorous benchmarking and model development. In this thesis, we adopt a three-axis lens—**Interpretation**, **Representation**, and **Generation**—to organize and explore the evolving landscape of Document AI. Each axis reflects a distinct research focus, yet they are inherently interconnected in building systems capable of reasoning about and synthesizing document layouts with human-like understanding.

2.1 Inherited Capabilities: Foundational Knowledge from Vision and Language Models

Modern Document AI systems build upon pretrained vision-language backbones that have demonstrated general capabilities across natural language understanding and visual reasoning. These inherited capabilities form the substrate upon which document-specific skills are developed:

Language Understanding: Pretrained language models [61, 205, 191] provide foundational abilities for text comprehension, semantic similarity, summarization, translation, and reasoning. These are crucial for interpreting document content beyond raw OCR outputs, enabling contextual understanding and multilingual adaptability [222].

Visual Feature Encoding: Vision backbones like ViT [64], ResNet [96], Swin Transformer [169, 168], and hybrid image-text models such as CLIP [202] and BLIP [148] capture high-level spatial and visual features. These models are increasingly adapted to document understanding tasks by learning region-level representations, enabling object detection [26], segmentation [13, 291], and the visual grounding of text elements [232, 174].

Multimodal Alignment with Layout Awareness: Vision-Language Models (VLMs) extend these capabilities by learning to bridge vision and language at scale. Trained on large corpora of image-text pairs, VLMs like Flamingo [4], PaLI [40], or OFA [255] exhibit strong generalization across image captioning, visual question answering, and open-ended generation tasks. However, their application to documents introduces a critical additional modality: *layout*.

Unlike natural images, documents are governed by an implicit yet structured visual grammar — headers, paragraphs, tables, footnotes, and spatial groupings collectively encode meaning. In this context, layout is not merely a visual feature but a core semantic signal that shapes how information is structured, interpreted, and retrieved. A unified Document AI system must therefore learn to associate textual content with its spatial organization, leveraging layout-aware inductive biases that go beyond standard vision-language alignment.

This motivates the emergence of a new class of models referred to as **Document Foundation Models** — large-scale, general-purpose pretrained architectures designed to serve as adaptable backbones for a wide range of document understanding and generation tasks. Analogous to the role of BERT in NLP or CLIP in vision-language modeling, a Document Foundation Model is expected to encode rich, transferable document-specific priors across modalities, languages, and tasks. For such a foundation model to be effective, **layout understanding must be a first-class design principle**. It must not only represent the visual content of documents (e.g., font, color, figure) and their linguistic content (e.g., entity spans, semantic roles), but also the hierarchical and spatial structure that binds them. This requires integrating layout signals directly into the model architecture — via 2D positional embeddings, relative bounding-box encodings, layout-aware token fusion, or region-based attention mechanisms — enabling

Figure 2.2: **Concept map of *layout encoding* strategies in Document AI.** The first ring lists encoding families; the outer ring gives a representative model and its typical strength (italic) of how different encodings emphasize complementary capabilities.

the model to interpret and reason over documents as structured visual-linguistic entities.

An ideal Document Foundation Model should therefore be **multimodal (vision + text), multi-granular (from token to region to page), and multi-task (pretrained to support both discriminative and generative objectives)**, with layout as a guiding axis. This thesis explores how such layout-centric modeling can be realized in practice — not only through task-specific architectures but also through shared design principles that treat layout as a latent language to be decoded and composed.

Table 2.1: Comparison of prominent Document Foundation Models: layout encoding type, OCR reliance, pretraining objectives, and supported tasks.

Model	**Layout Encoding**	**OCR Dep.**	**Pretraining Tasks**	**Tasks Supported**
LayoutLMv3 [104]	2D Positional	✓	MLM; Image–Text Alignment	KIE; DocVQA; Form Understanding
Donut [129]	None (OCR-free)	✗	Visual Masking; Sequence Decoding	Form Extraction; Generation; QA
DocFormer [6, 7]	Text + Image + Layout	✓	MLM; Cross-modal Masking	KIE; VQA; Classification
DiT [147]	Visual Layout	✗	Masked Image Modeling	Segmentation; Classification
Pix2Struct [140]	Image Grid	✗	Image-to-Text Translation	Captioning; QA; Structured Generation
StrucTexT [157]	Relative Layout	✓	Field Supervision	Entity Linking; Field Extraction
ERNIE-Layout [196]	Hierarchical Layout	✓	Entity-level Graph Reasoning; MLM	Long Document Understanding; Form Parsing
Doc2Graph++ [78, 22, 183]	Graph-based Layout	✓	Graph Contrastive Learning; Node Classification	Task-agnostic Document Parsing; Relation Extraction
PaLI / OFA / UDOP [238]	Implicit via Image	✗	Multitask (VQA; OCR; Captioning)	Generalist Multimodal Document Tasks

2.2 Document Foundation Models: Incorporating Layout as a First-Class Signal

The rapid expansion of Document AI has led to the development of a variety of foundation models tailored for document-centric tasks. These **Document Foundation Models** differ from generic vision-language models in that they explicitly account for the **layout modality**, integrating spatial priors, visual context, and text semantics in unified architectures. The concept map in Figure 2.2 provides a visual taxonomy of layout encoding strategies in Document AI, grouping them into eight broad categories—2D Positional, Relative Layout, Visual Layout, Image Grid, None (OCR-free), Hierarchical

Layout, Graph-based Layout, and Implicit via Image—each represented by a state-of-the-art model. The inner ring defines the encoding family, while the outer ring lists an exemplar model alongside a typical strength or application domain, illustrating the diversity of approaches and their complementary capabilities. The expanded comparison in Table 2.1 highlights that, beyond common layout encoding strategies such as 2D positional embeddings, relative layout, visual grids, and OCR-free representations, recent work has explored hierarchical and graph-based approaches to better capture document structure. **Hierarchical layout encoding** (e.g., ERNIE-Layout [196]) models structure at multiple levels of granularity — from tokens to lines, paragraphs, and regions — allowing the model to preserve semantic grouping and logical reading order. This is particularly valuable for long or multi-page documents where information is nested. **Graph-based layout encoding** (e.g., Doc2Graph [78]) represents layout elements as nodes and spatial or semantic relationships as edges, enabling relation-aware reasoning and cross-region inference. These richer encodings complement pixel-based or coordinate-based methods by embedding relational and hierarchical priors into the model, aligning with the thesis goal of treating *layout as a latent language* and equipping Document Foundation Models with deeper structural understanding. Below, we outline the most prominent families of Document Foundation Models, focusing on their treatment of layout and their applicability across understanding and generation tasks.

Layout-Aware Pretrained Language Models: These models extend transformer-based language models by incorporating visual and spatial embeddings, enabling them to jointly process text and layout.

LayoutLMv1 [274]: The first model of the LayoutLM series from Microsoft which Introduced 2D positional embeddings (x, y coordinates) alongside token embeddings to capture spatial layout.

LayoutLMv2 [273]: Adds image features (from ResNet) for multimodal understanding, with cross-modal pretraining objectives (MLM + Image-Text Alignment).

LayoutLMv3 [104]: Utilizes a unified masked pretraining approach over image, text, and layout tokens; demonstrates strong performance on DocVQA, CORD, and FUNSD.

LayoutXLM [275]: A multilingual extension of LayoutLMv1, supporting cross-lingual document tasks while preserving layout sensitivity.

StrucTexT/StrucTexTv2 [157, 284]: Encodes hierarchical structure via relative layout attention and introduces block-level supervision signals for better field-level extraction.

Vision-Enhanced Document Transformers: These models replace or augment text embeddings with visual tokens and are more robust to OCR noise or visual distortions.

DocFormer [6]: Combines text and image tokens in a single transformer sequence with layout-aware 2D positional encodings. Uses hierarchical features from textboxes, words, and regions for improved grounding.

DocLayNet + DiT [147]: DiT (Document Image Transformer) [147] is a vision-only trans-

former pretrained with masked image modeling on document pages, later fine-tuned for layout segmentation and classification. It shows that layout can be learned purely visually without OCR tokens.

LAMBERT [76]: A BERT-style model that replaces token IDs with character-level embeddings and relative spatial encoding, trained for key information extraction.

ERNIE-Layout [196]: Integrates text, layout, and image features through graph-based reasoning, extending layout modeling with entity-level edge encodings.

OCR-Free Document Understanding Models: OCR-free methods directly decode documents from raw pixels without relying on explicit text recognition steps, making them resilient to noisy or non-standard scripts.

Donut (Document Understanding Transformer) [129]: A fully visual encoder-decoder model that skips OCR, using Swin Transformer as the visual backbone and generating structured outputs directly via autoregressive decoding.

Pix2Struct [140]: Adapts ViT to encode rendered document images and decodes structured answers using a T5-style text decoder. Achieves strong results in form understanding and captioning.

FormNet / FormNetv2 [137, 138]: Enhances form understanding by modeling 2D spatial relationships with graph attention over visual and structural embeddings.

Multimodal and Cross-Domain Document Foundation Models: Recent work extends the generalist VLM paradigm to documents, training large-scale models across diverse document types and input modalities.

PaLI (Pathways Language and Image model) [40]: Trained on multilingual document-image pairs, PaLI is capable of document captioning, visual QA, and OCR tasks across 109 languages — handling layout implicitly via image rendering.

OFA (One For All) [255]: A unified model for vision-text tasks including table-to-text and image captioning. OFA encodes images and structure using flattened grid tokens and autoregressive decoding.

UDOP [238]: Unified Document Pretraining, extending Pix2Struct [140] with retrieval-based cross-document grounding and structured answer generation.

LAPDoc / DocPrompt [136, 268]: Recent prompt-tuning based methods adapt large vision-language models for document tasks using minimal labeled data, preserving layout via prompt-aware encoding strategies.

2.3 Pretraining Paradigms and Multimodal Learning for Layout

Pretraining in Document AI has evolved from task-specific pipelines to large-scale multimodal foundation models that learn from millions of image–text–layout samples. These models leverage self-supervised and weakly supervised objectives to capture textual semantics, visual features, and increasingly, layout structures. The choice of pretraining paradigm is critical—it determines how well the model can transfer knowledge across tasks, handle OCR noise or absence, and adapt to generation, retrieval, and reasoning scenarios. Unlike in natural image domains, documents present a rich, structured visual grammar: titles, paragraphs, lists, tables, figures, and annotations follow hierarchical spatial rules that convey meaning beyond the text itself. Pretraining must therefore go beyond conventional image–text alignment and actively model the interaction between spatial layout and semantics. Below, we categorize pretraining approaches into five broad paradigms as summarized in Table 2.2, showing their underlying principles, representative models, and layout-specific implications.

Masked Language Modeling (MLM) with Layout-Aware Embeddings: Masked Language Modeling, introduced by BERT [60], masks a subset of tokens and trains the model to predict them from surrounding context. In Document AI, these tokens are augmented with layout embeddings—absolute or relative 2D coordinates representing their positions on the page. This enables the model to disambiguate text meaning using spatial cues (e.g., "Total" in the bottom right corner is likely a sum field). While extremely effective for natural language understanding tasks such as question answering, sentiment classification, and semantic similarity, *BERT is agnostic to where words appear on a page*. For documents, this is a critical shortcoming, as layout often carries meaning that text alone cannot convey. RoBERTa [167] refined the same idea by optimizing the pretraining process: removing next-sentence prediction, increasing the amount of training data, and using dynamic masking. This resulted in stronger text representations but still *without any notion of spatial structure*.

The LayoutLM series was the first to adapt BERT's principles to the 2D nature of documents, treating spatial coordinates as integral to token representation. *LayoutLMv1* [274] extended the BERT architecture by adding 2D positional embeddings (x–y coordinates, width, and height) to each text token, sourced from OCR outputs. This allowed the model to consider not just what the word says, but also where it appears. For example, in a form, the word "Total" in the bottom-right corner can be disambiguated from "Total" in a paragraph heading purely through positional information. *LayoutLMv2* [273] went a step further by incorporating visual embeddings extracted from the document image itself using a ResNet backbone. Now, the model jointly attends to text, its position, and the underlying visual cues—allowing it to pick up on structural features like table borders or font styles. *LayoutLMv3* [104] unified text and image token masking into a single transformer backbone. Instead of treating visual and textual features as separate streams, it masked both in the same embedding space, enabling richer cross-modal interactions. This not only improves understanding but also supports gen-

Table 2.2: Pretraining paradigms for layout-aware Document AI: core ideas, representative models, data requirements, and typical applications.

Paradigm	Core Idea	Models	Requirements	Applications
MLM (Layout-Aware)	Mask tokens and predict with spatial embeddings to capture text–layout interactions.	BERT; RoBERTa; LayoutLMv1/ v2/ v3; LayoutXLM	Large OCR-processed corpora with layout metadata.	Form understanding; key-information extraction; DocVQA.
MIM	Mask image patches and reconstruct them to learn visual layout structure.	ViT; BEiT; MAE; DiT; DocFormer	Large image-only document datasets.	Layout classification; OCR-free DocVQA; table structure recognition.
Contrastive Multimodal	Align visual and text embeddings by matching paired samples.	CLIP; LiLT; GlobalDoc	Large paired image–text datasets.	Document retrieval; multimodal search; captioning.
Seq2Seq Multimodal	Encode visual+text inputs and autoregressively decode structured text.	T5; Pix2Struct; Donut; Dessurt; UReader	High-quality paired image–markup/ QA datasets.	End-to-end DocVQA; chart-to-text; image-to-markup conversion.
Entity-Level Supervision	Pretrain with labeled fields/ entities to bind layout and semantics.	StrucTexT; FormNet; BROS; Doc2Graph	Domain-specific labeled datasets.	Receipts; ID parsing; domain-specific forms.

erative capabilities in document contexts. *LayoutXLM* [275] extended LayoutLMv1's ideas into the multilingual domain, pretraining on diverse scripts and page layouts. This made it possible to transfer knowledge across languages without retraining from scratch for each new one—a critical capability for global enterprise document processing. Adding layout embeddings effectively elevates positional information from an auxiliary cue to a semantic signal. This is in lieu to the fact that in real-world documents, **layout is a powerful disambiguator**: (i) Identically worded fields in different parts of a page mean different things. (ii) Tables, headers, and form structures encode relationships not obvious in raw text. (iii) Spatial grouping helps infer reading order in multi-column layouts. Without layout-aware modeling, even the most powerful text encoders are blind to these cues, leading to suboptimal performance in document AI

tasks.

Strengths: By combining token semantics with spatial embeddings, they excel in (i) *structured text extraction*, accurately retrieving fields from forms, invoices, and identity documents. (ii) Their spatial reasoning capabilities enable robust *form understanding*, such as linking distant labels to corresponding values, and have proven highly effective in *Document Visual Question Answering (DocVQA)*, where they can locate and contextualize relevant regions for precise answers. (iii) Moreover, when OCR output is reliable, these models exhibit strong *cross-domain transfer*, generalizing well across diverse document types without significant fine-tuning.

Limitations: However, the architecture retains certain limitations. (i) A *text-centric bias* means that performance degrades when the document's meaning is conveyed primarily through visual elements such as diagrams, charts, or pictorial layouts. (ii) The *heavy OCR dependency* introduces a vulnerability: any transcription errors—such as missing tokens, misaligned bounding boxes, or segmentation artifacts—can significantly impair downstream reasoning, especially in low-quality, noisy, or handwritten documents. (iii) Finally, despite improvements in LayoutLMv2 and LayoutLMv3, the models' reasoning remains anchored in *token-level processing*, limiting their ability to capture purely visual patterns. This contrasts with OCR-free approaches such as Donut and Pix2Struct [129, 140], which directly model raw pixels and textual sequences end-to-end, bypassing the bottlenecks of explicit OCR.

Masked Image Modeling (MIM) for Visual Layout Comprehension: Masked Image Modeling (MIM) extends the principles of Masked Language Modeling into the visual domain, aiming to learn general-purpose visual representations by reconstructing missing parts of an image. In the context of natural images, methods such as BEiT [16] and MAE [93] have demonstrated that masking large portions of an input and predicting the missing visual content fosters powerful visual encoders. Unlike supervised image classification, MIM does not require explicit labels—making it well-suited for large-scale pretraining across diverse domains. From a foundational perspective, the Vision Transformer (ViT)[64] introduced a pure transformer-based architecture for vision, splitting an image into non-overlapping patches and processing them as a sequence of tokens. BEiT[16] enhanced this idea by introducing discrete visual tokens derived from a separate tokenizer (e.g., a dVAE) and formulating MIM as a token-prediction task. MAE [93] simplified the approach by directly reconstructing pixel values from a sparse set of visible patches, demonstrating that highly masked inputs (up to 75%) can still lead to strong learned representations.

In Document AI, MIM offers unique advantages over text-centric pretraining objectives. By operating directly on the page image, it captures holistic layout patterns such as the geometric regularity of table grids, the flow of multi-column text, the alignment of headers and paragraphs, and the spatial co-occurrence of figures with captions. Critically, this process bypasses the need for OCR, making MIM-based models inherently robust to handwriting variations, low-quality scans, and documents in scripts with limited OCR support. Document-specific adaptations include DiT [147], which applies the ViT+MIM paradigm to millions of rendered documents, enabling the

model to learn a rich "visual grammar" of page structures. DocFormer [6] and StructTextv2 [284] extends this idea by integrating MIM with Masked Language Modeling (MLM) in a multimodal transformer, jointly reconstructing both image patches and masked text tokens. This hybrid pretraining strategy allows the model to associate visual layout with textual semantics, thereby enhancing its cross-modal reasoning capabilities.

Strengths: Firstly, (i) MIM enables *OCR-free representation learning*, meaning that the model does not rely on any text extraction process and can therefore operate effectively on noisy, handwritten, or low-resource-script documents without degradation in performance due to OCR errors. (ii) Secondly, MIM promotes *global layout awareness* by forcing the model to reconstruct masked patches, which encourages the learning of spatial composition, page structure, and formatting conventions across a broad variety of document types. This structural understanding is essential for downstream tasks that depend on recognizing document organization rather than just textual content as shown in methods like Text-DIAE [231].

Limitations: (i) A primary limitation is the *lack of token-level semantics*, as purely visual pretraining focuses on spatial and visual patterns without inherently capturing the fine-grained meaning of textual content. This can lead to suboptimal performance on tasks requiring precise semantic interpretation unless MIM is combined with text-based objectives. (ii) Additionally, MIM exhibits *domain sensitivity*, where models trained on one document distribution may misinterpret or over-generalize layout patterns when applied to domains with markedly different formatting rules, such as transitioning from scientific articles to retail receipts. This issue often necessitates domain-specific fine-tuning to achieve optimal performance.

In essence, MIM equips document models with a strong layout-centric inductive bias that complements language-based pretraining objectives, making it a powerful tool in both OCR-free and hybrid pretraining strategies. However, its full potential is typically realized when paired with textual grounding, ensuring that learned visual structures are semantically meaningful rather than being limited to purely geometric similarity.

Multimodal Contrastive Objectives for Layout–Text Alignment: Multimodal contrastive learning has emerged as a powerful paradigm for aligning representations from different modalities into a shared embedding space. Inspired by approaches such as CLIP [202] in the vision–language domain, these objectives train models to maximize similarity between matching image–text pairs while minimizing similarity with mismatched pairs. In the context of documents, the modalities of interest extend beyond images and text to include layout structure, enabling models to reason jointly over spatial, visual, and semantic information. At its core, the contrastive loss function encourages embeddings of paired modalities—such as a document image and its corresponding OCR text—to be close in the latent space, while embeddings of unrelated pairs are pushed apart. This approach as in VLCDoc and GlobalDoc [12, 11] has proven especially effective for tasks requiring cross-modal retrieval, zero-shot classification, and representation transfer, as it does not rely on task-specific labels and instead exploits naturally co-occurring multimodal data.

From a foundational perspective, CLIP [202] demonstrated that large-scale pretraining on image–caption pairs could yield models capable of zero-shot recognition across diverse visual domains. ALIGN [115] extended this idea to even larger datasets, highlighting the scalability of contrastive objectives. These methods rely on strong encoders for each modality, typically a vision transformer for images and a transformer-based language model for text, trained jointly to produce compatible embeddings. In Document AI, multimodal contrastive objectives have been adapted to incorporate layout-specific cues. For example, LayoutCLIP-style adaptations encode not only the document image and text but also spatial embeddings derived from the positions of tokens. This enriches the contrastive pairing by ensuring that matched modalities reflect not just visual and linguistic similarity, but also geometric correspondence. Models such as LiLT [252] and UDOP [238] leverage this principle by aligning vision, language, and layout embeddings simultaneously, improving transferability across document tasks and languages.

Strengths: (i) First, contrastive objectives promote *cross-modal alignment*, enabling models to retrieve relevant text from a visual query or locate the appropriate document region from a textual query with high accuracy. (ii) Second, they are *label-efficient*, as they can be trained using weakly paired multimodal data—such as OCR text automatically extracted from large document collections—without the need for expensive manual annotations. (iii) Third, the learned joint embedding space facilitates *zero-shot and few-shot transfer*, where models pretrained on generic multimodal data can adapt quickly to new document understanding tasks without retraining from scratch.

Limitations: (i) One limitation is the *weak grounding of fine-grained elements*. Contrastive learning typically aligns entire modality representations (e.g., the whole page and its text) but may fail to capture fine-grained correspondences between specific words, regions, or layout components unless explicitly modeled. (ii) Another limitation is the *representation bias towards dominant modalities*, where high-capacity visual or textual encoders may dominate the shared space, overshadowing subtle but critical layout cues. (iii) Finally, contrastive objectives are sensitive to *noise in pairings*, meaning that OCR errors, layout parsing mistakes, or imperfect visual crops can degrade the quality of cross-modal alignment.

In summary, multimodal contrastive pretraining offers an elegant and scalable approach to aligning vision, language, and layout information in document AI. By situating these modalities within a unified embedding space, it enables strong transfer learning and retrieval capabilities across diverse document types. However, to fully harness its potential for layout-aware tasks, it is essential to incorporate finer-grained alignment strategies that explicitly model relationships between textual tokens, visual regions, and spatial structures as shown in TILT [199].

Sequence-to-Sequence Generation for Structured Outputs: Sequence-to-sequence (seq2seq) modeling reframes document understanding as a direct generation problem, where the model produces a structured output sequence, such as JSON, key–value pairs, or natural language conditioned on an input document image. Unlike encoder-only architectures that focus on representation learning, seq2seq models employ an

encoder–decoder structure, enabling them to generate arbitrary output formats while conditioning on multimodal inputs. This approach has gained prominence in OCR-free Document AI, as it eliminates intermediate tokenization steps and directly learns to map from raw pixels to structured textual outputs.

In the vision–language domain, pioneering architectures such as T5 [205] and BART [142] established the viability of large-scale text-to-text generation by pretraining on denoising objectives and fine-tuning for downstream tasks. Vision–language extensions, such as OFA [255], expanded this paradigm by unifying multiple tasks—captioning, visual question answering, and grounding—under a single generative interface. Document-specific adaptations have applied seq2seq principles directly to page images. *Donut* [129] dispenses with OCR entirely, using a vision transformer encoder and a transformer decoder to translate document images directly into structured text sequences (e.g., JSON-formatted entity extraction results). This allows the model to learn holistic mappings between visual layouts and target schemas, handling complex cases like receipts, forms, or invoices without explicit bounding-box annotations. Similarly, *Pix2Struct* [140] tokenizes an input image into a sequence of patches and processes them with a transformer encoder, while a text decoder generates answers to visual questions or structured descriptions of the layout. These models excel in tasks where both structure and semantics must be preserved, such as table-to-markdown conversion [170, **?**], chart description [184], or rich form extraction [137]. Beyond Pix2Struct, several recent models have also extended the seq2seq paradigm in Document AI: *Nougat* [28] and *KOSMOS-2.5* [173] adapt LLM-based architectures for end-to-end document-to-markdown conversion, producing spatially aware formatted text directly from PDFs or images. On the other hand, *DREAM* [154] targets holistic document reconstruction, generating sequences that jointly encode logical structure (paragraphs, tables, formulas) and physical layout. Document generation approaches like *DocSynthv2* [25], also a core chapter of this thesis, builds on this seq2seq paradigm with incorporation of text style and position attributes with aligned layout properties.

Strengths: (i) First, seq2seq architectures enable *end-to-end learning*, mapping directly from raw document pixels to final structured outputs without reliance on intermediate OCR or rule-based post-processing. (ii) Second, they are inherently *schema-flexible*, allowing the same model to produce outputs in arbitrary formats—natural language, structured JSON, or markup—depending on the task specification. (iii)Third, their generative nature supports *multi-task unification*, as the same model can be prompted or fine-tuned to handle diverse document understanding tasks within a single architecture.

Limitations: (i) A key limitation is *generation faithfulness*—without explicit grounding mechanisms, seq2seq models may hallucinate values or introduce subtle inconsistencies in structured outputs, especially under distribution shifts. (ii) Second, they are typically *computationally intensive* at inference time compared to encoder-only models, as generation is autoregressive and often requires beam search or other decoding strategies. (iii) Finally, while these models bypass OCR, they may still *struggle with fine-grained localization*—for example, distinguishing multiple identical fields in different

parts of a page—unless aided by explicit spatial cues or pointer-based mechanisms.

In summary, sequence-to-sequence generation offers a flexible, unified, and OCR-free approach to document understanding, capable of directly producing structured outputs from visual inputs. It represents a promising direction for layout-aware pretraining, particularly when coupled with grounding techniques to ensure output faithfulness and spatial precision. Models like Donut and Pix2Struct exemplify the potential of this paradigm, bridging the gap between visual perception and structured reasoning in Document AI.

Field-Level or Entity-Level Supervision: Field-level or entity-level supervision departs from the fully self-supervised pretraining paradigm by introducing explicit, semantically meaningful labels during pretraining or intermediate finetuning. Instead of only asking the model to predict missing tokens or image patches, we guide it toward recognizing and aligning structured components of a document such as key-value fields, table cells, or named entities, directly during representation learning. The rationale is straightforward: while generic pretraining can yield broad layout-aware embeddings, some applications demand precise semantic grounding. For example, in an invoice, the system must understand that a number in the bottom-right cell labeled "Total" is semantically different from a number under "Quantity" in a table row. Purely self-supervised objectives may learn some of these associations implicitly, but entity supervision makes the link explicit from the start.

A canonical example is StrucTexT[157], which uses structure-aware masking—masking both the text content and the positional cell grid—and supervising predictions at the cell level. This forces the model to jointly reason about what the content is and where it fits in the table's logical structure. Similarly, DocBank[151] provides large-scale annotations for document elements (titles, lists, tables, figures), enabling pretraining to capture document "part-of" hierarchies beyond token-level sequences. In the form understanding space, FormNet[137] leverages graph neural networks (GNNs) over form fields, treating each field as a node and explicitly encoding relationships between fields and their values. This formalizes the document not just as text or image but as a typed, connected graph, an approach well-suited to entity supervision. This similar approach has also been adapted to Doc2Graph [78, 183] and GeoContrastNet [22] where structured label information incorporated into GNN's gives a huge boost in model performance. Similarly, LayoutLMv3[104] and DocFormerv2 [7] incorporate entity annotations during intermediate finetuning (e.g., FUNSD [114], SROIE [106]), showing that even without entity-aware pretraining, injecting structured labels mid-way can significantly enhance layout–text grounding.

Strengths: (i) A primary advantage of field-level or entity-level supervision lies in its *direct alignment with downstream objectives*. By exposing the model to structured entities during pretraining, the learned representation space is naturally shaped toward the semantic granularity required for real-world extraction tasks. (ii) Furthermore, entities serve as *semantic anchors*, binding specific visual regions to well-defined meanings and thereby strengthening the association between spatial layout and textual semantics. This explicit binding is particularly advantageous in scenarios where structural

interpretation is essential, such as linking form fields to their corresponding values. (iii) In addition, the paradigm demonstrates *strong compatibility with structured output requirements*, as entity supervision inherently aligns with tasks that operate over predefined schemas, including receipt parsing, form understanding, and identity document processing.

Limitations: (i) Despite its advantages, entity-level supervision presents several notable constraints. First, the *annotation cost is substantial*, as creating high-quality entity labels demands domain expertise and significant manual effort. (ii) Secondly, the approach often exhibits *reduced generality*, with heavily supervised pretraining predisposing the model toward domain-specific patterns, thereby limiting its zero-shot transferability to novel document types. (iii) Thirdly, there is an *inherent schema rigidity*: models trained with a fixed set of entity definitions may require extensive retraining or adaptation when faced with tasks that diverge from the original annotation schema.

From a broader perspective, field-level supervision represents a deliberate trade-off, sacrificing some general-purpose flexibility in exchange for heightened precision in domain-critical applications. This trade-off is particularly justified in high-stakes domains such as finance, healthcare, and scientific publishing, where the cost of semantic error outweighs the need for broad generalization. A key research challenge lies in devising strategies to integrate such task-specific supervision with general-purpose multimodal pretraining objectives—including Masked Language Modeling, Masked Image Modeling, and contrastive learning—in order to unify universal layout understanding with domain-specific semantic grounding within a single foundation model.

Overall Takeaway: The pretraining paradigms surveyed in this section highlight the multifaceted nature of layout-aware learning in Document AI. While MLM with layout embeddings captures spatially informed text semantics, MIM builds global layout awareness without dependence on OCR. Multimodal contrastive learning strengthens alignment across visual, textual, and structural modalities, while seq2seq objectives enable direct generation of structured outputs. Finally, field-level supervision offers high-precision semantic grounding for domain-specific tasks. Collectively, these approaches define a continuum between general-purpose representation learning and domain-optimized modeling. The key research frontier lies in reconciling these approaches i.e. *designing foundation models that can seamlessly navigate between OCR-free visual reasoning, layout grammar induction, and schema-constrained entity extraction*, producing robust and adaptable systems capable of handling the full diversity of real-world documents.

2.4 Layout-Aware Document Generation

Document generation is emerging as a distinct yet highly interdependent branch of Document AI, aiming not merely to extract or represent layout information, but to synthesize new document instances that adhere to realistic structural and semantic con-

straints. Unlike generic text-to-image synthesis, layout-aware document generation operates within a **multi-modal, grammar-constrained space**, where the interplay between textual content, visual appearance, and spatial arrangement defines the fidelity and usability of the generated output.

The evolution of the document generation domain can be broadly traced to two complementary research directions: (i) **Layout Generation**, which focuses on arranging elements (e.g., text boxes, images, tables, graphics) on a page or canvas while respecting design constraints and semantic intent; and (ii) **Vector Graphic Generation**, which models the precise geometric primitives (e.g., strokes, shapes, curves) that constitute the visual elements themselves. While the latter controls fine-grained visual appearance, the former determines global structure, spatial relationships, and reading flow. In practice, document generation pipelines often rely on a layout generator as the structural backbone, upon which visual or textual content is rendered. This subsection reviews the progression of *layout generation* approaches—from early rule-driven systems to recent diffusion-based methods—highlighting their applicability to document AI.

Early Rule-Based Layout Synthesis: Initial research on automatic layout generation embedded human-crafted design heuristics directly into energy functions or constraint-solving frameworks [187, 192]. These systems operated by optimizing aesthetic and functional criteria (e.g., alignment, spacing, balance) but lacked the capacity to learn from data. As such, they were not robust enough to novel design styles and could not capture subtle correlations between element categories and their preferred spatial arrangements.

GAN and VAE-based Generative Models: The advent of deep generative models introduced data-driven layout synthesis. LayoutGAN [146] and LayoutVAE [120] pioneered the use of GANs and VAEs to generate scene and graphic layouts, learning distributions over element positions and sizes from large design corpora. NDN [139] modeled layouts as graphs of relative spatial relationships and applied a graph neural network–conditioned VAE. READ [73] used heuristics to derive relational structures between elements and trained a Recursive Neural Network (RNN) [68, 228] within a VAE framework to capture hierarchical layout organization. CanvasVAE [277] extended this to vector graphic documents, predicting structured canvas–element representations. Self-attention–driven VAEs, such as VTN [8] improved diversity and perceptual realism by better capturing global dependencies across elements.

Transformer-based Sequence Models: Inspired by the success of autoregressive sequence modeling in NLP, LayoutTransformer [86] and BLT [133] serialized layouts into discrete token sequences (category, coordinates, dimensions) and leveraged transformer architectures for structure-aware generation. This framing allowed flexible conditioning on partial layouts and efficient modeling of long-range spatial dependencies. Conditional layout generators such as LayoutNet [293], TextLogo3K [261], and ICVT [31] incorporated auxiliary attributes—ranging from style vectors to semantic class constraints—enabling controllable generation across domains (e.g., advertising, logos, publication layouts). LayoutGAN++ [126] combined transformer generators with learned

discriminators, further enhancing layout realism.

Diffusion-based Layout Generation: More recently, diffusion models have emerged as a state-of-the-art paradigm for structured generation. LayoutDM [108] framed layout synthesis as a denoising process over discrete spatial tokens, yielding improved diversity and structural coherence compared to GAN/transformer baselines. LACE [37] introduced constraint-guided diffusion, where user-defined spatial constraints (e.g., fixed regions, alignment masks) guide the denoising trajectory, supporting interactive and iterative layout editing. Sequence-domain diffusion models [35] operate over serialized element tokens, blending the discrete control of transformers with the stochastic refinement of diffusion. To benchmark these systems, the Document Earth Mover's Distance (Doc-EMD) [97] metric was proposed, capturing both spatial and categorical similarity between generated and reference layouts.

Strengths: (i) Modern generative models, particularly transformers and diffusion approaches, learn directly from large-scale layout corpora, capturing diverse spatial patterns beyond human-defined rules. (ii) Diffusion-based methods enable multi-modal control (masking, constraints) without retraining, and can sample multiple valid layouts for the same condition.

Limitations: (i) GAN/transformer models may suffer from mode collapse or limited diversity without architectural refinements. (ii) Diffusion models incur higher inference latency due to iterative denoising, and their latent factors are less interpretable without explicit structuring. (iii) Evaluation remains challenging—metrics such as Doc-EMD focus on geometry and category matching but do not fully reflect human aesthetic judgment or functional suitability.

Towards Synthetic Document Generation: Once coherent layouts have been generated, the natural progression is to synthesize complete document images, including textual content, visual elements, and formatting such that the generated output adheres closely to a given structural blueprint. This step bridges the gap between abstract spatial planning and fully realized document creation, enabling applications such as dataset augmentation, template design, and end-to-end automated publishing. One of the earliest works in this space, *DocSynth* [27], built on top of [288] demonstrated layout-guided document image synthesis by conditioning a generative pipeline on spatial layouts defined by bounding boxes and semantic labels. The model produces realistic document pages where the spatial organization matches the provided template while still generating varied, plausible content. This approach has been particularly useful for augmenting training datasets in layout analysis tasks [71], improving both diversity and robustness of downstream models. *SynthTiger* [280] was another approach that mainly OCR-driven for synthetic document generation, which was adapted in the Donut [129] foundation model pipeline.

Vector Graphic Document Generation: Unlike pixel-based rendering, vector formats such as Scalable Vector Graphics (SVG) enable resolution-independent rendering, free of rasterization artifacts, and facilitate post-generation editing due to their explicit geometric structure. This property is especially attractive for document generation, where

precision in element placement and typography is crucial. However, modeling documents in a vector format poses significantly greater complexity compared to traditional stroke- or path-level vector graphics generation [87, 111, 243]. In document settings, each element may possess multi-modal attributes such as embedded text, images, and associated metadata demanding a richer representation that integrates both geometric and semantic constraints. *FlexDM* [109] considered such multi-modal features to go beyond layout generation for intelligent graphic design assistance and developed the first unified multi-task model in this domain.

Early work in this space, such as *CanvasVAE* [277], tackled unconditional document-level vector graphic generation, directly producing structured representations of canvases and their constituent elements. While effective in capturing global structure, CanvasVAE is not inherently multi-task and cannot directly handle specific conditional design operations such as targeted element filling or attribute editing. *Doc2PPT* [72] explored a related setting in presentation design, generating slide layouts from longer, multi-modal documents. However, this approach is framed primarily as a summarization and transformation task, rather than a generative completion model, and cannot infer missing components from incomplete inputs. Building on this drawback, *DocSynthv2* [25] adopts an autoregressive transformer architecture that models both layout and textual content as unified sequences. This formulation enables context-aware generation where the model can complete partially specified layouts with plausible visual and textual elements while ensuring consistency between semantic content and spatial arrangement. It further eliminates the need for intermediate rasterization during synthesis, producing high-resolution, semantically coherent document pages directly from structural descriptions. *StarVector* [216] pursued a similar autoregressive approach to develop a foundational multimodal LLM for SVG generation. It processes both images and text instructions to produce compilable SVG code, leveraging SVG primitives to accurately represent vector graphics. Recently, *BigDocs* [215] built a large-scale structured Document Understanding dataset for devising tasks like multimodal code generation, reasoning over graphical user interfaces (GUI), websites and documents and generating code from images.

2.5 Conclusion and Open Challenges

The trajectory of Document AI, from early rule-based systems to today's multimodal foundation models, reflects a field in constant re-invention, shaped by breakthroughs in representation learning, scalable pretraining, and generative modeling. The progression outlined in Figure 2.1 underscores how each stage—interpretation, representation, and generation—has expanded the boundaries of what machines can understand and create from structured and unstructured documents. Layout-aware pretraining paradigms such as MLM, MIM, and contrastive learning have delivered powerful semantic–spatial representations, while hybrid architectures now enable cross-modal reasoning that was previously unattainable. In parallel, generative approaches are transforming document creation, from controllable layout synthesis to full docu-

ment rendering, opening the scopes for new opportunities in design automation, accessibility, and human–AI co-creation. Yet, despite these advances, several open challenges remain at the core of the field's future progress:

Robustness to Imperfect Inputs: Heavy dependence on OCR pipelines or specific rendering formats still makes many models vulnerable to noise, low-resolution scans, handwriting variability, and non-standard layouts.

Generalization Across Domains and Languages: Current models often degrade significantly when applied to unseen domains, multilingual settings, or cultural layout variations, highlighting the need for truly universal document encoders.

Integration of Rich Visual Semantics: While layout-aware models excel at token-level reasoning, they remain limited in understanding diagrams, charts, and dense visual elements without explicit textual grounding.

Faithful and Controllable Generation: Generative document models must balance creative flexibility with factual grounding and adherence to user-specified constraints, a challenge amplified in high-stakes domains such as healthcare or finance.

Explainability and Trustworthiness: As models take on more autonomous reasoning and generation tasks, the ability to attribute outputs to specific content and layout cues will be critical for building trust, meeting compliance requirements, and enabling human–AI collaboration.

Unified Pretraining for Multi-Objective Learning: Integrating interpretation, representation, and generation capabilities into a single, general-purpose foundation model without sacrificing task-specific precision, still remains an unsolved challenge.

Addressing these gaps will require not only architectural innovation but also richer pretraining corpora, unified multimodal evaluation benchmarks, and principled approaches to model transparency. In this thesis, we address these challenges by exploring how the **language of layouts** can serve as a unifying representation, enabling systems that do not just read or render documents, but reason about them in ways that align with human understanding.

Part I

Interpretation

Chapter 3

Beyond Bounding Boxes: Fine-Grained Document Segmentation

The details are not the details. They make the design.
– Charles Eames

Document Layout Analysis (DLA) is a fundamental task in Document Understanding pipeline, facilitating the automated extraction of structural elements such as text blocks, tables, figures, and lists. Existing Document Object Detection (DOD) methods rely on bounding boxes to localize these elements but often lack the precision needed for parsing overlapping objects and hierarchical structures in complex layouts. This chapter introduces instance-level segmentation for DLA, assigning pixel-wise masks to each element for more accurate localization. We outline the transition from Object Detection to Instance Segmentation with exhaustive experimentation highlighting its impact on layout parsing.

3.1 Introduction

With the rapid proliferation of digital documents across industries, the need for intelligent and automated methods for document analysis has become paramount. Manual processing is no longer scalable due to the sheer volume of data, driving modern research in document artificial intelligence (DocAI) to focus on AI-driven approaches for extracting, structuring, and retrieving information from complex document layouts [46]. Visually-rich Document Understanding (VrDU) has attracted increasing interest in recent years, encompassing tasks such as document image classification (DIC) [122,

Figure 3.1: **Comparison of Object Detection in Natural Scenes vs. Document Object Detection.** The left image shows object detection in natural scenes, identifying objects like people and buses using visual cues. The right image depicts DOD, segmenting structured elements like tables and charts. Unlike natural scenes, document layouts require hierarchical understanding, making instance segmentation essential for precise layout parsing and information extraction.

91, 112, 165], key information extraction (KIE) [158, 172, 226, 113, 233], document layout analysis (DLA) [23, 197, 47, 291, 26, 13, 174, 24], and document visual question answering (VQA) [180, 62, 179, 241]. DLA involves decomposing document images into semantically meaningful regions such as text blocks, tables, figures, and titles. Unlike DLA, which aims to segment and analyze the entire document structure, document object detection (DOD) focuses specifically on localizing and classifying individual document elements using bounding boxes to enable downstream tasks like information extraction, document parsing, and content retrieval efficiently.

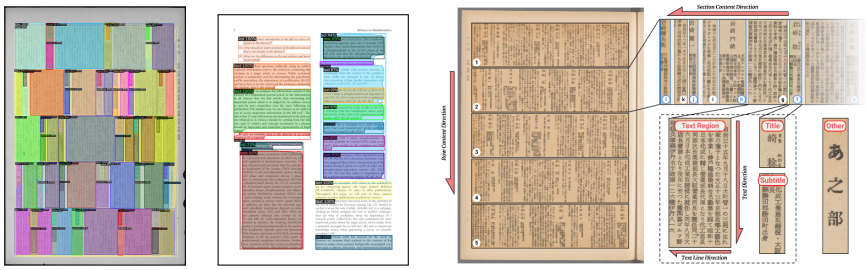
Current state-of-the-art (SOTA) DU models [104, 82, 274, 273] typically rely on modern optical character recognition (OCR) engines to extract text and combine it with spatial features to predict page layout and structure. However, these multimodal architectures face several limitations: (1) they depend heavily on Large Language Models (LLMs) [289] pretrained on millions of samples, prioritizing OCR text quality over visual features and document structure; (2) they can be computationally expensive due to the need to process and fuse information from multiple modalities; and (3) they may underperform in domains with poor OCR results or low-resource languages. To address these challenges, DLA serves as a critical preliminary step in document processing workflows [23, 47], enhancing downstream DU tasks such as DIC, KIE, and VQA. By imparting logical layout structure beyond the geometric layout provided by OCR [89], DLA enables more accurate content extraction and interpretation. A recent DU competition [247] has highlighted the need to bridge the gap between DLA and DocVQA [181] by introducing layout-navigating or multi-region questions, further emphasizing the importance of robust layout understanding.

Early DLA methods primarily relied on rule-based techniques [38, 245, 70, 223], which used heuristic rules, geometric layouts, and structural templates to segment and classify document components. While effective for well-defined document types, these methods lacked adaptability to diverse layouts and struggled with complex struc-

tures, overlapping elements, and multi-column formats, making them unreliable for real-world applications. The introduction of deep learning-based object detection architectures for natural scene images [160, 210, 207, 145] allowed models to learn hierarchical features and improve detection accuracy of document objects using bounding boxes [220, 190]. While both tasks share similarities, document images introduce unique challenges due to structural differences. One key distinction lies in the **large domain gap**—natural images contain diverse backgrounds, lighting conditions, and 3D environments, making object context crucial for detection. In contrast, documents exist in a 2D structured space, where elements like tables, figures, and text blocks are arranged logically rather than spatially, requiring a deeper understanding of layout relationships rather than just visual features. Additionally, document layouts exhibit **high inter-class variance**, not in terms of visual appearance but in their structural organization. Unlike natural objects, where shape and texture differentiate categories, document elements often share similar textual properties, making it harder for models to distinguish between tables, paragraphs, or figures without contextual awareness.

Another fundamental challenge is that documents contain **highly interdependent elements**, whereas natural scene objects are more modular and can be recognized independently. For example, a title only functions as a title if positioned correctly within the document hierarchy, and a figure caption depends on its linked figure. Standard object detection struggles with this hierarchical structure, requiring models that capture spatial relationships rather than treating document elements as isolated entities. Furthermore, while natural image object detection has benefited from large-scale annotated datasets like MS-COCO [162] and ImageNet [59], document layout **datasets remain scarce** and domain-specific, limiting generalization across different formats, languages, and structures. Given these challenges, bounding-box-based object detection is insufficient for fine-grained document parsing, necessitating instance segmentation, which provides pixel-level object boundaries to resolve overlapping regions (e.g., figures inside tables), preserves hierarchical dependencies, and enhances structural layout understanding. Thus, this chapter proposes moving beyond object detection and adopting instance segmentation as a more precise and effective approach for DLA.

The main contributions of this Chapter are: (1) We introduce an instance segmentation approach for DLA, transitioning from bounding-box-based object detection to pixel-level segmentation for a more precise understanding of complex document structures. (2) We establish strong baselines using our proposed Mask R-CNN-based model on two benchmark datasets, PubLayNet [291] and the Historical Japanese Dataset [224], demonstrating its effectiveness in segmenting diverse document layouts. (3) Our framework is evaluated against state-of-the-art object detection methods, highlighting its advantages in handling overlapping objects and hierarchical structures in documents. (4) We conduct ablation studies to analyze the impact of instance segmentation on document object detection, demonstrating its role in enhancing layout parsing, content extraction, and document intelligence applications.



(a) (b) (c)

Figure 3.2: **Illustration of Document Layout Challenges**: (a) and (b) show overlapping object categories in HJDataset [224] and PubLayNet [291], where bounding-box-based methods struggle. (c) highlights the hierarchical document structure in historical Japanese texts [224], demonstrating the need for instance segmentation to capture layout relationships accurately.

3.2 Related Work

Automatic information extraction from digital documents requires an understanding of spatial layouts, involving the detection of key elements such as tables, titles, figures, and text blocks. Several approaches have been proposed, evolving from rule-based segmentation to deep learning-based object detection and finally to instance segmentation for more fine-grained document understanding.

Early **rule-based methods** relied on heuristic techniques such as connected component grouping [188], white-space analysis [206], and Voronoi-based segmentation [132] to segment document elements. These methods were effective for structured layouts but struggled with complex multi-column layouts, handwritten content, and overlapping objects. Several improvements were made using Delaunay triangulation [271] and spatial autocorrelation [119], but these approaches still lacked generalization across diverse document types. **Machine learning-based** pre-deep-learning methods [175, 10] attempted to improve segmentation by using Multi-Layer Perceptrons (MLPs) and Support Vector Machines (SVMs) to classify document regions. However, these approaches depended on handcrafted features and assumed fixed layout structures, limiting their effectiveness on documents with varying formats. The inability to model complex structural relationships and hierarchical dependencies highlighted the need for data-driven deep learning approaches.

The introduction of **deep learning** transformed DLA, allowing models to learn feature representations directly from data. Faster R-CNN [210] became a standard two-stage object detector, widely adopted for document object detection [220]. Other two-stage methods such as DeepDeSRT [220] demonstrated improvements in table detection and structure recognition, enabling OCR-free processing of document layouts. One-stage detectors like YOLO [208] and SSD [166] were explored for faster document object

detection, but struggled with small, dense objects and layout complexities. Additionally, Fully Convolutional Networks (FCNNs) [190] were applied for pixel-wise segmentation of text and figures in historical documents, improving layout parsing. Graph-based models [211, 153] and transformer-based architectures like LayoutLM [274] further enhanced document understanding by integrating spatial relationships and textual content. However, , bounding-box-based object detection suffered from two key limitations: (1) *Overlapping Objects* – Figures inside tables, multi-column text layouts, and nested objects are difficult to separate using only bounding boxes. (2) *Hierarchical Layout Understanding* – Many document elements have structural dependencies, such as section headings linking to body text, which bounding-box detectors fail to capture. These limitations motivated the shift towards instance segmentation for more precise document parsing.

Instance segmentation offers a promising approach for segmenting layouts by providing pixel-level masks, allowing for better differentiation of overlapping and structured elements. Mask R-CNN [95] introduced this approach by integrating object detection with segmentation masks, enabling more detailed layout parsing. Mask Scoring R-CNN [105] further refined this by incorporating confidence-based scoring, leading to more reliable segmentation outputs. The development of large-scale annotated datasets has supported progress in document instance segmentation. PubLayNet [291] provides bounding box and mask annotations for scientific documents, while HJDataset [224] extends this to historical manuscripts, including hierarchical structures and reading order information. However, existing research in DLA has primarily focused on bounding-box-based object detection, leaving the potential of instance segmentation largely unexplored. This chapter presents the first work applying instance segmentation to DLA, leveraging mask-level annotations from these datasets. By moving from bounding-box detection to pixel-wise segmentation, we aim to improve layout parsing, content extraction, and document structure understanding.

3.3 Instance-Level Segmentation Framework

This section presents our end-to-end instance-level segmentation model, inspired by Mask R-CNN [95] and Mask Scoring R-CNN [105]. Unlike traditional bounding-box-based object detection, our approach extends DLA to pixel-level segmentation, enabling precise localization of elements such as tables, figures, paragraphs, and titles. State-of-the-art object detectors like Faster R-CNN [210] and RetinaNet [161] have demonstrated strong performance in DOD. However, these models are limited by their reliance on coarse bounding-box annotations, making it difficult to distinguish overlapping and nested objects. To address this limitation, we introduce a novel instance segmentation framework that enhances fine-grained layout parsing.

Figure 3.3 illustrates an overview of our instance segmentation pipeline, which consists of four core modules: (1) Feature Extraction and Selection - Extracts multi-scale features from the document image. (2) Object Detection Head – Predicts bound-

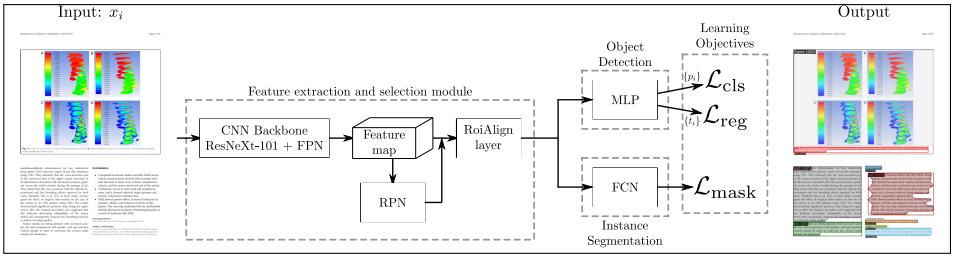


Figure 3.3: **Proposed Instance-Level Segmentation framework:** Given an input image of a document, the model predicts the different layout elements, with object detection on one head and instance-level segmentation on another head.

ing boxes and object categories. (3) Instance Segmentation Head – Generates pixel-wise masks for each detected element. (4) Learning Objectives – Defines loss functions for detection and segmentation.

3.3.1 Feature Extraction and Selection Module

Our model adopts a deep convolutional backbone for feature extraction, utilizing ResNeXt-101 [272], an advanced variant of ResNet, due to its strong multi-path representation capabilities. Given an input document image $\mathcal{I} \in \mathbb{R}^{H \times W \times C}$, the backbone extracts hierarchical feature maps at different levels $\mathcal{F}_l$, which are further refined using a Feature Pyramid Network (FPN) to capture both small and large-scale layout elements.

The FPN constructs a hierarchical multi-scale representation by iterating from coarse to fine resolution, refining feature maps through upsampling and lateral connections. The final feature representation is obtained as:

$$\mathcal{F} = f_{\text{conv3x3}}\Big(\text{Upsample}(P_{l+1}) + f_{\text{conv1x1}}(\mathcal{F}_l)\Big), \tag{3.1}$$

where $\mathcal{F}$ represents the final multi-scale feature map used for object detection and segmentation. The feature maps $\mathcal{F}_l$ are extracted from the ResNeXt-101 backbone at different levels l, while P_{l+1} denotes the feature map at a coarser scale. The operation $\text{Upsample}(P_{l+1})$ enhances spatial resolution, and $f_{\text{conv1x1}}(\mathcal{F}_l)$ applies a 1×1 convolution to refine features before merging. Finally, a 3×3 convolution $f_{\text{conv3x3}}(\cdot)$ is used to smooth and finalize the feature representation. This final representation $\mathcal{F}$, with a fixed dimension of 512 channels, serves as the input to the subsequent detection and segmentation heads, enabling robust DLA results.

3.3.2 Region Proposal Network & Region of Interest Alignment

The Region Proposal Network (RPN) is responsible for generating candidate object regions, each assigned an objectness score that determines whether the region likely contains a document element. From the set of generated proposals, the top 1,000 are selected using Non-Maximal Suppression (NMS) to filter out redundant and overlapping regions. Each proposal $\mathbf{R}_i$ is parameterized by its bounding box coordinates $\mathbf{b}_i = (x_{\min}, y_{\min}, x_{\max}, y_{\max})$ and an associated objectness score s_i.To ensure precise spatial alignment of these proposals, we employ RoIAlign [95], which eliminates the quantization errors present in RoIPooling. Given an input region proposal $\mathbf{R}_i$, RoIAlign applies bilinear interpolation to compute feature values at non-integer locations, preserving fine-grained spatial information. The interpolated feature value at a sampled point (x, y) inside the region is computed as:

$$\mathscr{F}_{\text{roi}}(x, y) = \sum_{i,j} w_{ij} \cdot \mathscr{F}(x_i, y_j), \tag{3.2}$$

where $\mathscr{F}(x_i, y_j)$ represents the extracted feature map at the nearest integer grid points (x_i, y_j), and w_{ij} are bilinear interpolation weights computed based on the relative distances from (x, y). This process ensures smooth feature extraction without loss of spatial precision. By applying RoIAlign to all selected region proposals, the model generates well-aligned feature representations that are crucial for accurate object detection and instance segmentation.

3.3.3 Detection and Segmentation Heads

The **object detection head** is responsible for recognizing and localizing document layout elements within each region of interest (RoI). It consists of a fully connected Multilayer Perceptron (MLP) that processes RoI features to generate two outputs: (i) a classification score for different document elements (e.g., tables, figures, paragraphs) and (ii) bounding box coordinates to refine object localization. The classification branch assigns a category label, while the bounding box regression branch adjusts coordinates to better fit detected objects.

The **instance segmentation head** extends the detection branch by providing pixel-level mask predictions for each identified document element. Unlike bounding boxes, segmentation masks precisely delineate object boundaries, allowing for improved handling of overlapping and structured elements. A Fully Convolutional Network (FCN) is employed to predict class-specific binary masks for each RoI. Given an RoI-aligned feature representation, the segmentation head produces an $m \times m$ mask per instance, where $m = 28$ in our implementation. This structured representation significantly enhances layout parsing by capturing the exact shape and spatial extent of different document entities.

3.3.4 Learning Objectives

To optimize both detection and segmentation, we employ a multi-task loss function that jointly minimizes classification, bounding box regression, and mask prediction errors.

The *detection loss* $\mathscr{L}_{\text{det}}$ consists of two components: classification loss $\mathscr{L}_{\text{cls}}$, which measures the accuracy of object category predictions, and bounding box regression loss $\mathscr{L}_{\text{reg}}$, which refines the predicted bounding box coordinates. The overall detection objective is formulated as:

$$\begin{aligned}\mathscr{L}_{\text{det}}\left(\{p_i\},\{b_i\}\right) = &\frac{1}{N_{\text{cls}}}\sum_i \mathscr{L}_{\text{cls}}\left(p_i, p_i^*\right) \\ &+ \frac{\lambda}{N_{\text{reg}}}\sum_i p_i^* \cdot \mathscr{L}_{\text{reg}}\left(b_i, b_i^*\right),\end{aligned} \tag{3.3}$$

where: p_i is the predicted probability that the i-th RoI belongs to a specific object category, while p_i^* is the ground-truth label. $b_i = (b_x, b_y, b_w, b_h)$ represents the predicted bounding box coordinates. b_i^* denotes the ground-truth bounding box coordinates. $\mathscr{L}_{\text{cls}}$ is the binary cross-entropy loss for object classification. $\mathscr{L}_{\text{reg}}$ is the smooth L_1 loss for bounding box refinement: N_{cls} and N_{reg} are normalization factors for classification and regression losses. λ is a balancing weight between classification and bounding box losses.

For *segmentation objective* $\mathscr{L}_{\text{mask}}$, we employ a per-pixel binary cross-entropy loss, ensuring that each predicted mask closely matches its corresponding ground truth mask:

$$\mathscr{L}_{\text{mask}} = -\frac{1}{m^2}\sum_{i,j}\left[y_{ij}^* \log y_{ij} + (1 - y_{ij}^*)\log(1 - y_{ij})\right], \tag{3.4}$$

where: y_{ij}^* is the ground truth mask value at pixel (i, j). y_{ij} is the predicted mask probability for the same pixel. $m \times m$ represents the spatial resolution of the predicted mask. The final objective function integrates both detection and segmentation losses to ensure a unified learning framework:

$$\mathscr{L} = \mathscr{L}_{\text{det}} + \alpha\mathscr{L}_{\text{mask}}. \tag{3.5}$$

where α is a weighting factor to balance the contributions of detection and segmentation tasks. This multi-task optimization enables the model to accurately classify and localize document objects while refining their spatial structure through instance-level segmentation.

3.4 Experimental Validation

To evaluate the effectiveness of the proposed approach, we conduct extensive experiments on benchmark datasets with diverse document structures. Our method is assessed against state-of-the-art models, demonstrating competitive performance in both object detection and instance segmentation. Furthermore, we perform detailed ablation studies to quantify the contribution of different architectural components. The implementation, trained models, and benchmark results are made publicly available at: `https://github.com/biswassanket/instasegdoc`.

3.4.1 Evaluation Metrics

For performance evaluation, we adopt the Intersection over Union (IoU) metric to measure the accuracy of object proposals. Following standard evaluation protocols, we compute mean Average Precision (mAP), which averages the Average Precision (AP) at IoU thresholds ranging from 0.5 to 0.95 in steps of 0.05. This evaluation methodology aligns with the standard MS-COCO benchmark [162] for object detection and instance segmentation. Additionally, we report AP@0.5 and AP@0.75 to analyze performance at specific IoU thresholds. Model performance is evaluated both per-category and as an overall mAP score.

3.4.2 Datasets

DLA has historically suffered from a lack of large-scale annotated datasets due to the confidentiality of real-world document collections. However, recent efforts have led to the release of publicly available datasets, enabling further advancements in the field. We evaluate our approach on two large-scale datasets: PubLayNet [291] and HJDataset [224].

PubLayNet PubLayNet [291] is one of the most comprehensive datasets for DLA, introduced at ICDAR 2019. It consists of over 360,000 document images sourced from PubMed Central [214], making it comparable in scale to major computer vision datasets. The dataset provides five annotated categories: text, title, lists, tables, and figures. Both bounding boxes and segmentation masks are available, allowing for instance-level evaluation. For training, we use 335,703 images, while 11,245 images are used for validation. Due to the absence of released ground-truth for the official test set (ongoing competition), we report results on the validation set. A breakdown of object category distributions is presented in Table 3.1.

HJDataset HJDataset [224] consists of 2,048 historical Japanese document images containing over 250,000 annotated layout elements across seven categories. Unlike modern document datasets, HJDataset provides hierarchical structure annotations and reading order metadata, making it particularly useful for evaluating instance segmentation in complex layouts. The dataset is divided into 1,433 images for training, 307 for validation, and 308 for testing. The distribution of annotated layout instances is summarized

Table 3.1: Statistics of the PubLayNet dataset used in our evaluation.

Object Category	# Instances	
	Train	Validation
Text	2,343,356	88,625
Title	627,125	18,801
Lists	80,759	4,239
Figures	109,292	4,327
Tables	102,514	4,769
Total samples	3,263,046	120,761

in Table 3.2.

Table 3.2: Statistics of the HJDataset used in our evaluation.

Object Category	# Instances	
	Train	Validation
Body	1,443	308
Row	7,742	1,538
Title	33,637	7,271
Bio	38,034	8,207
Name	66,515	7,257
Position	33,576	7,256
Other	103	29
Total samples	181,097	31,866

3.4.3 Performance Evaluation on PubLayNet

Qualitative Analysis The qualitative results on the PubLayNet dataset, as shown in the Fig. 3.4, highlight the effectiveness of our instance segmentation model in parsing complex document layouts. Several key observations can be drawn from these results:

- *Precise Segmentation of Overlapping Elements:* The model successfully differentiates between overlapping text blocks, tables, and figures. For example, in Fig. 3.4(c) and (d), the segmentation masks correctly capture figures embedded within text regions, preventing misclassification. The results demonstrate the model's ability to capture hierarchical structures within scientific documents. Notably, in Fig. 3.4(c), text blocks, section titles, and figure captions are distinctly

(a) (b) (c)

(d) (e) (f)

Figure 3.4: **Instance segmentation results on the PubLayNet dataset.** The images showcase the model's ability to accurately segment diverse document elements. Each detected element is highlighted with distinct instance masks, demonstrating the effectiveness of the proposed approach in handling complex document layouts, overlapping structures, and multi-column formats.

segmented semantically, indicating that the model understands their contextual placement within the page.

- *Robust Detection of Nested Tables and Figures:* Tables and figures are among the most challenging elements in DLA due to their variability in size and placement. Fig. 3.4(b) showcases an example where the model effectively isolates a large table spanning multiple columns, while also simultaneously segmenting the large figure with an embedded legend. Moreover,
- *Challenges with Small Objects:* Instance-level segmentation enhances the precise extraction of smaller document elements. As shown in Fig. 3.4(a) and (e), in-

Table 3.3: Results for the PubLayNet dataset for the tasks of Document Object Detection and Document Instance Segmentation.

Category	Detection			Segmentation
	F-RCNN [210]	M-RCNN [95]	Ours	Ours
Text	0.910	0.916	0.918	0.906
Title	0.826	0.840	0.844	0.818
List	0.883	0.886	0.913	0.821
Table	0.954	0.960	0.971	0.970
Figure	0.937	0.949	0.951	0.948
AP	0.902	0.910	**0.920**	0.893
AP@0.5	-	-	0.977	0.977
AP@0.75	-	-	0.959	0.953

dividual paragraphs and lists are accurately segmented, maintaining their structure. The model also effectively separates figure and table captions, even when closely spaced, as seen in (f) and (c). However, minor inaccuracies persist for small objects like section titles and footnotes. In (a), some titles merge with text blocks, highlighting the challenge of distinguishing closely positioned text elements.

- *Multi-Column Layout Adaptability:* The results also demonstrate the model's ability to handle multi-column layouts. In Fig. 3.4(d), the segmentation masks properly differentiate two-column text arrangements, ensuring structural consistency in complex documents.

Quantitative Analysis The results of training and evaluating our proposed instance-level segmentation model on the PubLayNet dataset are presented in Table 3.3. The mean Average Precision (mAP) has been computed across all object categories, including text, lists, tables, titles, and figures. In addition, we establish a new instance segmentation baseline, evaluating the predicted masks generated by our model—an important contribution of this work. Our object detection results are also compared against state-of-the-art baselines, demonstrating an overall AP of 0.92, outperforming existing methods such as Faster R-CNN and Mask R-CNN by Zhong et al. [291]. For instance segmentation, our model achieves an overall AP score of 0.893, setting a new benchmark. In particular, high AP scores are obtained for object categories such as tables, figures, lists, and text blocks, reflecting the model's effectiveness. However, the AP score for title detection is relatively lower, likely due to the small size of titles and their variability across document layouts. Titles are sometimes misclassified as text blocks, especially when minimal spacing exists between the two elements. This challenge is also reflected in instance segmentation, where the title category achieves an AP of 0.81. Similarly, the list category records a lower segmentation AP due to false

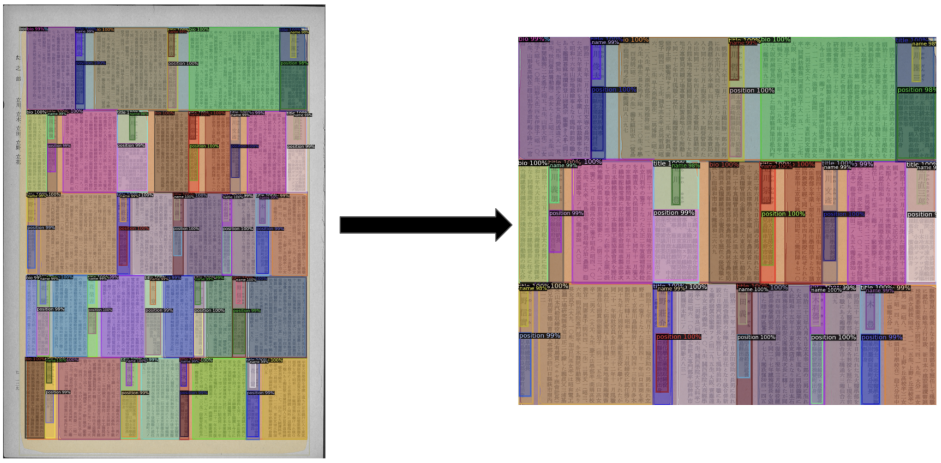


Figure 3.5: **Instance segmentation results on the Historical Japanese dataset.** The left image shows the full-page segmentation, where overlapping and nested structures challenge traditional methods. The right image presents a zoomed-in view, highlighting the model's ability to accurately differentiate hierarchical elements such as text blocks, names, and positional markers with precise instance masks.

positives, where list elements are confused with text blocks.

3.4.4 Performance Evaluation on HJDataset

Qualitative Analysis The visual results in Fig. 3.5 illustrate how our model accurately detects and segments key document elements, including titles, names, positions, and biographies, despite the inherent challenges of dense text layouts and interleaved elements. Unlike modern documents, historical manuscripts in Japanese language exhibit diverse text orientations and varying font sizes. The model effectively segments nested structures in historical documents, preserving spatial relationships and preventing misclassification of smaller elements, as shown in Fig. 3.5 (right), where a zoomed-in view highlights its ability to maintain the document's logical hierarchy. The model distinguishes vertical and horizontal text regions, ensuring proper segmentation despite layout variations. It also performs considerably well in overlapping and nested text regions, separating bio sections, names, and positions even when they appear stacked within the same spatial area. While the model performs well in most cases, some misclassifications occur in highly dense layouts, particularly where titles and body text are tightly spaced. Additionally, name-biography relationships require further refinement to reduce minor merging errors in densely populated sections.

Quantitative Analysis The quantitative results for document object detection and instance segmentation on the HJDataset are presented in Table 3.4. Our proposed model

Table 3.4: Results for the HJDataset for the tasks of Document Object Detection and Document Instance Segmentation.

Category	Detection				Segmentation
	F-RCNN [210]	M-RCNN [95]	Retina [161]	Ours	Ours
Body	0.990	0.991	0.990	0.992	0.996
Row	0.988	0.985	0.950	0.978	0.996
Title	0.876	0.895	0.696	0.891	0.913
Bio	0.945	0.868	0.895	0.937	0.944
Name	0.659	0.715	0.726	0.698	0.681
Position	0.841	0.842	0.859	0.862	0.862
Other	0.440	0.398	0.144	0.399	0.348
AP	0.819	0.813	0.752	**0.822**	0.820
AP@0.5	-	-	-	0.892	0.890
AP@0.75	-	-	-	0.876	0.878

achieves the highest overall detection AP of 0.822, surpassing Faster R-CNN [210] , Mask R-CNN [95], and the single-stage detector RetinaNet [161]. Additionally, our instance segmentation model achieves an AP of 0.820, demonstrating its effectiveness in handling complex historical document layouts. Our model excels in detecting Body, Row, and Title, achieving 0.992, 0.978, and 0.891 in detection AP, respectively. The segmentation AP for these categories is also strong, with 0.996 for both Body and Row, and 0.913 for Title. Detection and segmentation of Name and Position categories show relatively lower scores (0.698 and 0.681 for Name, 0.862 for Position). This is likely due to the smaller size and variability in their placement within historical documents. The "Other" category, which includes less frequent or ambiguous document elements, records the lowest performance across all models. However, our model still outperforms the previous baselines under overall performance.

3.4.5 Implementation Details

The model is implemented using Detectron2 [269], a PyTorch-based framework optimized for object detection. All experiments are conducted on NVIDIA Titan X GPUs. ResNeXt-101 pretrained on ImageNet [134] serves as the backbone. We show an overall summary of hyperparameters, with their values and description shown in Table 3.5. To fine-tune the model, we initialized the weights from pretrained networks [**?**] and trained only the head layers using our datasets [291, 224]. The training process spanned 30,000 iterations with an initial learning rate of 0.00025. To generate k=32 anchor boxes, various anchor scales were selected to ensure comprehensive coverage of the image. Stochastic Gradient Descent (SGD) with Nesterov Momentum was employed

Table 3.5: Choice of training hyperparameters for the proposed DOD model

Hyperparameter	Value	Description
Backbone	ResNeXt-101 [272]	Convolutional feature extractor
Batch Size	128	Number of sampled RoIs per image
Data Augmentation	Random flipping (H/V)	Horizontal and vertical flipping applied
Detection Confidence	0.7	Minimum confidence threshold for detection
Learning Rate	0.00025	Initial step size for weight updates
Learning Rate Schedule	Warmup Cosine Annealing	Adjusts learning rate every 10,000 iterations
NMS Threshold	0.3	Non-Maximum Suppression IoU threshold
Optimizer	SGD (Nesterov Momentum)	Stochastic Gradient Descent with acceleration
Pretraining Dataset	ImageNet [134]	Pretrained weights used for initialization

as the optimizer, using a batch size of 128 in the RoI heads. The learning rate followed a Warmup Cosine Annealing schedule, updating every 10,000 iterations. For inference, we set a minimum confidence score of 0.7 and applied Non-Maximum Suppression (NMS) with a threshold of 0.3. The dataloader utilized 4 worker threads for efficient processing. After fine-tuning, the testing threshold in the RoI heads was set to 0.6, as it yielded optimal results. Additionally, default data augmentation from the Detectron2 framework was applied, incorporating random vertical and horizontal flipping during training.

3.4.6 Ablation Study

To evaluate the contribution of different components, we perform ablation studies on document object detection and instance segmentation tasks.

Choice of Feature Backbones We compare ResNet-101 [96] and ResNeXt-101 [272] as backbone architectures. The results on PubLayNet, summarized in Table 3.6, show that ResNeXt-101 achieves better mAP, justifying its choice as our primary backbone.

Effectiveness of FPNs FPNs [160] enhance multi-scale feature learning. We analyze their impact using Faster R-CNN and Mask R-CNN on PubLayNet. The results, presented in Table 3.7, indicate a significant improvement in mAP when FPNs are incorporated.

Table 3.6: Backbone network comparison in terms of mAP.

Model	ResNet-101	ResNeXt-101
Faster R-CNN	0.828	0.843
Mask R-CNN	0.869	0.875

Table 3.7: Performance analysis on PubLayNet of the DOD model with and without FPN.

Model	FPN	mAP
Faster R-CNN	✗	0.843
Faster R-CNN	✓	0.871
Mask R-CNN	✗	0.875
Mask R-CNN	✓	0.904

3.5 Conclusion and Future Scope

The instance-level segmentation model developed in this chapter demonstrates strong capabilities in detecting and segmenting diverse document layouts, particularly excelling in handling overlapping structures and fine-grained elements. However, several limitations persist. The model still struggles with small object segmentation, such as section titles and footnotes, which can be misclassified due to their close proximity to larger text blocks. Additionally, while CNN-based architectures effectively capture local features, they have difficulty integrating global contextual information, leading to occasional misinterpretations in highly structured or complex layouts. Moreover, the reliance on bounding boxes for region proposals introduces spatial constraints, limiting segmentation accuracy in cases of extreme overlap or non-rectangular object structures.

To address these challenges, future research should explore more advanced architectures that integrate transformers for better global reasoning and relational modeling. A shift towards bounding-box-free approaches, such as direct mask prediction using self-attention mechanisms, could further enhance instance segmentation accuracy. Additionally, incorporating graph-based models could refine document layout understanding by capturing hierarchical relationships between elements. These advancements, as introduced in the next chapter on DocSegTr, leverage transformer-based architectures to achieve more efficient and accurate document segmentation without bounding box dependencies.

Chapter 4

DocSegTr: A Transformer Approach to Layout Segmentation

Structure is not merely the canvas of meaning—it is meaning in disguise.
– Roland Barthes

Building upon the instance-level segmentation framework introduced in the previous chapter, this work explores a transformer-based approach to address the challenges of complex document layouts. Existing CNN-based models, while effective, struggle with long-range dependencies, limiting their segmentation accuracy. To overcome these challenges, we propose DocSegTr, a transformer-driven instance segmentation model that utilizes a twin attention module for improved semantic reasoning and computational efficiency. Our approach achieves state-of-the-art or competitive AP scores on PubLayNet, PRImA, Historical Japanese (HJ), and TableBank, demonstrating its ability to generalize across diverse document structures. DocSegTr establishes a strong baseline for transformer-driven document segmentation and layout understanding.

4.1 Introduction

The field of Intelligent Document Processing (IDP) has seen rapid advances, driven by the increasing digitization of workflows across sectors such as finance, healthcare, law, and insurance. Modern Robotic Process Automation (RPA) systems have enabled a paradigm shift—transforming static documents into active agents of information

through AI systems that do not merely scan but begin to understand. This evolution signals a fundamental recognition: *the layout of a document is not simply a structural cue but also a carrier of meaning—much like language.* In recent years, state-of-the-art deep learning systems have tackled the problem of information extraction by combining text semantics with visual layout cues. Document Object Detection (DOD) methods reformulate this challenge by treating layout elements—like tables, paragraphs, and titles—as spatial "objects" to be detected and labeled via bounding boxes [149]. However, this object-centric view falls short in capturing the relational and overlapping nuances present in complex document layouts. As highlighted in our previous work [26], such rigid object boundaries limit expressiveness and lead to ambiguity in highly structured, real-world documents.

This thesis adopts a more expressive interpretation: viewing layout as a *semantic grammar*—a visual language that requires fine-grained parsing at the instance level. Instance segmentation offers a powerful paradigm here, moving beyond boxes to pixel-level masks that preserve visual syntax. Yet, common convolutional neural networks (CNNs), while adept at capturing local patterns, often lack the capacity for global reasoning and long-range dependencies crucial to decoding this layout grammar. To address these challenges, we propose a new bottom-up instance-level segmentation framework for document layouts, built on the principle of dynamic instance mask generation inspired by SOLOv2 [259]. Crucially, our method bypasses bounding box dependencies and instead interprets the document layout as a whole composition. We integrate CNNs with transformer-based architectures to jointly capture both the microstructure (local features) and the macrostructure (global context) of layout language. Transformers, with their powerful self-attention mechanisms [249], serve as global aggregators of visual semantics which is key to understanding spatial hierarchies and overlap in complex layouts [32, 6].

Moreover, we adopt the sparse twin-attention mechanism from Guo et al. [85], enabling efficient semantic reasoning while preserving computational efficiency. We introduce a novel inverse focal loss to speed convergence and improve segmentation on challenging datasets like PRImA, further supporting the thesis argument that recognizing layout as structured language benefits from both architectural innovation and tailored optimization strategies. Our contributions in this work are summarized as follows: (i) **Unified CNN-Transformer Architecture**: We present DocSegTr, a single-stage segmentation pipeline that is bounding-box-free and OCR-independent, emphasizing visual-semantic layout understanding. (ii) **Inverse Focal Loss**: A new loss function designed for faster convergence and better generalization on sparse and complex layouts. (iii) **Twin-Attention Transformer Module**: Adapted from SOTR, enabling scalable and accurate global layout parsing as demonstrated in Figure 4.1. (iv) **Layout-Aware Data Augmentation**: Inspired by LayoutParser [225], our augmentations enrich the model's ability to generalize across domains—scientific articles, magazines, and historical manuscripts alike.

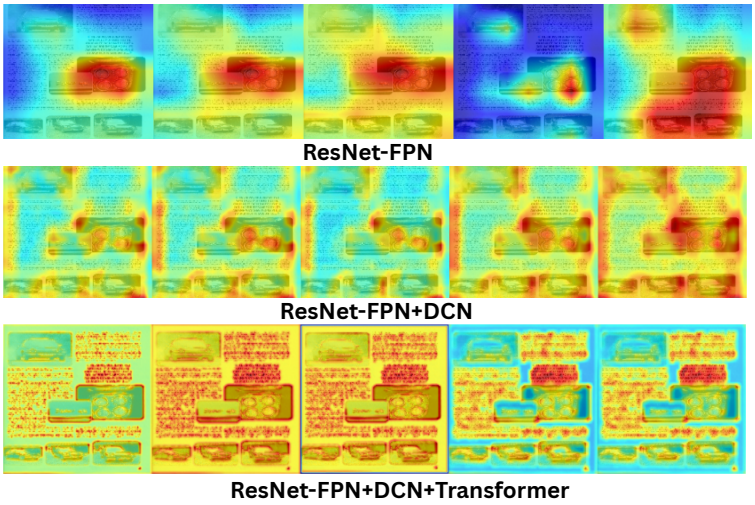


Figure 4.1: **Attention map comparison showcasing the progressive enhancement in layout understanding.** The baseline ResNet-FPN backbone captures coarse visual cues, which are refined with Deformable Convolutions (DCN). The addition of transformer layers significantly boosts contextual reasoning, allowing the model to focus sharply on both large and small layout elements, demonstrating the importance of global attention for document segmentation.

4.2 Related Work

Layout elements such as tables, text blocks, headers, and figures form the syntax of the document's visual language. Existing approaches in Document Layout Analysis have attempted to model this structure through various lenses, from heuristic rules to deep learning and transformer-based reasoning. Initial efforts to parse document layouts treated layout elements as geometric primitives, relying heavily on rule-based strategies that mimicked hand-crafted syntactic rules. These approaches broadly fell into **top-down**, **bottom-up**, and **hybrid** methodologies: *Bottom-up methods* [9] started from individual pixels or small components and grouped them iteratively into coherent regions—analogous to constructing syntax from character-level features. *Top-down strategies* [119] recursively split documents into blocks, following assumptions like Manhattan layouts or fixed column structures. These systems could segment text from graphics by applying orientation-based directional analysis. *Hybrid methods* [244] combined both cues, balancing local flexibility with structural consistency. Despite their intuitive design, such heuristic models lacked generalizability and often struggled with layout variability. However, they offered early insights into the grammatical roles that different layout elements play—particularly in structured objects like tables [70], which motivated further research into feature-aware layout modeling.

Treating document layout components as visual objects—localized and classified using object detection techniques became popular with the emergence of deep learning. Building upon natural image detectors like Faster R-CNN [210], Mask R-CNN [95], and RetinaNet [161], several notable contributions followed: DeepDeSRT [220] pioneered table structure recognition by applying object detectors to transformed document images. Fully Convolutional Networks (FCNs) [92] and frameworks like dhSegment [190] extended detection to pixel-level classification of multiple object types, including figures and tables. Instance segmentation methods such as Mask-RCNN [26] pushed this further by associating each object with precise pixel masks, enabling more granular understanding of the visual syntax, particularly in complex scientific or historical documents. To consolidate these efforts, LayoutParser [225] provided a unified deep learning toolkit for document layout analysis. At the same time, cross-domain generalization challenges led to benchmarks like PubLayNet [291] and efforts in domain adaptation [149], underscoring the need for layout understanding systems to interpret structure robustly across varying document types and distributions.

The introduction of transformer architectures [249] redefined how contextual dependencies are modeled—ushering in a new era for document understanding where layout, text, and visual appearance could be encoded jointly. Pioneering models like LayoutLM [274] and its successors [104] leveraged positional embeddings and multimodal fusion to understand the spatial arrangement and semantics of text. These models demonstrated state-of-the-art performance in Visual Document Understanding (VDU) tasks like form parsing, key-value extraction, and receipt analysis. Donut [129], SelfDoc [152], and SegGPT [176] explored OCR-free modeling and multimodal pretraining, further advocating for layout as a latent modality similar to language. In particular, [176] proposed a semantic segmentation pipeline over PubLayNet without traditional text inputs. While these methods have shown promising results, many rely heavily on OCR outputs—raising concerns around privacy, robustness, and generalization. In contrast, our thesis advocates for a **purely visual, OCR-independent approach**, where layout is learned as a compositional structure via visual features alone.

Motivated by these trends, we present *the first end-to-end transformer-based segmentation framework (DocSegTr)* that models document layout as a visual language, integrating both local visual cues (via CNNs) and global semantic context (via sparse attention transformers). Our model sidesteps bounding boxes and OCR reliance, offering a layout understanding pipeline that is more fluid, generalizable, and structurally aware. We benchmark against prior instance segmentation methods [26] and unified toolkits like LayoutParser [225], and demonstrate superior performance on datasets that challenge both layout fidelity and semantic coherence.

4.3 DocSegTr: A Layout-Aware Visual Language Parser

In line with the thesis paradigm that layout functions as a language, our proposed framework, DocSegTr, is designed as a visual parser that decodes documents into struc-

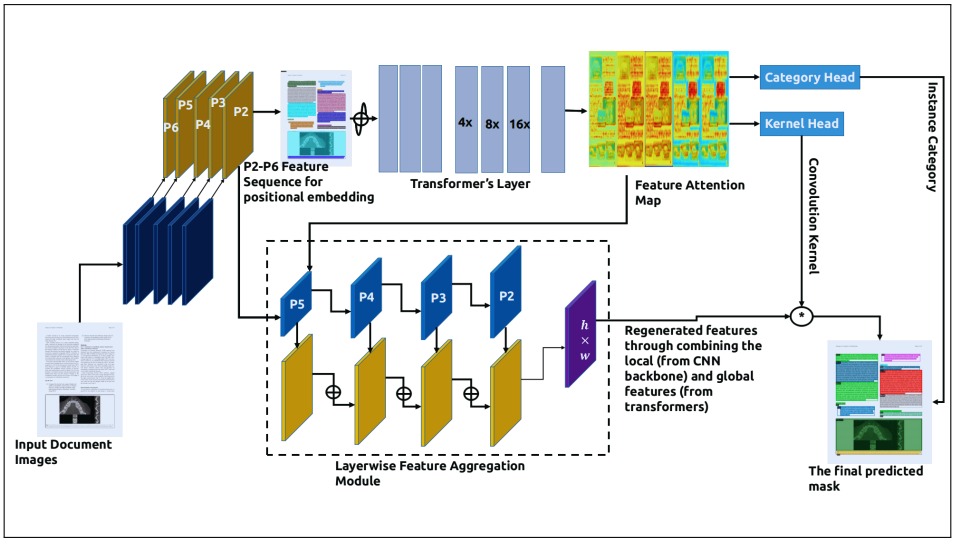


Figure 4.2: **Overview of the proposed DocSegTr architecture for instance-level document layout segmentation.** The model follows a single-stage pipeline that combines multi-scale local features extracted via a CNN-FPN backbone with global contextual reasoning via transformer layers using twin attention. The dynamic convolution-based decoder employs category and kernel heads to produce pixel-level instance masks without relying on bounding boxes or OCR. A final fusion module integrates multi-scale features through layerwise aggregation to generate high-resolution layout segmentation outputs.

tured semantic representations. To segment layout elements at the instance level, DocSegTr employs a hybrid CNN-transformer architecture that fuses local spatial encoding with long-range contextual reasoning—two essential capabilities for understanding layout grammar. Unlike traditional object detection approaches that rely on bounding box priors, our model adopts a fully end-to-end patch-wise segmentation strategy, operating directly on visual cues, independent of OCR. DocSegTr decomposes the document image into interpretable patches and assigns semantic classes to each by dynamically generating convolutional kernels for mask prediction. This section outlines the complete system pipeline and its constituent modules.

4.3.1 Architecture Overview

DocSegTr consists of three PRImAry modules that correspond to key linguistic competencies for layout understanding:

- **Local Feature Extractor (CNN + FPN):** Acts as the perceptual layer that identifies local syntactic units such as text spans, separators, and tables.

- **Transformer Encoder with Twin Attention:** Serves as the semantic aggregator, capturing dependencies and relations between layout units over spatial scales.
- **Feature Aggregator and Dynamic Decoder:** Functions as the composer, blending local and global features into unified representations and predicting segmentation masks via dynamically generated convolutional filters.

The high-level structure is illustrated in Figure 4.2, and its inner components are detailed below.

4.3.2 Modeling Layout Dependencies with Twin Attention

Understanding a document's layout language demands reasoning over both horizontal and vertical dependencies—akin to tracking how syntax flows across rows and columns. We integrate a twin attention mechanism (inspired by [85]) that efficiently captures this 2D semantic structure while significantly reducing computation compared to standard self-attention [249]. Twin attention operates in two steps: (i) **Row-wise attention** aggregates context across horizontal layout spans (e.g., paragraph continuity). (ii) **Column-wise attention** captures vertical structures (e.g., headers, hierarchical sections).

These two branches are then merged through a global attention layer, enabling the model to build a structured interpretation of layout syntax across the document. By embedding positional cues into patch-based feature maps from the CNN+FPN backbone, twin attention builds a layout-aware attention map that reflects both spatial proximity and relational importance—highlighting objects like titles, tables, and figures differently based on context (see Figure 4.1).

4.3.3 Transformer Layer: Encoding Semantic Grammar of Layout

The transformer layer is the core module responsible for building a global representation of the document. It follows a residual architecture composed of: (i) Layer Norm $\rightarrow$ Twin Attention $\rightarrow$ Residual Connection (ii) Layer Norm $\rightarrow$ MLP $\rightarrow$ Residual Connection . These layers model inter-object dependencies at multiple scales. Unlike typical multi-head attention layers, our twin attention adaptation is sparse and structured, supporting flexible layout parsing across documents of varying resolutions and hierarchies. Over a stack of K such transformer layers, we obtain a dense feature sequence encoding both local texture and global semantic roles, ready to be interpreted through task-specific heads.

4.3.4 Functional Heads: Decoding Layout Semantics

DocSegTr employs two task-specific functional heads to decode layout structure: (i) **Category Head:** A multi-layer perceptron that classifies each patch into its layout cat-

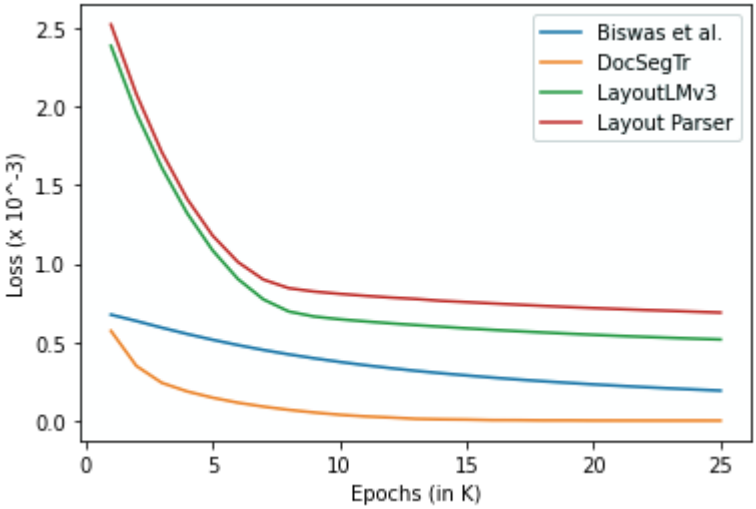


Figure 4.3: **Training loss comparison across models on the PRImA dataset.** DocSegTr demonstrates the fastest convergence and lowest final loss, outperforming prior baselines including LayoutLMv3, LayoutParser, and the Mask-RCNN-based approach by Biswas et al., highlighting the effectiveness of its dynamic segmentation strategy and inverse focal loss.

egory (e.g., paragraph, figure, title). It outputs a tensor of size $n \times n \times q_c$, where q_c is the number of semantic classes. (ii) **Kernel Head:** A linear projection head that generates dynamic convolution kernels for each patch. These kernels are later used to produce fine-grained instance segmentation masks.

To mitigate class imbalance, especially the overrepresentation of large layout objects, we introduce a novel inverse focal loss:

$$FL(p_t) = -\frac{1}{(1+p_t)^{\gamma}} \log(p_t) \tag{4.1}$$

Here, p_t represents the model's predicted confidence score for the ground-truth class, while γ is a tunable focusing parameter that adjusts the strength of down-weighting confident predictions. This loss emphasizes smaller objects (i.e., low-confidence regions) by inversely scaling the gradient contribution from well-predicted regions. As shown in Figure 4.3, this loss function improves convergence speed and accuracy, particularly in low-resource datasets where smaller objects (e.g., titles) are often overshadowed by dominant classes.

4.3.5 Compositional Segmentation via Mask Feature Fusion

Segmenting instances in layout requires a joint representation that integrates both local appearance and global context. We introduce the Layerwise Feature Aggregation Module (LFAM), which fuses: (i) High-resolution positional features from CNN layers (P2–P4) (ii) Global contextual features from the P5 transformer block. These are combined through point-wise convolution and upsampling to produce a unified $h \times w$ mask feature map that encodes layout compositionality at multiple levels (see Figure 4.2).

4.3.6 Instance Mask Prediction with Dynamic Convolution

Each patch in the document has an associated kernel from the kernel head. To segment instances, we apply dynamic convolution between the learned kernels and the unified feature map:

$$M_f^{h\times w\times n\times n} = f^{h\times w\times c} * k^{n\times n\times b} \tag{4.2}$$

- f = final feature map
- k = learned kernel
- $b = \theta^2 \cdot c$ = kernel size (with θ as kernel height/width, and c as the number of channels)
- M_f = output instance masks

This operation yields a spatial mask per patch, which is then refined using Matrix NMS [103] and optimized via Dice Loss [254], producing final instance-level layout segmentations. This reflects how layout instances are composed by dynamically convolving the learned patch-specific kernels over the shared visual-language feature map.

4.4 Experimental Validation

To assess the effectiveness of DocSegTr in decoding the visual grammar of layouts, we conducted comprehensive evaluations across four benchmark datasets, each representing distinct *layout "dialects"*—from dense scientific writing to complex historical manuscripts. Our goal is to validate the model's ability to generalize across document types and to understand which architectural components contribute most to its ability to parse layout as language. All experiments, ablations, and visualizations are reproducible, with code available at: `https://github.com/biswassanket/DocSegTr`.

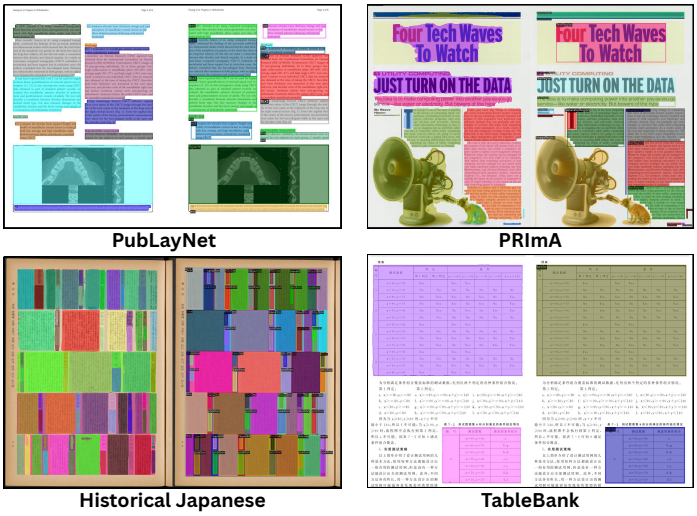


Figure 4.4: **Qualitative results of DocSegTr on four diverse benchmark datasets.** The model successfully segments complex layout structures in scientific articles (PubLayNet), magazine-style pages (PRImA), historical handwritten documents (Historical Japanese), and table-rich documents (TableBank), demonstrating strong generalization across layout styles, domains, and languages.

4.4.1 Benchmark Datasets and Evaluation Metrics

The lack of standardized datasets has long hindered layout segmentation research. However, recent public benchmarks now provide the foundation for cross-domain evaluation. We evaluate DocSegTr on: (i) **PubLayNet:** Large-scale scientific articles [291] (text, titles, figures, tables, lists). (ii) **PRImA:** Historical printed books and manuscripts with complex visual syntax. [5] (iii) **Historical Japanese (HJ):** Vertical scripts, hierarchical elements, culturally specific layout cues. [224] (iv) **TableBank:** Table-heavy documents from Word and LaTeX sources. [150] These datasets encompass diverse structural patterns, challenging the model to capture layout as a semantic visual language across domains.

Evaluation Metric: We adopt the standard mean Average Precision (mAP) for instance-level segmentation, computed over IoU thresholds [0.5–0.95]. Per-class AP is also reported to evaluate performance across different layout categories.

4.4.2 Qualitative Insights: Visual Layout Parsing in Practice

Figure 4.4 presents sample segmentations by DocSegTr across the four datasets: (i) *PubLayNet* : Clean, double-column scientific layouts are segmented with sharp accu-

Table 4.1: Quantitative evaluation of *DocSegTr* on PubLayNet and PRImA datasets compared to Layout Parser (LP), Biswas et al. (BSW), and LayoutLMv3 (LMv3). Best results are in **bold**.

PubLayNet					**PRImA**				
Object	LP	BSW	DSTR	LMv3	Object	LP	BSW	DSTR	LMv3
Text	90.1	90.6	91.1	**94.5**	Text	**83.1**	77.2	75.2	70.8
Title	78.7	81.8	75.6	**90.6**	Image	**73.6**	68.1	64.3	50.1
Lists	75.7	82.1	91.5	**95.5**	Table	**95.4**	82.4	59.4	42.5
Figures	95.9	97.1	**97.9**	**97.9**	Math	**75.6**	55.6	48.4	26.5
Tables	92.8	95.1	97.1	**97.9**	Separator	**20.6**	17.2	1.8	9.6
					Other	**39.7**	22.8	3.0	17.4
AP	86.7	89.3	90.4	**95.1**	AP	**64.7**	56.2	42.5	40.3
AP@0.5	97.2	97.7	**97.9**	-	AP@0.5	**77.6**	67.3	54.2	-
AP@0.75	93.8	95.3	**95.8**	-	AP@0.75	**71.6**	61.9	45.8	-

racy, including correctly separating overlapping captions and figures. (ii) *PRImA* : Magazine pages exhibit visual clutter and small objects. Although predictions are accurate, they are slightly blurred—likely due to class imbalance and sparse samples. (iii) *Historical Japanese* : Layouts are multi-column, densely packed, and visually noisy. DocSegTr succeeds in segmenting small, overlapping regions, a testament to its contextual modeling. (iv) *TableBank*: Tables are large and regular, making segmentation relatively straightforward. DocSegTr performs with high confidence and precision.

These visual analyses support the model's claim of being capable of learning the syntactic and semantic composition of layouts, from dominant to fine-grained structures.

Table 4.2: Quantitative evaluation of *DocSegTr* on Historical Japanese and TableBank datasets compared to Layout Parser (LP), Biswas et al. (BSW), and LayoutLMv3 (LMv3). Best results are in **bold**.

Historical Japanese					**TableBank**				
Object	LP	BSW	DSTR	LMv3	Object	LP	BSW	DSTR	LMv3
Body	99.0	**99.6**	99.0	99.0	Table	91.2	91.7	**93.3**	92.9
Row	98.8	**99.6**	99.1	99.0					
Title	87.6	91.3	**93.2**	92.9					
Bio	94.5	94.4	**94.7**	**94.7**					
Name	65.9	68.1	**70.3**	67.9					
Position	84.1	86.2	87.4	**87.8**					
Other	44.0	34.8	**43.7**	38.7					
AP	81.6	82.0	**83.1**	82.7	AP	91.2	91.7	**93.3**	92.9
AP@0.5	-	89.0	**90.1**	-	AP@0.5	94.2	94.9	**98.5**	-
AP@0.75	-	87.8	**88.1**	-	AP@0.75	92.1	92.8	**94.9**	-

Table 4.3: Ablation study evaluating the impact of architectural components within the *DocSegTr* framework on PRImA. Best results per section are highlighted in **bold**.

Model Configuration	AP	AP@0.5	AP@0.75
ResNet vs ResNeXt			
ResNet-101-FPN	20.12	31.32	16.78
ResNeXt-101-FPN	**32.59**	**58.62**	**29.73**
Deformable Convolution Networks (DCN)			
ResNeXt-101-FPN	32.59	**58.62**	**29.73**
ResNeXt-101-FPN (+DCN)	**33.21**	49.13	27.39
Importance of Transformer for Contextual Reasoning			
without transformer	5.21	7.12	3.22
without self-attention heads (with transformer layers)	29.14	41.23	20.22
DocSegTr (Overall model)	**40.31**	**59.72**	**29.54**

4.4.3 Quantitative Results Across Layout Domains

As shown in Table 4.1 and Table 4.2 , DocSegTr was comprehensively evaluated across four diverse layout domains (PubLayNet, PRImA, Historical Japanese, and TableBank) demonstrating strong generalization and adaptability. It achieved near-perfect accuracy on structured scientific layouts (PubLayNet), particularly excelling in list, figure, and table segmentation, showcasing the strength of transformer-based contextual reasoning with the utility of its twin-attention transformer decoder and multi-scale fusion. In magazine-style layouts (PRImA), while figures and tables were well-segmented, performance dropped for sparse or ambiguous classes like *Separator* and *Others*, reflecting limitations in handling visually cluttered designs. For densely packed and degraded manuscripts (Historical Japanese), DocSegTr outperformed prior models, affirming its robustness in parsing non-linear and high-density content. Finally, in the table-rich TableBank dataset, the model maintained top performance, though results suggest that text-aware models like LayoutLMv3 [104] may hold a slight edge in capturing complex table structures, emphasizing a potential complementarity between layout and textual cues.

4.4.4 Ablation Studies: Dissecting the Layout Decoder

To understand which components contribute most to layout parsing, we performed detailed ablation studies as shown in Table 4.3 on the PRImA dataset—a setting that stresses learning from fewer, more varied samples. (i) **CNN Backbone Variants:** Switching from ResNet-101 to ResNeXt-101 improves AP due to enhanced representation of local syntax. Introducing Deformable Convolutions (DCNs) yields further gains, as they help adapt the convolutional filters to irregular layout shapes—especially useful in historical or handwritten documents. (ii) **Transformer and Attention:** Removing the transformer entirely leads to a collapse in AP (5%), confirming the importance

of global contextual reasoning in layout understanding. Adding transformer layers without self-attention improves performance moderately, but integrating our twin-attention mechanism significantly enhances segmentation. This setup captures horizontal and vertical dependencies, which mirrors how humans understand layout by scanning both left-to-right and top-to-bottom. (iii) **Attention Map Visualization:** As illustrated in Figure 4.1, traditional backbones (ResNet-FPN) emphasize large, dominant objects while missing subtle layout elements. DCNs refine boundaries but lack semantic awareness. Only with transformers does DocSegTr accurately attend to both large and small instances—evidence of its structured layout parsing ability. (iv) **Cross-Domain Transfer:** To evaluate generalization, we directly transferred weights from DocSegTr pretrained on PubLayNet to the smaller PRImA dataset—without fine-tuning. Even in this *zero-shot setting*, the model achieves 15% mAP, demonstrating that DocSegTr internalizes a generalized layout grammar that can be reused across domains.

4.4.5 Implementation Details

DocSegTr is trained using SGD with Nesterov momentum (0.9) and a warm-up schedule of 1000 iterations. The initial learning rate is 0.001, reduced at 210K and 250K steps. Models are trained for 300K iterations on 2× NVIDIA A40 GPUs (48GB) using PyTorch and Detectron2, with a batch size of 8. Training each model takes approximately 4–5 days.

4.5 Conclusion and Future Directions

In this chapter, we proposed DocSegTr, a transformer-based instance segmentation framework designed to parse the visual language of documents in a *bounding-box-free, OCR-independent* manner. Built on a hybrid CNN-transformer architecture, DocSegTr decodes document layouts into fine-grained structural elements by combining local syntactic features with global semantic reasoning. Across multiple benchmark datasets, it has demonstrated strong generalization and state-of-the-art segmentation performance, especially in handling large and complex document objects.

Yet, this work also reveals some key limitations that challenge the goal of universal layout parsing:

- **Domain Sensitivity:** DocSegTr, like most supervised segmentation models, exhibits performance drops when applied to unseen domains with different layout "dialects" (e.g., transitioning from scientific reports to historical or artistic content).

- **Data Dependency:** Training robust layout models still requires substantial annotated data, which may not be available for specialized or low-resource domains.

- **Contextual Reasoning Limitations:** While transformers capture high-level relationships, smaller layout components and subtle visual cues remain underrepresented, especially under distribution shifts or limited supervision.

These limitations highlight the need for adaptive and data-efficient layout parsing systems—models that not only understand the language of layout but can adapt to new dialects and learn from sparse examples. The next chapter of this thesis addresses these open challenges through two key directions: (i) **SwinDocSegmenter** introduces a domain-adaptive layout segmentation model based on hierarchical transformers, capable of transferring learned layout grammars across visually divergent domains. (ii) **SemiDocSeg** explores semi-supervised learning to reduce reliance on annotated data, enabling layout understanding in low-resource settings by treating pseudo-labels as hypotheses to refine the layout syntax. Both works build upon the architectural foundations and visual grammar framework introduced in DocSegTr.

Chapter 5

Advancing Robustness in Document Layout Segmentation: From SwinDocSegmenter to SemiDocSeg

What we see is not what we look at—it is what we know how to look for.
– John Berger, *Ways of Seeing*

This chapter presents a comprehensive advancement in document layout segmentation, transitioning from the supervised SwinDocSegmenter to the semi-supervised SemiDocSeg framework. Motivated by the limitations of purely supervised learning in handling diverse document types, we design a transformer-based segmentation model that leverages co-occurrence priors and weak support queries to enhance performance in low-annotation regimes. SwinDocSegmenter provides a strong supervised backbone, while SemiDocSeg introduces support-guided learning and semantic class priors to address label scarcity and distribution imbalance. Through this work, we highlight the importance of structural priors and semi-supervision as a pathway toward scalable, robust, and context-aware document understanding.

5.1 Introduction

Documents are not merely containers of text and images — they follow a visual language defined by spatial hierarchies, alignment cues, and multimodal composition. Much like spoken or written language, document layouts follow a set of syntactic and

grammatical rules, from heading hierarchies to caption placement, margin consistency, and table structure. These visual grammars guide how we perceive and interpret information, and understanding them is at the core of IDP systems.

In the preceding chapter, we introduced DocSegTr [24], a bottom-up instance-level segmentation transformer that demonstrated strong performance across diverse document layout domains. While DocSegTr offered a box-free segmentation approach with powerful global reasoning, it showed limitations in three key aspects: (1) lack of mutual guidance between object detection and segmentation modules; (2) difficulty in segmenting small or low-frequency object classes; and (3) limited adaptability to new domains without significant annotated data. These challenges become especially apparent when dealing with complex magazine layouts, noisy historical scans, or layouts from unseen domains.

To overcome these shortcomings, this chapter presents SwinDocSegmenter — a unified transformer framework that treats layout parsing as a joint reasoning problem over visual tokens, capturing both the semantics and syntax of document structure. By leveraging Swin Transformers [169, 168] as the backbone, the model builds hierarchical feature maps that preserve the local and global grammatical rules of layout — akin to parsing a sentence with both word-level and sentence-level dependencies. The introduction of anchor-based dynamic queries allows segmentation to guide detection and vice versa, mimicking how readers use spatial cues and prior knowledge to contextualize ambiguous layout regions.

Inspired by recent advances in masked denoising [144], we introduce a contrastive denoising training strategy to enhance the model's sensitivity to rare or low-frequency layout tokens — akin to refining language models for better understanding of uncommon syntactic constructs. To promote domain robustness, we adopt a hybrid bipartite matching scheme [144] that enables zero-shot transfer from pretrained vision backbones (e.g., MS-COCO [162]) to layout tasks, bypassing the need for large-scale domain-specific annotation. This capability allows the model to adapt to new visual dialects such as historical manuscripts [224] or magazines [43, 5] without retraining from scratch.

To address scalability and reduce dependency on annotated corpora, we propose Semi-DocSeg, a semi-supervised framework guided by class co-occurrence. Inspired by cognitive models of contextual reasoning [128], we estimate the joint probabilities between frequent and rare layout elements. Cropped support examples containing both base and novel class patterns are used as implicit prompts, allowing the model to reason about the presence and position of novel tokens based on known ones — a visual analog to zero-shot and few-shot learning. Unlike meta-learning or prompt tuning, this approach avoids labeled support or natural language descriptions, relying instead on layout structure itself. This strategy is well-aligned with semi-supervised learning paradigms [18, 292], and has the dual benefit of reducing false positives and increasing robustness to occlusion, crucial towards production-grade document parsing.

The overall contributions of this chapter can be summarized as follows: (i) We introduce *SwinDocSegmenter*, a unified instance-level layout segmentation model combin-

ing Swin Transformer features with anchor-based content queries, achieving stronger task synergy between detection and segmentation. (ii) We propose a *contrastive denoising training scheme* to improve performance on rare or noisy layout elements and enhance representation robustness. (iiI) We implement a *hybrid bipartite matching strategy* for effective domain adaptation, enabling transfer from natural image pretraining to diverse document domains. (iv) We propose *SemiDocSeg*, a novel semi-supervised framework that leverages co-occurrence-aware support crops for contextual layout reasoning and generalization to unseen or novel classes without needing explicit labels.

5.2 Related Work

Transformers in Visual Document Understanding. Recent advances in Visual Document Understanding (VDU) increasingly treat layout as a visual language—where documents encode structured information through a learned grammar of spatial, textual, and visual cues. Transformer-based architectures, with their self-attention and positional embeddings, have become central to modeling such layout-aware representations [249]. DiT [147] achieves strong results on large-scale datasets through self-supervised pretraining but fails to generalize to visually diverse domains like PRImA. Similarly, StructText [157] fuses structured layout and text but struggles with semantically similar content blocks. Encoder-decoder models like TILT [199] and LayoutTransformer [278] improve joint modeling of text and layout but are highly reliant on OCR quality and large-scale annotations. LayoutLMv3 [104] further unifies text, layout, and visual features at the token level, achieving strong performance on benchmark tasks. However, it exhibits pretraining bias, limited domain shift adaptability, and lacks robustness on low-resource datasets with sparse class distributions. DocSegTr [24] introduces a hybrid CNN-transformer pipeline that converges well in low-data regimes, but lacks unified reasoning across detection and segmentation. Other joint-pretraining models like DocFormer [6], XYLayoutLM [84], and UniDoc [82] offer strong baselines for VDU tasks but still underperform in layout generalization particularly in class imbalance scenarios. In contrast, our proposed SwinDocSegmenter addresses these limitations through unified instance segmentation, enhanced query selection, and domain-adaptive training—modeling documents as structured visual language beyond token-level text.

Semi-Supervised Document Layout Segmentation. Despite the increasing interest in document intelligence, semi-supervised learning (SSL) for document layout analysis (DLA) remains underexplored. Most existing works on semi-supervised vision tasks focus on handwritten document segmentation or invoice detection [56, 156], and fail to generalize across diverse layout domains or handle instance-level segmentation. However, for real-world scenarios where annotated layouts are scarce or evolving, semi-supervised techniques are essential to scale layout parsing without exhaustive manual labeling. We categorize prior semi-supervised strategies into three paradigms: weakly supervised, zero-shot, and few-shot.

Weakly supervised methods aim to reduce reliance on pixel-level annotations, often leveraging pseudo labels or proposal propagation [**?**, **?**]. Yet, multi-stage pipelines involving teacher-student training or iterative refinement [262, 127] are computationally heavy and poorly suited for high-resolution document inputs. In contrast, our work pursues a one-stage transformer-based alternative tailored to dense and structured layouts. **Zero-shot methods** attempt to segment unseen classes using semantic priors or text descriptions [283, 260]. While effective in open-set scenarios, such approaches often require pretrained vision-language models or external captions, making them less viable for domain-specific documents lacking rich textual metadata. **Few-shot methods** explore instance-level adaptation from limited labeled samples [69, 285]. However, many rely on episodic training, pre-specified support sets, or contrastive reweighting [121, 258]. Meta-DETR [285], for example, introduces a transformer-based few-shot segmentation model, but requires two-stage training and lacks dynamic class discovery.

To address these limitations, our SemiDocSeg framework proposes a co-occurrence guided semi-supervised strategy: we exploit object co-occurrence distributions to dynamically encode support instances of novel classes without explicit labeling. These visual anchors are embedded into the segmentation transformer as additional queries, leveraging contextual layout dependencies to improve generalization. This setup aligns with the layout-as-language perspective (i.e. modeling documents not as static annotations, but as evolving grammars of visual and spatial co-occurrence). Unlike previous methods, our strategy supports open-vocabulary layout classes, eliminates the need for multi-stage retraining, and enables robust adaptation to underrepresented structures. As demonstrated in our experiments, this method bridges the gap between zero-shot flexibility and few-shot precision offering a principled, one-stage semi-supervised alternative for structured layout parsing.

5.3 Layout-Aware Segmentation Framework

5.3.1 Supervised Baseline: SwinDocSegmenter

The proposed *SwinDocSegmenter* introduces a unified end-to-end transformer-based architecture designed to interpret the *visual grammar* of document layouts using hierarchical representations and semantic-aware decoding. As illustrated in Figure 5.1, the architecture is composed of three main components: a hierarchical Swin Transformer backbone [169], a Transformer encoder-decoder pair, and a segmentation branch guided by class instance mapping.

Feature Extraction via Swin Transformer. The Swin Transformer encodes the input document image into multi-scale visual tokens through local window-based self-attention and patch hierarchy, enabling efficient modeling of both local layout structures and global document syntax. These features are then flattened and downsampled to reduce memory cost before being fed into the encoder.

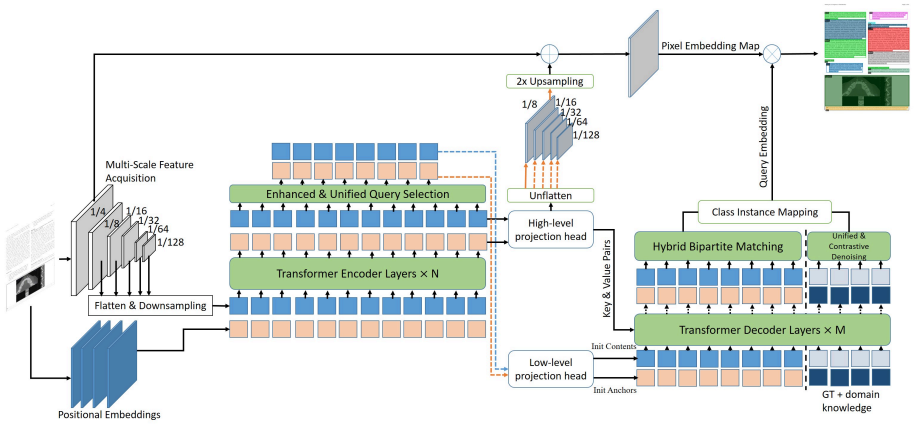


Figure 5.1: **SwinDocSegmenter architecture.** A unified transformer-based framework for document layout segmentation, combining Swin Transformer features with enhanced query selection, hybrid bipartite matching, and contrastive denoising. The model aligns pixel embeddings and semantic queries for instance-level prediction, enabling domain-shift adaptability and robust visual grammar modeling.

Encoder-Decoder Design. The encoder enhances these visual embeddings using position aware convolutional encodings. We employ a *unified mixed query selection strategy*. The classification and detection heads predict class-wise confidences, from which top-ranked features are selected as content queries. Anchors are initialized using box predictions derived from segmentation masks, bridging pixel-level semantics and region-level geometry. The decoder applies deformable attention layers [270] with Contrastive Denoising Training (CDN) [286], handling hard negatives and ambiguous layout instances via layer-wise gradient propagation. The hybrid decoder outputs are matched to ground truth masks using a hybrid bipartite matching loss involving class, localization, and mask similarity.

$$\mathscr{L}_{\text{CDN}} = \sum_{j=1}^{N} \left(\mathscr{L}_{\text{cls}}(q_j, c_j) + \mathscr{L}_{\text{reg}}(b_j, \hat{b}_j) + \mathscr{L}_{\text{mask}}(m_j, \hat{m}_j) \right) \tag{5.1}$$

Here, q_j denotes the output query embedding from the decoder for the j^{th} instance. c_j, b_j, and m_j correspond to the ground truth class label, bounding box, and binary segmentation mask respectively, while $\hat{b}_j$ and $\hat{m}_j$ denote the predicted bounding box and mask for the same instance. The loss function is composed of three terms: the classification loss $\mathscr{L}_{\text{cls}}$ (typically cross-entropy or focal loss), the regression loss $\mathscr{L}_{\text{reg}}$

(e.g., L_1 or G_{IoU} loss), and the segmentation loss $\mathscr{L}_{\text{mask}}$ (e.g., Dice loss or Binary Cross-Entropy). The total loss is accumulated across all N matched object queries using a bipartite matching algorithm.

Segmentation Output. A Pixel Embedding Map (PEM) is constructed by fusing high-resolution features from the Swin backbone and encoded Transformer tokens. Final instance segmentation is performed through a dot product between learned query embeddings and the PEM, enabling end-to-end learning of document instances. Eq. 5.2 details the fusion mechanism used in mask prediction.

$$\hat{M}_i = \sigma\left(Q_i^\top \cdot \text{PEM}\right) \tag{5.2}$$

Here, $Q_i \in \mathbb{R}^d$ denotes the i^{th} decoder query embedding, and $\text{PEM} \in \mathbb{R}^{d \times H \times W}$ represents the pixel embedding map, obtained by fusing multi-scale features from both the Swin Transformer backbone and the Transformer encoder. The function $\sigma(\cdot)$ is the element-wise sigmoid activation used to generate pixel-wise probabilities. The output $\hat{M}_i \in \mathbb{R}^{H \times W}$ is the predicted binary mask corresponding to instance i.

Projection Heads and Contrastive Learning. The decoder performance is strengthened using low-level and high-level projection heads. The low-level projection head $\mathscr{L}_{\text{low}}$ (Eq. 5.3) enforces fine-grained visual distinctions using a contrastive loss, while the high-level projection head $\mathscr{L}_{\text{high}}$ (Eq. 5.4) introduces class-level prototypes for semantic regularization.

$$\mathscr{L}_{\text{low}} = -\log \frac{\exp(\text{sim}(z_i, z_j)/\tau)}{\sum_{k=1}^{N} \exp(\text{sim}(z_i, z_k)/\tau)} \tag{5.3}$$

Here, z_i and z_j are the projected feature embeddings corresponding to a pair of positive instances. The function $\text{sim}(\cdot,\cdot)$ denotes the cosine similarity between embeddings, τ is a temperature scaling factor that controls the sharpness of the distribution, and N is the total number of projected embeddings in the batch used for normalization.

$$\mathscr{L}_{\text{high}} = -\sum_{c=1}^{C} y_c \log \frac{\exp(\text{sim}(q_i, p_c)/\tau)}{\sum_{k=1}^{C} \exp(\text{sim}(q_i, p_k)/\tau)} \tag{5.4}$$

Here, q_i is the query embedding corresponding to instance i, and p_c denotes the prototype embedding for class c. The indicator variable y_c represents the ground-truth label for class c, and C is the total number of classes considered in the classification task.

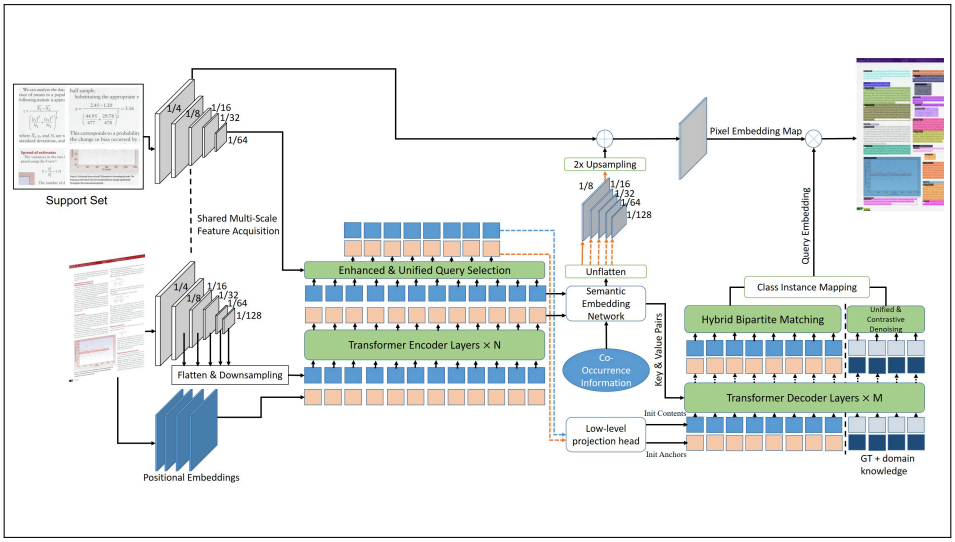


Figure 5.2: **The SemiDocSeg Setup.** We introduced a support set and extracted the features with shared Swin Transformer backbone. Later on, we utilize a semantic embedding network with utilizing the co-occurrence information.

The overall training objective of the SwinDocSegmenter unifies the segmentation supervision with both fine-grained and semantic-level contrastive signals. This is achieved through a composite loss function that balances instance mask prediction with two contrastive regularization terms (λ_1 and λ_2 as shown in eq. 5.5 :

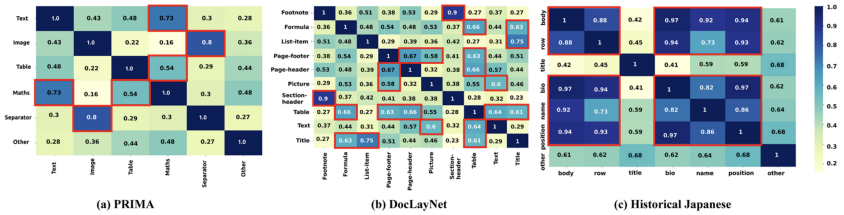
$$\mathscr{L}_{\text{total}} = \mathscr{L}_{\text{seg}} + \lambda_1 \cdot \mathscr{L}_{\text{low}} + \lambda_2 \cdot \mathscr{L}_{\text{high}} \tag{5.5}$$

5.3.2 Semi-Supervised Extension: SemiDocSeg

To improve generalization under low-data regimes and domain shifts, we extend SwinDocSegmenter with a semi-supervised strategy, termed **SemiDocSeg** as illustrated in Figure 5.2, that leverages a visual-semantic co-occurrence prior for query conditioning and label propagation.

Support Set Integration. A shared Swin backbone is used to extract features from both support and query images. A co-occurrence matrix is computed over the labeled base dataset to identify semantic affinity between classes. This matrix guides the selection of novel and base classes used in training, allowing high co-occurrence classes to be transferred via pseudo-labeled support examples (see Figure 5.3).

Class-Specific Query Generation. Class-conditional queries are constructed by merg-



(a) PRIMA (b) DocLayNet (c) Historical Japanese

Figure 5.3: **Computation of Co-Occurrence Matrix.** The initial class-wise count matrix is transformed into a symmetric co-occurrence matrix via conditional and max marginalization. The resulting prior encodes inter-class layout dependenciesß. Red boxes indicate the classes have high co-occurrence with the rest

ing semantic vectors (from the support set) with the decoder's object queries. This allows the decoder to specialize in detecting novel instances despite the absence of direct annotations . Semantic projections are handled by a low-level projection head that embeds support semantics into the visual space as shown in eq. 5.6.

$$\mathbf{q}_c = f_{\text{proj}}(f_{\text{backbone}}(x_c)) \cdot P_c \tag{5.6}$$

where x_c is the support image for class c, f_{proj} is the low-level projection, and P_c is the co-occurrence prior vector for class c.

Modified CDN and Bipartite Matching. The CDN loss is extended to operate per class, optimizing three separate heads: regression, classification, and contrastive. In contrast to the fully supervised case, here each query carries a class-specific identity, and only matches its own class anchors. Bipartite matching (Eq. 8) is modified to consider class-specific losses ΔC_j, enforcing tighter coupling between visual instance and class semantics as shown in eq. 5.7.

$$\Delta C_j = \lambda_{\text{cls}} \cdot \mathscr{L}_{\text{cls}}(y_j, \hat{y}_j) + \lambda_{\text{reg}} \cdot \mathscr{L}_{\text{reg}}(b_j, \hat{b}_j) + \lambda_{\text{con}} \cdot \mathscr{L}_{\text{con}}(z_j, \hat{z}_j) \tag{5.7}$$

where y_j, b_j, and z_j are the class, box, and feature for the j^{th} query; and $\hat{y}_j$, $\hat{b}_j$, $\hat{z}_j$ are the predicted counterparts.

The final training loss aggregates across labeled and pseudo-labeled data (refer eq. 5.8:

$$\mathscr{L}_{\text{total}} = \mathscr{L}_{\text{CDN}}^{\text{labeled}} + \alpha \cdot \mathscr{L}_{\text{CDN}}^{\text{pseudo}} + \beta \cdot \mathscr{L}_{\text{contrastive}} \tag{5.8}$$

where $\mathscr{L}\text{CDN}^{\text{labeled}}$ denotes the loss computed over fully labeled queries using the three-head CDN objective, $\mathscr{L}\text{CDN}^{\text{pseudo}}$ corresponds to the loss on pseudo-labeled

queries propagated from the support set, and $\mathscr{L}_{\text{contrastive}}$ enforces semantic consistency between query and support embeddings. The scalars α and β act as balancing coefficients for the semi-supervised terms.

5.4 Experimental Validation

5.4.1 Benchmark Datasets and Evaluation Metrics

To evaluate the capability of SwinDocSegmenter [13] and its semi-supervised extension SemiDocSeg [15] in interpreting the visual syntax and semantics of document layouts, we perform extensive experiments across four diverse benchmarks: PRImA [43, 5], HJDataset [224], TableBank [150], and DocLayNet [197]. Each dataset reflects a different dialect of visual communication—from catalog-style object-centric layouts to dense, multilingual scientific and historical formats. Our experiments aim to demonstrate (i) how SwinDocSegmenter benefits from hierarchical vision backbones for robust segmentation under full supervision, and (ii) how SemiDocSeg successfully transfers layout priors across domains and enhances generalization in low-data regimes by modeling co-occurrence-driven class semantics. The complete codebase and pretrained models are available at: `https://github.com/ayanban011/SwinDocSegmenter`.

Evaluation: For evaluation, we adopt standard instance-level segmentation metrics including mean Average Precision (mAP) at multiple IoU thresholds (e.g., AP@0.50, AP@0.75), Dice coefficient, and IoU, alongside per-class accuracy and Macro F1-score in low-data or semi-supervised scenarios. This unified evaluation strategy allows us to quantify both spatial localization and semantic understanding, key aspects of decoding the visual grammar of documents.

5.4.2 Qualitative Insights: Visual Layout Parsing in Practice

Figure 5.4 showcases the qualitative results of the proposed *SwinDocSegmenter* across four distinct document layout benchmarks, each presenting unique structural and visual challenges. The visualizations confirm the model's capacity to generalize and accurately decode a wide variety of layout styles by leveraging only visual cues. In Figure 5.4(a), we observe the model's performance on the PRImA dataset, which is composed of richly designed magazine pages. Despite the artistic and cluttered nature of these layouts, SwinDocSegmenter successfully distinguishes between visually similar elements like furniture images and textual annotations. The segmentation masks tightly align with object boundaries, and even smaller components such as legends and side notes are precisely isolated—demonstrating robustness to scale and layout density. Figure 5.4(b) illustrates outputs on the Historical Japanese (HJ) dataset, characterized by vertically aligned handwritten scripts and complex page arrangements. These historical manuscripts exhibit a highly structured yet non-standard layout with tight interline spacing and no modern separators. SwinDocSegmenter successfully

Figure 5.4: **Sample document layouts from benchmark datasets used in our study.** (a) **PRImA**: Scanned magazine pages with diverse furniture elements and decorative layouts. (b) **HJ**: Historical Japanese books exhibiting complex multi-column structures and dense text blocks. (c) **TableBank**: Academic documents featuring tabular content in multiple languages and layouts. (d) **DocLayNet**: Modern digital documents with varied design, including articles, advertisements, and mobile interfaces. These datasets collectively represent diverse layout structures and domains essential for robust document layout segmentation.

segments individual blocks of vertical text, maintaining accurate separation even in densely populated regions. This showcases the model's ability to adapt to culturally and linguistically diverse layout grammars. Figure 5.4(c) focuses on TableBank, which contains scanned and digital documents with table-centric layouts. The model exhibits outstanding table boundary delineation, even in the presence of visually subtle grid lines and varied font styles. It precisely captures table structure while avoiding spillover into surrounding text, confirming its sensitivity to intra-document structure variation. Finally, Figure 5.4(d) presents results on DocLayNet, a recently proposed industrial document benchmark featuring a diverse mixture of posters, forms, and multi-column reports. The model effectively segments overlapping elements such as headers, footers, captions, and images—even in visually complex or heavily decorated layouts. Notably, it maintains fine-grained boundaries across small entities like footnotes or floating buttons (e.g., mobile UI icons), demonstrating resilience to noisy backgrounds and decorative elements. Together, these examples highlight the strong visual generalization capabilities of SwinDocSegmenter. Its ability to handle structured, unstructured, and culturally diverse layouts using only visual features positions it as a strong visual-only baseline for document layout understanding

Table 5.1: Performance comparison of **SwinDocSegmenter** against state-of-the-art methods on the PubLayNet and PRImA benchmarks. Bold values indicate the best result per category.

	PubLayNet					**PRImA**			
Object	LP	DocSegTr	LMv3	Swin	**Object**	LP	DocSegTr	LMv3	Swin
Text	90.1	91.1	94.5	**94.55**	Text	83.1	75.2	70.8	**87.72**
Title	78.7	75.6	**90.6**	87.15	Image	73.6	64.3	50.1	**75.92**
Lists	75.7	91.5	**95.5**	93.03	Table	**95.4**	59.4	42.5	49.89
Figures	95.9	97.9	97.9	**97.91**	Math	75.6	48.4	46.5	**78.19**
Tables	92.8	97.1	**97.9**	97.25	Separator	20.6	1.8	9.6	**27.56**
–	–	–	–	–	Other	39.7	3.0	17.4	7.05
AP	86.7	90.4	**95.1**	93.72	AP	64.7	42.5	40.3	**54.39**
AP@0.5	97.2	97.9	–	**97.94**	AP@0.5	77.6	54.2	–	**69.31**
AP@0.75	93.8	95.8	–	**96.28**	AP@0.75	71.6	45.8	–	**52.97**

5.4.3 Quantitative Results on Supervised Benchmarks

To validate the effectiveness of SwinDocSegmenter, we conducted comprehensive evaluations across four benchmark datasets: PubLayNet, PRImA, Historical Japanese (HJ), and TableBank. Each dataset reflects a distinct layout "dialect", ranging from dense modern layouts to structured historical manuscripts. The results demonstrate that our model achieves strong performance in segmenting visually rich layouts, despite relying solely on visual cues and without any OCR-derived text embeddings.

As shown in Table 5.1, SwinDocSegmenter attains competitive or superior performance compared to LayoutLMv3 [104] and LayoutParser [225], both of which incorporate textual signals during training. On PubLayNet, it matches or outperforms these methods across most categories, particularly excelling in detecting text and list elements. While performance for the "Title" class slightly lags behind (likely due to missing text semantics), SwinDocSegmenter surpasses LayoutLMv3 in both *AP*@0.5 and *AP*@0.75, despite using only visual features. It also rivals DiT [147] and UDoc [82] (well-established multimodal baselines) without requiring large-scale pretraining. In the case of PRImA, which features historical magazine layouts with smaller objects, LayoutLMv3 underperforms due to weak visual generalization. SwinDocSegmenter overcomes this limitation and sets a new benchmark in detecting fine-grained regions like separators and mathematical content. The only exception lies in the "Other" category, which lacks clear visual structure, making it inherently ambiguous without auxiliary text cues. Further insights are evident in Table 5.2. On the Historical Japanese dataset, our model marginally surpasses DocSegTr [24] overall but shows marked improvement in difficult semantic categories such as "Name" and "Position". On TableBank, SwinDocSegmenter achieves a significant leap (+5% AP) over all previous methods, establishing itself as a strong table detector for documents with minimal layout variability.

Table 5.2: Evaluation of **SwinDocSegmenter** on the Historical Japanese and TableBank datasets. Notable improvements are observed in key semantic categories.

Historical Japanese					**TableBank**				
Object	LP	DocSegTr	LMv3	Swin	**Object**	LP	DocSegTr	LMv3	Swin
Body	99.0	99.0	99.0	**99.72**	Table	91.2	93.3	92.9	**98.04**
Row	98.8	**99.1**	99.0	99.0	–	–	–	–	–
Title	87.6	**93.2**	92.9	89.5	–	–	–	–	–
Bio	94.5	**94.7**	94.7	86.26	–	–	–	–	–
Name	65.9	70.3	67.9	**83.8**	–	–	–	–	–
Position	84.1	87.4	87.8	**93.0**	–	–	–	–	–
Other	**44.0**	43.7	38.7	40.57	–	–	–	–	–
AP	81.6	83.1	82.7	**84.55**	AP	91.2	93.3	92.9	**98.04**
AP@0.5	–	90.1	–	**90.78**	AP@0.5	–	98.5	–	**98.95**
AP@0.75	–	88.1	–	**88.22**	AP@0.75	–	94.9	–	**98.90**

Table 5.3: Performance comparison on **DocLayNet** benchmark. **MR**: MaskRCNN, **FR**: FasterRCNN, **YV5**: YOLOv5. Results are reported in terms of Average Precision (AP) per class.

Class	**MR**	**FR**	**YV5**	**Ours**
Caption	71.5	70.1	77.7	**83.56**
Footnote	71.8	73.7	**77.2**	64.82
Formula	63.4	63.5	**66.2**	62.31
List-item	80.8	81.0	**86.2**	82.33
Page-footer	59.3	58.9	61.1	**65.11**
Page-header	70.0	**72.0**	67.9	66.35
Picture	72.7	72.0	77.1	**84.71**
Section-header	69.3	68.4	**74.6**	66.50
Table	82.9	82.2	**86.3**	**87.42**
Text	85.8	85.4	88.1	**88.23**
Title	80.4	79.9	**82.7**	63.27
Mean AP	73.5	73.4	**76.8**	**76.85**

Finally, we introduce SwinDocSegmenter as the **first Transformer-based layout segmentation model evaluated on DocLayNet**, a large-scale industrial dataset [197]. Results in Table 5.10 reveal that our method competes robustly against CNN-based architectures such as MaskRCNN and FasterRCNN, outperforming them on several structural components including "Caption", "Picture", "Page-footer", and "Text". These

gains underscore SwinDocSegmenter's visual grammar decoding capabilities, without relying on OCR signals, making it a viable foundation for layout understanding in diverse document types.

5.4.4 Ablation Studies: Dissecting Model Design Choices

To rigorously evaluate the architectural and training decisions of SWINDOCSEGMENTER, we conduct extensive ablation studies centered around five key aspects: the feature extraction backbone, input image resolution, the number of decoder queries, the choice of learning objectives, and pre-training initialization. Unless otherwise specified, all ablations are performed on the PRImA benchmark due to its compact size and layout complexity.

Impact of Feature Extraction Backbone. We begin by analyzing the effect of the backbone on instance-level segmentation performance. As shown in Table 5.4, convolutional backbones such as ResNet and ResNeXt focus well on local features but lack the global context needed for understanding larger layout regions. Vision Transformers (ViTs) introduce self-attention but require considerable training data to reach stable generalization. Swin Transformers offer a trade-off through hierarchical representations, with SWIN-L achieving the best performance, outperforming even ViT-B by a margin of nearly 8

Table 5.4: Effect of Feature Extraction Backbone on PRImA. **Bold** indicates best performance.

Backbone	**No. of Parameters**	**AP**	**AP@50**	**AP@75**	**AP_S**	**AP_M**	**AP_L**
ResNet-50	52M	36.065	52.362	41.112	20.152	23.327	38.142
ResNet-101	102M	37.112	54.982	41.872	22.242	26.153	41.986
ResNeXt-101	104M	38.405	58.405	41.916	25.982	29.364	44.129
ViT-S	126M	40.342	59.763	42.158	29.176	33.129	48.526
ViT-B	164M	46.128	62.689	47.358	31.389	33.458	50.508
Swin-T	178M	49.349	65.956	50.317	34.128	36.909	52.049
Swin-L	**223M**	**54.393**	**69.313**	**52.965**	**39.327**	**42.061**	**60.142**

Impact of Input Image Resolution. Given the model's size, input resolution plays a significant role in training stability and layout fidelity. Table 5.5 demonstrates that performance improves substantially with higher resolutions, as coarse features in low-resolution inputs limit the model's discriminative ability. However, memory constraints prevented us from experimenting beyond 1024×1024 resolution.

Impact of Decoder Queries. The number of decoder queries in a DETR-style model determines how many object proposals are learned. Table 5.6 reveals that using fewer queries (e.g., 100) reduces model expressiveness, while 300 queries strikes a balance

Table 5.5: Effect of Input Image Resolution. **Bold** indicates best.

Resolution	AP	AP@50	AP@75	AP_S	AP_M	AP_L
256 × 256	45.02	60.19	46.26	28.37	32.46	53.57
512 × 512	50.13	66.24	52.32	32.24	36.91	54.15
1024 × 1024	**54.39**	**69.31**	**52.97**	**39.33**	**42.06**	**60.14**

between recall and memory constraints.

Table 5.6: Effect of Number of Decoder Queries. **Bold** indicates best.

Queries	AP	AP@50	AP@75	AP_S	AP_M	AP_L
100	50.02	65.19	52.26	32.37	36.46	53.97
150	50.13	66.24	52.32	32.24	36.91	54.15
200	51.39	67.31	52.77	37.31	41.01	60.11
250	52.09	68.21	52.96	37.51	42.06	60.13
300	**54.39**	**69.31**	**52.97**	**39.33**	**42.06**	**60.14**

Impact of Loss Objectives. We also evaluate combinations of reconstruction and classification objectives. The L1 + Focal loss combination yields the best performance. The L1 component promotes sparse and precise bounding masks, while Focal Loss down-weights easy negatives, encouraging learning from hard examples.

Effect of Pre-training Biases. Lastly, we examine how the pre-training dataset influences model performance (Table 5.7). When pre-trained on PubLayNet [291], the model shows strong performance for overlapping classes (e.g., Table) but struggles with rare classes like Separator. In contrast, MS-COCO [162] pretraining yields better generalization and balance due to its diverse query-space and task-agnostic pretraining.

Table 5.7: Pre-training Biases: PubLayNet vs. MS-COCO. **Bold** indicates best.

Pretraining	Overall						Class-wise (PRImA)					
	AP	AP@50	AP@75	AP_S	AP_M	AP_L	Text	Image	Table	Math	Sep.	Other
PubLayNet	49.36	64.43	51.45	32.94	34.07	54.21	85.55	72.51	**70.68**	56.05	8.55	2.83
MS-COCO	**54.39**	**69.31**	**52.97**	**39.33**	**42.06**	**60.14**	**87.72**	**75.92**	49.89	**78.19**	**27.56**	**7.05**

5.4.5 Evaluating Semi-Supervised Settings with SEMIDOCSEG

To understand the generalization capability of our semi-supervised setup, we begin by visualizing the labeled training instances across two datasets. As shown in Table 5.8,

Table 5.8: Training Instances Distribution in Semi-Supervised Setup (PRImA and DocLayNet)

PRImA						DocLayNet					
Category	**#Instances**	**Category**	**#Instances**	**Category**	**#Instances**	**Category**	**#Instances**	**Category**	**#Instances**	**Category**	**#Instances**
TextRegion	0	ImageRegion	0	TableRegion	0	Caption	0	Picture	0	Table	0
MathsRegion	**27**	SeparatorRegion	477	OtherRegion	**34**	Text	0	Footnote	5964	Formula	22367
–	–	–	–	–	–	List-item	**170889**	Page-footer	64717	Page-header	50700
–	–	–	–	–	–	Section-header	18003	Title	**4423**	–	–
Total			**538**			**Total**				**337,063**	

several class categories (e.g., *TextRegion*, *ImageRegion*, *TableRegion*) have no annotated training samples and are only seen during testing. These novel classes are indirectly learned through co-occurrence priors and support set propagation, demonstrating that our framework enables scalable semi-supervised learning without modifying model complexity.

We evaluate the proposed SEMIDOCSEG approach under three setups:

Full Test Set Evaluation (Labeled + Unlabeled). In this setting, the model is evaluated on the complete test set. As shown in Table 5.9, performance on PRImA drops by approximately 9% compared to the fully supervised baseline due to limited labels and PRImA's small training size. This illustrates the overfitting risk for large models (223M parameters) when training on scarce annotations. However, the performance on DocLayNet (Table 5.10) remains competitive, showing only 3% drop—attributed to its abundance of weakly supervised signals through support queries and co-occurrence.

Table 5.9: Performance on PRImA Dataset under Semi-Supervised Settings

	Text	**Image**	**Table**	**Maths**	**Separator**	**Other**	**AP**	**AP@50**	**AP@75**	**AP_S**	**AP_M**	**AP_L**
Overall	**81.2**	**70.5**	**40.6**	**53.3**	**26.1**	**3.7**	**45.9**	**61.6**	**48.8**	**39.0**	**38.7**	**47.9**
Base	–	0.1	–	45.3	10.2	0.4	9.3	15.4	7.7	1.0	2.6	10.4
Novel	70.0	50.6	12.8	–	–	–	25.9	38.9	26.3	26.7	26.6	29.2

Supervised-Only Evaluation (Labeled Classes Only) This setup evaluates performance exclusively on labeled classes. As visible in both datasets, performance on PRImA remains modest due to data scarcity (e.g., only 27–34 training instances for certain classes). This highlights overfitting risks for large transformers and motivates future distillation efforts. In contrast, DocLayNet performs better in this setup, validating the value of labeled diversity.

Zero-Shot Evaluation (Unlabeled Classes Only) This evaluates model generalization on novel/unseen classes. SEMIDOCSEG shows promising results for such classes via layout co-occurrence priors and support guidance. However, performance on generic labels like "Text" (DocLayNet) is reduced due to semantic overlap with other labeled regions.

The semi-supervised evaluation demonstrates that SemiDocSeg effectively leverages

Table 5.10: Performance evaluation of semi-supervised settings on DocLayNet dataset

	Caption	Footnote	Formula	List-item	Page-footer	Page-header	Picture	Section-header	Table	Text	Title	AP	AP@50	AP@75
Overall	**82.7**	62.1	62.3	76.3	**66.4**	66.4	81.7	63.1	83.1	83.1	77.2	73.1	**90.9**	79.6
Base	-	**63.7**	**62.9**	**79.2**	65.2	**66.7**	-	67.8	-	-	**80.1**	**74.1**	90.0	**80.2**
Novel	38.9	-	-	-	-	-	37.3	-	44.6	3.6		31.3	38.7	35.3

support sets and co-occurrence cues to generalize across both labeled and unlabeled layout classes, achieving competitive performance with significantly fewer annotations. On the PRImA dataset, despite its limited size, the model attains an overall AP of 45.9%, with particularly strong performance on small objects (AP_s = 39.0%), addressing a key challenge in document layout analysis. In the larger DocLayNet benchmark, the performance remains stable (AP = 73.1%, AP@50 = 90.9%), showing minimal drop compared to the supervised baseline. Compared to state-of-the-art few-shot and zero-shot instance segmentation methods, SemiDocSeg achieves superior or comparable accuracy while maintaining a more straightforward and unified training strategy. These results highlight the framework's practical scalability and robustness in low-label and high-structure document environments, paving the way for more annotation-efficient layout understanding systems.

5.4.6 Implementation Details

We train our model using the Adam optimizer with an initial learning rate of 1×10^{-5}, applying a cosine annealing scheduler over 5000 cycles and a weight decay of 1×10^{-6}. The training is conducted for 300K iterations, with a learning rate drop by an order of magnitude between 230K and 270K iterations. All experiments are performed on an NVIDIA A40 GPU with 48 GB RAM, completing in 6 days using stochastic learning. In comparison, the conventional supervised training for SwinDocSegmenter [13] typically requires approximately 2 weeks. The implementation is based on the PyTorch and Detectron2 libraries.

5.5 Conclusion and Future Work

In this first part of the thesis, we explored the **Interpretation** axis of the proposed theme *"Layout as Language"*, focusing on the semantic understanding of document layouts through deep learning-based instance segmentation. We framed document layout elements not merely as visual regions, but as meaningful units analogous to linguistic constructs, whose structured relationships encode document semantics. In this last chapter of Part-I, we began with a robust supervised model, **SwinDocSegmenter**, designed to interpret complex layouts across domains with hierarchical tokenization and query-based decoding. The extensive ablation studies demonstrated the model's flexibility and effectiveness across various architectural and data design choices. Recognizing the limitations posed by the scarcity of labeled data, we extended this ap-

proach to a co-occurrence-guided semi-supervised framework, **SemiDocSeg**, which introduced novel ways to leverage support sets and unlabeled classes using weak priors like layout co-occurrence. Our comparative studies showed that this model not only generalizes well to unseen layout categories but also significantly improves the performance on low-resource settings, outperforming several few-shot and zero-shot baselines. The experiments across PRImA and DocLayNet revealed that layout understanding is not solely a matter of recognizing visual patterns, but of interpreting structural priors embedded in document design. By treating layout as a communicative system, we showcased that structural and semantic context can serve as an implicit form of supervision, reducing dependence on full annotations.

Future Work: Moving forward, this layout-as-language perspective opens up multiple promising research directions:

- **Structural Prompting:** Exploring the integration of layout prompts or layout-based attention cues for adaptive segmentation in evolving document types.
- **Layout Reasoning:** Extending interpretation to relational reasoning tasks such as understanding logical reading order, visual discourse structures, and layout entailment.
- **Cross-modal Alignment:** Bridging layout interpretation with textual or multimodal representations, allowing models to perform layout-conditioned comprehension or VQA-style tasks.
- **Efficient and Compact Models:** Incorporating knowledge distillation and transformer pruning for real-time layout interpretation in on-device or low-resource environments.

With this, we conclude Part-I of the thesis. In the subsequent parts, we shift focus from interpreting layout to **representing** and **generating** it — continuing our journey of treating layout not just as a visual artifact, but as a rich language to be understood, learned, and expressed.

Part II

Representation

Chapter 6

Encoding Structure as Language: Towards Graph-based Representation of Document Layouts

The structure of information is as important as the information itself.
– Edward Tufte

---

*This chapter explores the evolution of document layout modeling through the lens of graph-based reasoning. We introduce **Doc2GraphFormer**, a lightweight hybrid framework that treats document structure as a language by representing it as a multimodal graph and inferring semantic relations via transformer-based attention. The model jointly addresses key tasks such as Semantic Entity Recognition (SER), Subgraph Clustering, and Relation Extraction (RE) using shared node representations and task-specific heads. Evaluated on standard benchmarks like FUNSD and XFUND, Doc2GraphFormer delivers state-of-the-art results while remaining computationally efficient, thanks to its parameter-light design.*

6.1 Introduction

Understanding documents involves more than just reading text—it requires grasping the structure, semantics, and visual organization that together convey meaning. In the spirit of this thesis's guiding theme, "*Layout as Language*," we view document layouts not merely as spatial arrangements, but as syntactic constructs—comparable to the

grammar of natural language. Titles act as headlines, tables as semantic groupings, and spatial proximity often mirrors conceptual relationships. Traditional NLP models such as BERT [60] and RoBERTa [167], while highly effective in natural language tasks, process text in a purely sequential manner and lack awareness of spatial or visual structures, making them ill-suited for understanding documents where meaning is often conveyed through layout, formatting, and visual cues [274, 273, 104].

Graph-based reasoning offers a natural lens to interpret this structural language. Our line of work begins with *Doc2Graph* [78], a unified graph representation framework for structured document understanding. It captures layout entities (e.g., text blocks, images, tables) as nodes, and uses edges to encode their spatial and semantic dependencies. This task-agnostic formulation supports both semantic entity recognition (SER) and relation extraction (RE) by modeling documents as structured graphs, rather than unstructured token sequences. However, classical Graph Neural Networks (GNNs) used in Document AI based on GraphSAGE [88] often suffer from local message passing limitations, struggling with long-range dependencies and global layout reasoning. To address this, we extended the paradigm with *GeoContrastNet* [22], which introduces a contrastive learning objective that aligns geometric (layout) features with high-level semantics. This technique improves spatial discrimination in graph reasoning, empowering the model to better group related entities that are spatially distant but structurally connected.

Going further, *Doc2Graph-X* [183] expands the graph representation framework to multilingual settings. By integrating multilingual text embeddings at both word and sentence levels, we build robust, language-agnostic graph models. This allows structured document reasoning to scale across languages with minimal overhead—bridging layout understanding in multilingual corporate or governmental documents. To unify the benefits of graph-based structure and transformer-based context modeling, we propose *Doc2GraphFormer*, a hybrid graph-transformer model that combines structured layout graphs with global attention mechanisms. While transformers like LayoutLMv3 [104] excel at modeling long-range interactions, they operate over sequential tokens and lack explicit structure. Doc2GraphFormer overcomes this by injecting graph priors into attention layers, enabling both local precision and global coherence. It also fuses multimodal features (text, vision, and geometry) into a unified representation, allowing rich layout-aware reasoning.

We evaluate these models on FUNSD [114], XFUND [276], and R-FUND [164], demonstrating strong performance across SER, subgraph clustering, and entity linking tasks. Collectively, the Doc2Graph line of research offers a compelling answer to the question: **Can layout be encoded like language?** Our results affirm that layout, when represented as graphs and learned through structural priors, functions as a powerful syntax for document understanding—cutting across modalities, tasks, and languages.

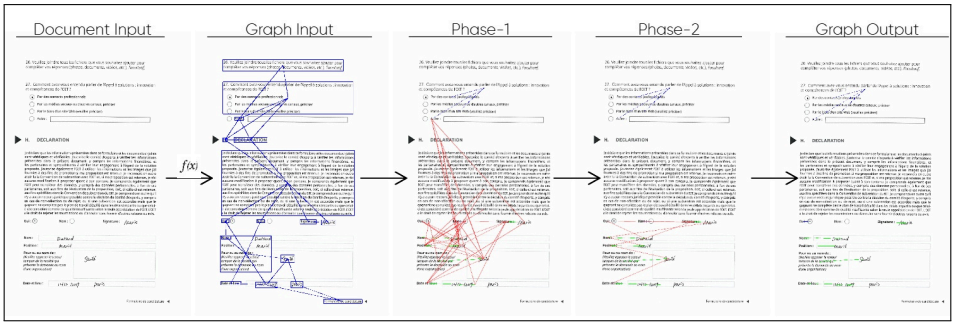


Figure 6.1: **Graph-based document processing with Doc2Graph [78] Framework.** The input document is first transformed into a fully connected graph using K-NN-based spatial proximity. Through a two-phase multimodal message-passing scheme, the model progressively refines entity relationships: Phase 1 contextualizes initial connections, while Phase 2 filters noise and strengthens key semantic links. The resulting graph captures the latent layout structure as a language of entities and relations, supporting accurate semantic entity recognition and relation extraction.

6.2 Related Work

Document layout understanding has long relied on the interplay between spatial structure and textual semantics. In this context, GNNs have emerged as a natural tool for modeling documents, where layout elements are treated as nodes and their spatial or semantic dependencies as edges. Early work in this direction focused on using GNNs for layout-based tasks such as table detection and structure recognition [211, 201], exploiting their ability to encode geometric cues while preserving language independence, a critical advantage in administrative documents where textual content is often sensitive [212, 22]. For example, table extraction in invoices [211] demonstrated how layout alone can be a sufficient modality for reliable parsing. Subsequent research extended this paradigm to more general document understanding tasks. Notably, the FUNSD benchmark [114] inspired graph-based form understanding methods that grouped and labeled word entities using k-NN-based edge construction over bounding boxes and word embeddings [131]. However, these early models lacked visual grounding, limiting their sensitivity to visual structure. The FUDGE framework [54] addressed this gap by combining CNN-based visual relationship detection [52] with a GCN backbone, leading to improved key-value extraction.

The Doc2Graph framework [78] pushed this idea further, proposing a unified, task-agnostic GNN that simultaneously tackled semantic entity recognition (SER) and relation extraction (RE) via joint node and edge classification as shown in Figure 6.1. Yet, Doc2Graph remained restricted to monolingual settings, limiting its utility in real-world multilingual applications. This challenge was addressed in Doc2Graph-X [183], where multilingual embeddings were integrated to extend the graph reasoning frame-

work across languages, enabling robust cross-lingual parsing with minimal parameters. GeoContrastNet [22] further introduced a contrastive learning objective to align semantic and geometric layouts, to strengthen layout-aware representation learning. In parallel, hybrid architectures began to emerge. While lightweight models like GLAM [253] framed layout analysis as a graph segmentation problem using compact GNNs, pure transformer models such as StrucTexT[157], UDOP [238], and LiLT [252] introduced structural and multilingual pretraining strategies to capture broader context. A recent effort by Le et al. [164] proposed a unified pipeline for line extraction, grouping, and linking to address multi-line entities—a common challenge in real-world forms.

Building on these foundations, this chapter introduces a **graph-transformer hybrid** that explicitly models the language of layout. As illustrated in Figure 6.1, we convert each document into a graph and refine its structure in a two-phase message-passing process, guided by multimodal fusion and task-specific decoding heads. This approach helps to bridge local graph reasoning and global self-attention to dynamically learn meaningful relationships across structured documents. Our contributions offer a lightweight yet effective solution for multilingual layout understanding, demonstrating strong performance on FUNSD and related benchmarks.

6.3 Graph-Augmented Attention Modeling

In line with the thesis theme "Layout as Language" , we introduce `Doc2GraphFormer`, a hybrid framework that combines the structural expressiveness of graph representations with the contextual power of transformer-based attention for structured document understanding. Vanilla GNNs excel at modeling local structural dependencies but often lack global reasoning capabilities, while transformers offer broad contextualization but typically operate over sequential token inputs, ignoring explicit document layout. Doc2GraphFormer bridges this gap by treating documents as structured graphs and learning to attend across layout-guided entity relationships, as in Figure 6.2.

6.3.1 Multimodal Graph Representation of Documents

At the heart of this framework lies a unified graph-based representation of documents. We define each document as an undirected graph $G = (V, E)$, where:

- V denotes the set of nodes, each representing a semantically meaningful entity—such as a word, phrase, or layout block.
- E denotes the set of edges, capturing the structural or semantic relationships between node pairs.

Each node feature vector $\mathbf{h}_i$ is a multimodal representation composed of four key components: textual embeddings $\mathbf{t}_i$, visual descriptors $\mathbf{v}_i$, layout encodings $\mathbf{l}_i$, and geometry-

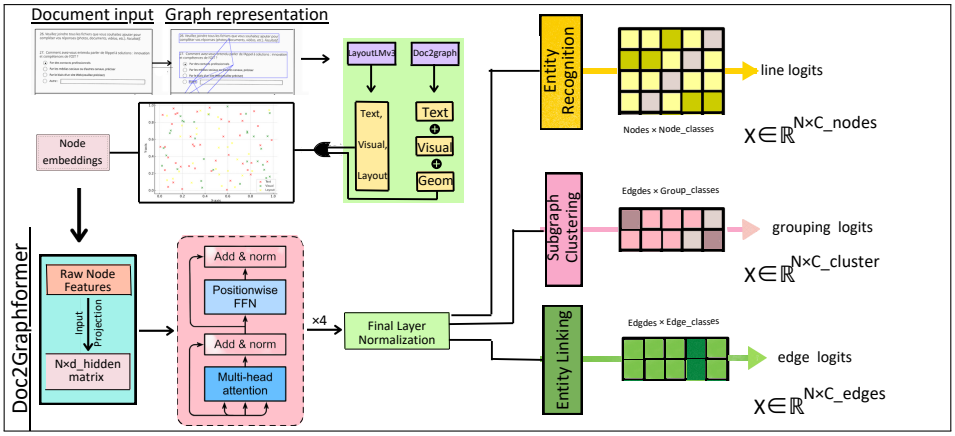


Figure 6.2: **Doc2GraphFormer architecture for structured document understanding.** The pipeline encodes document elements as multimodal graph nodes with features from LayoutLMv3 [104] and Doc2Graph [78] encoders. A graph-transformer module refines node interactions via self-attention. Task-specific heads perform Semantic Entity Recognition, Subgraph Clustering, and Entity Linking, enabling robust layout-aware reasoning across diverse document types.

aware features $\mathbf{g}_i$. The textual component $\mathbf{t}_i$ is derived from pre-trained language models such as SBERT or LayoutLMv3, capturing the semantic context of the document entity. The visual component $\mathbf{v}_i$ incorporates appearance-level descriptors extracted from region-based features, such as U-Net-derived maps or visual tokens. Spatial positioning is encoded through $\mathbf{l}_i$, which represents the 2D coordinates of bounding boxes and integrates contextual layout information using models like LayoutLMv3. Finally, $\mathbf{g}_i$ embeds relative geometric cues including distances and angular relationships using a polar coordinate formulation inspired by Doc2Graph [78]. Each node $v_i \in V$ is enriched with a multimodal feature vector that integrates textual content, visual appearance, and geometric layout as in eq. 6.1:

$$\mathbf{h}_i = [\mathbf{t}_i; \mathbf{v}_i; \mathbf{l}_i; \mathbf{g}_i] \tag{6.1}$$

This multimodal graph representation forms the backbone of the Doc2GraphFormer pipeline. By encoding nodes with rich cross-modal features and defining graph-based structural priors, we enable robust downstream reasoning for semantic entity recognition, inter-entity relationship extraction, and subgraph-level grouping—treating layout not as mere metadata, but as a compositional language of structure.

6.3.2 Graph Construction and Attention Masking

In `Doc2GraphFormer`, we model the document as a fully connected graph $G = (V, E)$, where each node is initially connected to all other nodes. Instead of relying on static graph construction heuristics such as k-nearest neighbors (KNN), the model dynamically learns inter-entity relationships through a structure-aware self-attention mechanism. This design enables the network to emphasize semantically and spatially meaningful connections, while suppressing irrelevant links, guided by the learned layout-aware attention patterns.

To encode this inductive bias, we define an adaptive attention mask:

$$\mathbf{A}_{ij} = \begin{cases} \text{softmax}(\mathbf{Q}_i \mathbf{K}_j^T) & \text{if } (i, j) \in E \\ 0 & \text{otherwise} \end{cases} \tag{6.2}$$

where $\mathbf{Q}_i$ and $\mathbf{K}_j$ are the query and key vectors of node i and j, respectively, derived from the Graphformer's self-attention layers. This formulation enables dynamic pruning and refinement of edge relevance during training, eliminating manual connectivity rules while preserving document layout priors.

6.3.3 Graphformer-Based Feature Processing

At the core of `Doc2GraphFormer` lies a Graphformer encoder that blends graph-based structure with transformer attention. Each node embedding is iteratively updated using multi-head self-attention and position-wise feedforward transformations, allowing the model to reason over both local structure and long-range dependencies:

$$\mathbf{h}_i^{(l+1)} = \text{LayerNorm}\left(\mathbf{h}_i^{(l)} + \text{FFN}\left(\text{MultiHead}(\mathbf{h}_i^{(l)})\right)\right) \tag{6.3}$$

where:

- **MultiHead** denotes the multi-head self-attention function,
- **FFN** is a feedforward network shared across positions,
- **LayerNorm** ensures stable updates.

This hybrid formulation surpasses GNNs that rely solely on localized message passing by enabling global layout-aware reasoning through deep attention mechanisms.

6.3.4 Task-Specific Heads

To support structured document understanding, `Doc2GraphFormer` includes three task-specific prediction heads that operate on the shared graph-augmented node embed-

dings: *Semantic Entity Recognition (SER)*, *Subgraph Clustering*, and *Relation Extraction (RE)*.

Semantic Entity Recognition (SER). For each node $v_i \in V$, we compute the predicted entity label $\hat{y}_i$ as shown in eq. 6.4:

$$\hat{y}_i = \text{softmax}(W_{\text{SER}}\mathbf{h}_i + b_{\text{SER}}) \tag{6.4}$$

where $W_{\text{SER}} \in \mathbb{R}^{d_{\text{hidden}} \times C_{\text{node}}}$ and $b_{\text{SER}} \in \mathbb{R}^{C_{\text{node}}}$ are learnable parameters, and C_{node} is the number of entity classes. The loss function for SER is the standard cross-entropy loss as shown in eq. 6.5:

$$\mathscr{L}_{\text{SER}} = -\sum_{i \in V} y_i \log \hat{y}_i \tag{6.5}$$

Subgraph Clustering (Entity Grouping). Structured documents often contain logically related entities—such as multi-line key-value pairs—that are not explicitly linked but share semantic or spatial proximity. To capture such latent relationships, our model includes a subgraph clustering head that predicts a grouping score for each edge $(i, j) \in E$. The edge-wise binary grouping score is computed as:

$$z_{g,ij} = \text{ReLU}\left(W_{\text{group}_1}\begin{bmatrix}\mathbf{h}_i \\ \mathbf{h}_j\end{bmatrix} + b_{\text{group}_1}\right) \tag{6.6}$$

$$\hat{g}_{ij} = \sigma(W_{\text{group}_2} z_{g,ij} + b_{\text{group}_2}) \tag{6.7}$$

where $W_{\text{group}_1} \in \mathbb{R}^{2d_{\text{hidden}} \times d_{\text{hidden}}}$ projects the concatenated node embeddings into a hidden representation, $W_{\text{group}_2} \in \mathbb{R}^{d_{\text{hidden}} \times 2}$ maps the hidden features to a binary classification space, and σ denotes the sigmoid activation function used to predict the grouping probability. The binary cross-entropy loss for subgraph clustering is defined as:

$$\mathscr{L}_{\text{Cluster}} = -\sum_{(i,j) \in E} \left[g_{ij} \log \hat{g}_{ij} + (1 - g_{ij}) \log(1 - \hat{g}_{ij}) \right] \tag{6.8}$$

Relation Extraction (Entity Linking). In many structured document tasks, it is essential to explicitly identify which entities are semantically linked, such as matching a "Total" label to its corresponding monetary value. For this purpose, Doc2GraphFormer includes a relation extraction head that predicts whether an edge (i, j) represents a valid entity relationship. The relation prediction follows a similar architecture:

$$z_{r,ij} = \text{ReLU}\left(W_{\text{rel}_1}\begin{bmatrix}\mathbf{h}_i \\ \mathbf{h}_j\end{bmatrix} + b_{\text{rel}_1}\right) \tag{6.9}$$

$$\hat{r}_{ij} = \sigma(W_{\text{rel}_2} z_{r,ij} + b_{\text{rel}_2}) \tag{6.10}$$

where $W_{\text{group}_1} \in \mathbb{R}^{2d_{\text{hidden}} \times d_{\text{hidden}}}$ projects the concatenated node embeddings into a hidden representation, $W_{\text{group}_2} \in \mathbb{R}^{d_{\text{hidden}} \times 2}$ maps the hidden features to a binary classification space, and σ denotes the sigmoid activation function used to predict the grouping probability. The loss function for the RE task is also defined via binary cross-entropy:

$$\mathscr{L}_{\text{RE}} = -\sum_{(i,j)\in E} \left[r_{ij} \log \hat{r}_{ij} + (1 - r_{ij}) \log(1 - \hat{r}_{ij})\right] \tag{6.11}$$

6.3.5 Final Learning Objective

The overall optimization objective of Doc2GraphFormer jointly trains all three heads through a multi-task loss:

$$\mathscr{L}_{\text{total}} = \lambda_1 \mathscr{L}_{\text{SER}} + \lambda_2 \mathscr{L}_{\text{Cluster}} + \lambda_3 \mathscr{L}_{\text{RE}} \tag{6.12}$$

where $\lambda_1, \lambda_2, \lambda_3$ are scalar hyperparameters used to balance the contribution of each task to the final loss.

By integrating and combining structural entity classification, grouping, and linking in a unified framework, Doc2GraphFormer mimics the thesis vision of treating document layout not merely as spatial metadata but as an expressive and compositional language for structured reasoning.

6.4 Experimental Validation

This section presents a comprehensive empirical evaluation of the `Doc2GraphFormer` framework, focusing on its effectiveness in two core structured document understanding tasks: *Semantic Entity Recognition* (SER) and *Relation Extraction* (RE). We benchmark our model against a range of state-of-the-art baselines, perform systematic ablation studies to analyze the contribution of individual design components, and provide both quantitative and qualitative analyses to validate the model's performance.

Table 6.1: **Comparison of Doc2GraphFormer with state-of-the-art models** for Semantic Entity Recognition (SER) and Relation Extraction (RE). The table compares modality usage (T = Text, V = Visual, G = Geometric), architectural design (Graph-based vs. Transformer-based), and model size (in millions of parameters). Best results are highlighted in **bold**.

Model	Modalities	Graph	Transformer	SER (↑)	RE (↑)	# Params (M)
BROS [101]	T + V	✗	✓	0.8121	0.6696	138
LayoutLM [274]	T + V	✗	✓	0.7895	0.4281	343
FUNSD Baseline [114]	T + G	✓	✗	0.5700	0.0400	–
FUDGE [54]	V + G	✓	✗	0.6507	0.5241	12
Doc2Graph [78]	T + G + V	✓	✗	0.8225	0.5336	6.2
GeoContrastNet [22]	G + V	✓	✗	0.6476	0.3245	14
Doc2GraphFormer	T + G + V	✓	✓	**0.8439**	0.5548	**3.62**
Doc2GraphFormer + GL	T + G + V	✓	✓	**0.8617**	**0.5548**	**3.62**

6.4.1 Datasets and Evaluation Metrics

To ensure a rigorous and fair evaluation, experiments are conducted on two widely adopted benchmarks: `FUNSD`[114] and `XFUND`[276]. The FUNSD dataset comprises English-language scanned administrative forms annotated with semantic entities and their interrelations. XFUND extends this setting to a multilingual context, covering additional languages including Chinese, French, Japanese, German, Italian, Spanish, and Portuguese. Both datasets offer rich annotations at the entity and relationship levels, making them ideal for assessing structured document parsing capabilities.

Model performance is measured using the *micro-averaged F1 score*, which provides a balanced assessment of precision and recall. For SER, this metric quantifies the accuracy of entity detection and classification, while for RE, it reflects the correctness of predicted links between related entities. In all experiments, the core Graphformer backbone remains fixed. We evaluate multiple configurations of Doc2GraphFormer by varying the modality-specific encoders used for input features namely, Sentence-BERT (SBERT) [209] and LayoutLMv3 [104] to assess the impact of different representation strategies.

6.4.2 Comparison with State-of-the-Art Methods

Table 6.1 reports the comparative performance of Doc2GraphFormer against leading transformer-based and graph-based document understanding models. Despite its architecture with only **3.62M parameters**, Doc2GraphFormer achieves competitive results across both SER and RE tasks. Specifically, it surpasses heavy-weight transformer models such as LayoutLM [274] (343M parameters) and BROS [101] (138M parameters) on SER and performs comparably on RE.

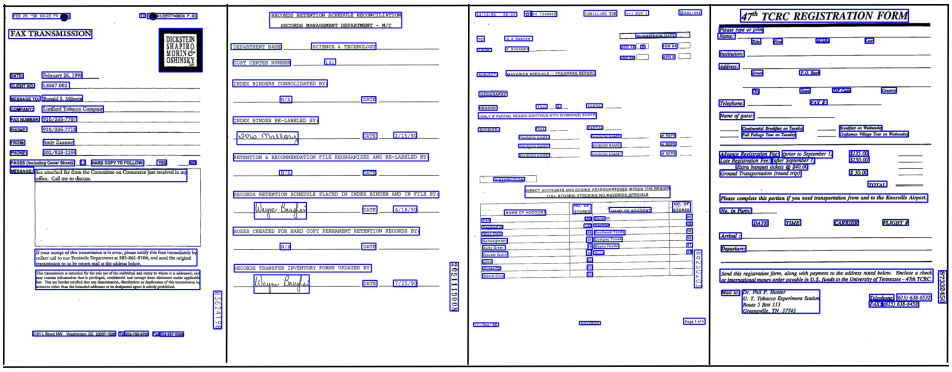


Figure 6.3: **Semantic Entity Recognition (SER) Performance Comparison.** (a) Ground truth annotations with labeled entities. (b) Predicted results from Doc2GraphFormer, where green boxes indicate correctly detected entities, and red boxes highlight incorrect predictions. The model effectively captures structured information but struggles with certain misclassified or missing entities, showcasing areas for improvement in handling complex layouts.

While BROS [101] demonstrates strong SER capabilities (F1 = 0.8121) owing to its robust bidirectional textual encoding, it lacks explicit structural reasoning, which results in weaker RE performance (F1 = 0.6696). Similarly, LayoutLM [274], though multimodal in design, struggles with both SER (0.7895) and RE (0.4281), likely due to its limited incorporation of graph-based relational modeling. Graph-based approaches such as FUDGE [54], Doc2Graph [78] demonstrate better performance on RE tasks, as they incorporate layout structure explicitly. Among them, Doc2Graph stands out with solid performance (SER = 0.8225, RE = 0.5336) by leveraging text, geometric, and visual cues (T+G+V). We did not include Voutharoja et al.'s method [251] in the SOTA Table 6.1 although achieves the highest RE score (0.8540) but *lacks scalability due to re-*

liance on heuristic rule-based reasoning. GeoContrastNet [22], which integrates graph learning with U-Net-based visual modeling, underperforms on both SER (0.6476) and RE (0.3245), primarily due to the absence of a language modeling component.

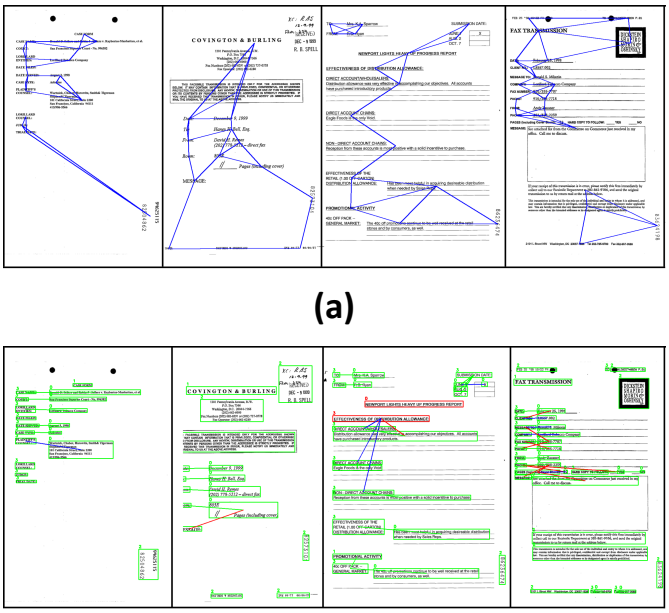


Figure 6.4: **SER and RE Performance Comparison.** (a) Ground truth annotations with entity labels and relationships. (b) Predicted results, where green boxes denote correct entities, red boxes highlight entity classifications, and blue lines represent predicted links. The model successfully captures structured dependencies and entity relationships in documents.

In contrast, Doc2GraphFormer effectively combines textual, visual, geometric, and layout-aware information within a unified graph-transformer framework. It achieves the highest SER score (0.8617) and a strong RE score (0.5548), demonstrating that rich multimodal fusion and graph reasoning can be achieved with minimal parameter overhead. Notably, our model delivers these results with over 90% fewer parameters compared to LayoutLM, underscoring its efficiency and scalability.

6.4.3 Qualitative Analysis

Semantic Entity Recognition (SER). Doc2GraphFormer task results are consistent across handling variations in script structure. As depicted in Figure 6.3, the model preserves document layout integrity, ensuring key entities are recognized correctly even when

embedded within complex tabular structures or very dense text regions. There are minor segmentation errors in where closely spaced characters lead to slight misclassification of tokens. Often overlapping or dense content areas can lead to fragmented entity detections, though overall detection remains robust.

Relation Extraction (RE). Analysis as depicted in Figure 6.4 presents the entity linking (relation extraction) results, where the model predicts semantic relationships between detected entities. We observe that the model correctly identifies key-value pair relationships, even in different writing directions. It can handles nested entities, linking fields correctly within tabular structures. There are minimal false positives, ensuring most connections are meaningful and aligned with document semantics. The edge ambiguity in complex tables, where relationships are less explicit due to layout variations or missing contextual cues. Despite these challenges, Doc2GraphFormer consistently boosts capturing document structure, leveraging graph-based reasoning to improve both entity recognition and relation extraction.

Table 6.2: **Impact of modality combinations on Semantic Entity Recognition (SER) and Relation Extraction (RE)**. Each row shows the effect of including different subsets of input modalities: Text (T), Visual (V), Layout (L), and Geometric (G).

T	V	L	G	SER (↑)	RE (↑)
✗	✗	✗	✓	0.4077	0.0165
✗	✓	✓	✓	0.6589	0.0801
✓	✗	✓	✓	0.8418	0.5109
✓	✓	✗	✓	0.6991	0.1094
✓	✓	✓	✗	0.8366	0.5138
✓	✓	✓	✓	**0.8439**	**0.5548**

6.4.4 Ablation Studies

Effect of Multimodal Layout Encoding. To better understand the contribution of each modality in the Doc2GraphFormer framework, we conducted a detailed ablation study (Table 6.2) by selectively activating combinations of textual (T), visual (V), layout (L), and geometric (G) features. The results reaffirm our central thesis—layout is not merely metadata, but an expressive modality akin to language, capable of guiding structure-aware interpretation.

When geometric features are used in isolation, performance remains minimal (0.4077 SER, 0.0165 RE), underscoring the insufficiency of spatial priors alone. Incorporating visual and layout cues (V+L+G) considerably boosts performance (0.6589 SER, 0.0801 RE), suggesting that visual-spatial regularities help contextualize document regions. Text features, when combined with layout and geometric encodings (T+L+G), significantly enhance semantic entity detection (0.8418 SER) and relation prediction (0.5109 RE), demonstrating the necessity of integrating semantic content with layout-aware

Table 6.3: **Ablation study on the contribution of edge weights and attention masks within the graph-based attention module**. SER and RE represent the F1 scores for Semantic Entity Recognition and Relation Extraction respectively.

Edge Weights	Attention Mask	SER	RE
✗	✓	0.8418	0.5022
✓	✗	0.8396	0.5046
✓	✓	**0.8439**	**0.5548**

Note: ✓ indicates inclusion of component; ✗indicates exclusion.

attention. Interestingly, while the combination T+V+G slightly lowers SER (0.8366), it improves RE (0.5138), reinforcing the intuition that geometric features act as relational syntax that supports linking semantically distant entities. Ultimately, the full multimodal configuration (T+V+L+G) achieves the best performance (0.8439 SER, 0.5548 RE), validating that modeling layout as a first-class representational language yields more robust and coherent document understanding.

Impact of Graph Structure: Edge Weights vs. Attention Masks We further assess the influence of graph connectivity mechanisms by analyzing the role of edge weights and attention masks in guiding the message-passing process. Attention masks alone, without explicit edge weighting, yield strong performance (0.8418 SER, 0.5022 RE), indicating that learned attention paths already provide effective soft layout-aware reasoning. Introducing edge weights without attention masks slightly decreases SER (0.8396) but modestly improves RE (0.5046), suggesting that hard-coded spatial priors contribute to relational modeling even when attention is absent. The highest performance (0.8439 SER, 0.5548 RE) is achieved when both edge weights and attention masks are used in tandem, reflecting the synergy between learned attention flows and structure-aware connectivity. This confirms that layout-guided graph augmentation enriches the model's ability to infer complex semantic and relational structures within documents.

Table 6.4: Ablation analysis of task-specific heads in the Doc2GraphFormer architecture. SER and RE represent the F1 scores for Semantic Entity Recognition and Relation Extraction respectively.

Entity Rec.	Subgraph Clus.	Grouping Labels	Entity Link	SER	RE
✓	✗	✗	✓	0.8426	0.5109
✓	✓	✗	✗	0.8418	0.5022
✓	✓	✗	✓	0.8439	**0.5548**
✓	✓	✓	✓	**0.8617**	**0.5548**

Note: ✓ = enabled; ✗= disabled. All configurations include the Graph-Transformer encoder.

Contribution of Task-Specific Heads To assess the individual and joint impact of task-

Table 6.5: **Fine-tuning F1 performance on XFUND.** Results shown for Semantic Entity Recognition (SER) and Relation Extraction (RE) after language-specific fine-tuning and testing. SBERT outperforms in SER, while LayoutLMv3 excels in RE.

Task	Fusion Strategy	ZH	FR
SER	SBERT	**70.02**	**78.95**
	LayoutLMv3	65.39	73.75
RE	SBERT	29.18	27.83
	LayoutLMv3	**34.21**	**33.36**

specific supervision in Doc2GraphFormer, we conduct a comprehensive ablation analysis as shown in Table 6.4. The experimental configurations progressively activate the following modules: *Entity Recognition*, *Subgraph Clustering*, *Grouping Labels*, and *Entity Linking*. The baseline setting, which includes only Entity Recognition and Entity Linking, achieves competitive performance (SER: 0.8426, RE: 0.5109), underscoring the effectiveness of the core graph-transformer encoder for entity-level prediction and basic relationship modeling. Adding Subgraph Clustering alone does not significantly improve results (SER: 0.8418, RE: 0.5022), suggesting that clustering without explicit linking or grouping signals provides limited benefit for relational reasoning. In contrast, enabling Entity Linking alongside Subgraph Clustering notably improves the RE score to 0.5548, while slightly enhancing SER (0.8439), indicating that learning explicit pairwise links between entities is essential for accurate document parsing. Finally, when all heads are jointly activated including Grouping Labels (GL), Doc2GraphFormer achieves the highest SER score (0.8617) and sustains strong RE performance (0.5548). This configuration benefits from both fine-grained intra-entity grouping and inter-entity linking, supporting the thesis that document layout can be treated as a structured language, where both compositional grouping and relational semantics are essential for holistic understanding.

Cross-Lingual Generalization on XFUND We further examine the multilingual generalization capabilities of Doc2GraphFormer on the XFUND benchmark across Chinese (ZH) and French (FR), using two distinct embedding strategies—SBERT and LayoutLMv3. As shown in Table 6.5, SBERT achieves higher scores for Semantic Entity Recognition in both languages (ZH: 70.02, FR: 78.95) compared to LayoutLMv3 (ZH: 65.39, FR: 73.75), indicating that its semantically rich embeddings contribute to more accurate entity categorization in multilingual contexts.

However, for Relation Extraction, LayoutLMv3 outperforms SBERT significantly (ZH: 34.21 vs. 29.18, FR: 33.36 vs. 27.83), highlighting its advantage in modeling structural relationships due to its multimodal encoding of text, layout, and visual features. This trade-off underscores a critical insight: while semantically fine-tuned sentence embeddings aid entity classification, layout-aware multimodal features are essential for robust relational inference. These findings reinforce the importance of adaptive fu-

Table 6.6: **Comparison of multimodal fusion strategies.** F1 scores for Semantic Entity Recognition (SER) and Relation Extraction (RE) highlight the effectiveness of LayoutLMv3-based embeddings for document understanding.

Model	**SER** ($F_1 \uparrow$)	**RE** ($F_1 \uparrow$)
Doc2Graph [78]	0.8210	0.2929
$\text{Doc2Graph}_{\text{SBERT}}$	0.8188	0.3250
LayoutLMv3 [104]	**0.8439**	**0.5548**

sion strategies when designing multilingual Document AI systems and suggest potential benefits from hybrid approaches combining SBERT's semantic strength with LayoutLMv3's spatial reasoning.

Multimodal Fusion Strategy Analysis We compare three multimodal fusion strategies in Table 6.6, each representing different balances between semantic, spatial, and visual cues. Doc2GraphFormer with LayoutLMv3-based features achieves the highest performance across both tasks (SER: 0.8439, RE: 0.5548), confirming that rich layout-aware embeddings effectively capture both content and structure in document images. $\text{Doc2Graph}_{\text{SBERT}}$ improves over its original variant in terms of relation prediction (RE: 0.3250 vs. 0.2929), highlighting that semantically rich sentence-level embeddings enhance entity linkage despite weaker spatial modeling. The original Doc2Graph model performs competitively in SER (0.8210) due to its graph-based structure encoding but lags in RE due to its limited contextual scope and reliance on static graph connectivity.

This comparative study affirms that transformer-based multimodal fusion, especially with pre-trained models like LayoutLMv3, substantially benefits both entity and relation modeling, aligning with the thesis vision of interpreting layout as a latent language that governs both content semantics and structural dependencies.

6.4.5 Implementation Details

All experiments were conducted using a single NVIDIA GPU with 24 GB of memory (e.g., RTX 3090 or equivalent). The proposed **Doc2GraphFormer** model is lightweight ($\sim$ 3.62M parameters), allowing it to achieve competitive performance without reliance on large-scale hardware setups. The model was implemented using the `PyTorch` deep learning framework, with supporting libraries including `HuggingFace Transformers` for pre-trained textual encoders (e.g., SBERT, LayoutLMv3) and `Deep Graph Library (DGL)` for graph-based operations. We used the AdamW optimizer with a linear warm-up followed by a cosine decay schedule. Training was performed for 100 epochs with a batch size of 16 and an initial learning rate of 5×10^{-5}, which provided stable and reproducible results across datasets. Notably, the compact design of Doc2GraphFormer enables it to be trained and deployed on CPU-only environments with reasonable efficiency. This makes it a practical choice for real-world document understanding sys-

tems operating under compute-constrained settings.

6.5 Conclusion and Future Work

In this chapter, we presented **Doc2GraphFormer**, a lightweight yet effective hybrid framework that integrates graph-based reasoning with transformer-based attention for structured document understanding. By constructing a fully connected document graph and dynamically learning structural relationships through adaptive attention masking, our model eliminates the reliance on heuristic-based graph construction. We demonstrated how Doc2GraphFormer supports multiple downstream tasks—Semantic Entity Recognition (SER), Subgraph Clustering, and Relation Extraction (RE)—within a unified architecture, enabled by a set of shared node representations and task-specific heads. Extensive experiments on standard benchmarks (FUNSD, XFUND) confirm that our model achieves strong performance across both SER and RE tasks, outperforming several state-of-the-art approaches while maintaining a significantly lower parameter count. Through detailed ablation studies, we analyzed the impact of multimodal feature combinations, graph-based attention mechanisms, and individual task heads, thereby highlighting the interpretability and robustness of the proposed design.

Future Work. While Doc2GraphFormer provides an efficient and scalable solution for document understanding, several avenues remain open for future exploration. These include:

- *Cross-document reasoning*: Extending the model to handle multi-page or multi-document contexts where inter-document links and global structure play a crucial role.
- *Graph pre-training*: Incorporating pretext tasks or unsupervised objectives for pre-training the graph structure and node embeddings on large unlabeled document corpora.
- *Knowledge-injected decoding*: Augmenting the relation extraction head with external knowledge graphs or ontologies to guide more precise entity linking and structured generation.
- *Resource-constrained deployment*: Further optimizing the architecture for deployment on edge devices or integrating with low-latency inference pipelines in real-world document processing systems.

In summary, Doc2GraphFormer bridges the gap between semantic content and layout structure in documents through a graph-augmented transformer framework, laying the groundwork for future models that require both accuracy and efficiency in complex document understanding tasks.

Chapter 7

Self-Supervised Visual Representation Learning for Document Layouts

The eye is the most refined of our senses.
It is the gateway to language, geometry, and design.
– Johannes Itten

This chapter explores the potential of self-supervised learning for document layout understanding, introducing a purely vision-based framework named SelfDocSeg. In contrast to approaches that rely heavily on annotated labels or multimodal cues from optical character recognition (OCR), SelfDocSeg investigates whether meaningful document structure can be learned directly from raw images. The framework leverages synthetically generated layout masks to guide self-distillation through a Bootstrap Your Own Latent (BYOL) formulation. By approximating layout as a latent visual grammar—akin to how language models treat syntax, it enables the encoder to establish both object localization and region-level representation. Empirical results across diverse datasets demonstrate that the learned representations are not only data-efficient but also generalizable, outperforming several baselines in downstream segmentation tasks.

7.1 Introduction

In the evolving landscape of intelligent document processing, the ability to interpret and extract structural information from visually complex documents remains a critical

challenge. Document Layout Analysis (DLA), which aims to identify and segment semantically meaningful regions—such as text blocks, tables, figures, and headings—has long served as a foundational task in document understanding systems [176, 26]. While deep learning has driven significant advances in layout segmentation, the vast majority of high-performing methods rely on large-scale annotated datasets [13] or auxiliary cues derived from OCR systems [104], limiting their scalability in real-world scenarios where labeled data is scarce or costly to obtain.

The rapid expansion of digital and scanned documents from diverse sources, ranging from invoices and legal contracts to scientific articles and historical manuscripts, has outpaced the availability of ground-truth annotations. This growing annotation bottleneck has prompted increased interest in self-supervised and weakly supervised learning paradigms [15]. However, in the domain of document segmentation, most existing self-supervised strategies incorporate text-based priors [82, 6, 238] or leverage synthetic layouts [25] tied to pre-trained OCR pipelines, thereby diminishing the independence and generality of the visual representation itself.

In this context, we explore an alternative hypothesis: **can the spatial structure of document layouts be modeled purely from visual cues, without the need for textual supervision?** More specifically, we ask whether layout can be treated analogously to language—as a compositional system of spatial arrangements and alignment rules—such that a model trained via self-distillation can internalize both the semantics and geometry of document regions. To address this, we introduce `SelfDocSeg`, a self-supervised framework for layout-aware representation learning. Built on the BYOL (Bootstrap Your Own Latent) paradigm [81], our method pre-trains an image encoder using augmented views of a document and synthetically generated layout masks derived from classical image processing techniques. These layout masks serve as pseudo-labels that guide both representation learning and object localization in a fully unsupervised fashion. Unlike contrastive methods [11], SelfDocSeg does not require negative samples and avoids dependence on OCR features, making it both lightweight and adaptable to various document domains. (See Figure 7.1)

The main contributions of this chapter are as follows: (i) We propose *SelfDocSeg*, a self-supervised, vision-only framework that learns document layout representations without requiring any textual or annotated supervision. The method is built on a BYOL-style self-distillation backbone, adapted to handle multiple layout objects per image. (ii) We introduce a novel *Layout Mask Generation* pipeline, which derives approximate structural masks from raw document images using classical image processing operations. (iii) We design a *dual-objective training strategy* combining spatial layout prediction and representation alignment, which allows the encoder to learn both where and what the layout objects are, in the absence of human-labeled data. (iv) We demonstrate the *generalizability of our learned representations* across multiple benchmark datasets through downstream fine-tuning, achieving competitive results compared to supervised and multimodal self-supervised baselines, with significantly less training data.

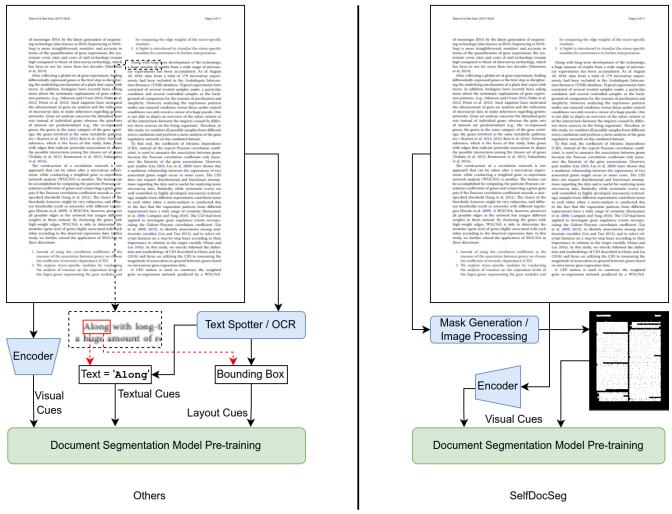


Figure 7.1: **Comparison of SelfDocSeg with existing pre-training methods.** Vanilla self-supervised document segmentation pipelines (left) rely heavily on multimodal cues derived from OCR systems, including text tokens and bounding box layout information. These signals are fused with visual features to guide representation learning. In contrast, SelfDocSeg (right) avoids any textual or OCR-derived supervision and employs classical image processing techniques to generate approximate layout masks directly from the document image using self-distillation.

7.2 Related Work

In this section, we review relevant literature along three major axes: self-supervised learning for visual representation, methods for document understanding, and the evolution of document layout analysis techniques. These form the foundation upon which SelfDocSeg is situated.

Self-Supervised Learning in Computer Vision. Self-supervised learning has emerged as a powerful alternative to supervised representation learning, particularly in domains where labeled data is limited. Early approaches such as MoCo [94] and SimCLR [39] popularized contrastive learning by maximizing agreement between augmented views of the same image. Subsequent models such as SwAV [33] and DINO [34] extended these ideas to clustering-based and vision transformer frameworks. In contrast to contrastive methods, BYOL [81] and SimSiam [41] introduced self-distillation strategies that eliminate the need for negative samples. More recently, masked autoencoders (MAE) [93, 231] and BEiT [16] have shown strong performance by reconstructing masked portions of the input, drawing inspiration from language modeling.

Despite significant success in natural images, the adaptation of these methods to object detection—and specifically to document layout segmentation—has been limited.

Only a handful of works such as UP-DETR [48] and DETReg [17] have extended self-supervised learning to detection tasks. However, these approaches are primarily designed for natural scenes and do not exploit the structured regularities inherent in document layouts. Our work contributes to this gap by introducing a self-supervised method tailored for layout-aware document representation learning, without relying on textual priors or bounding box annotations.

Document Understanding Systems. Document understanding (DU) encompasses a broad range of tasks, including key information extraction [113], classification [91], question answering [180], and machine reading comprehension [236], especially over visually rich documents (VRDs). Recent approaches have emphasized multimodal representation learning that combines visual appearance, textual content, and spatial layout. Models such as LayoutLMv3 [104], DocFormer [6], and UDoc [82] use OCR-derived token embeddings alongside image patches to pre-train large transformers for document tasks. Alternatively, methods like Donut [129] and Dessurt [53] avoid explicit OCR by leveraging synthetic document generation pipelines [27] and image-to-sequence modeling [129]. While effective, these approaches typically require extensive computational resources and pre-existing OCR systems, which may not generalize well to noisy or multilingual domains. In contrast, our proposed method focuses purely on visual signals, demonstrating that spatial layout can be internalized through self-supervised representation learning, without reliance on OCR, text extraction, or token classification.

Document Layout Analysis Document layout analysis (DLA) plays a central role in structuring unstructured documents by identifying semantic regions such as headers, paragraphs, tables, or figures. Traditionally, DLA was addressed using heuristic or rule-based methods [70, 1]. The advent of deep learning enabled convolutional neural networks (CNNs) to replace handcrafted features, leading to more robust segmentation pipelines [220, 225]. The availability of large-scale datasets like PubLayNet [291] and DocLayNet [197] accelerated progress by allowing training of region-based detectors like Mask R-CNN [95], RetinaNet [161], and more recently, transformer-based architectures such as DocSegTr [24] and SwinDocSegmenter [13]. However, these models require substantial annotation efforts and often struggle with generalization in low-resource settings or historical document domains [43]. Self-supervised approaches remain underexplored in layout analysis. Prior works such as LayoutLMv3 [104] and UDoc [82] leverage OCR and textual alignment for pseudo-label generation. In contrast, our method bypasses textual inputs entirely and demonstrates that effective document object representations can be learned from visual structure alone.

7.3 Methodology

SelfDocSeg is a self-supervised vision-based framework for learning document layout representations without requiring labeled annotations or OCR. The pipeline comprises: (i) Pseudo-layout mask generation using classical image processing, (ii) A BYOL-

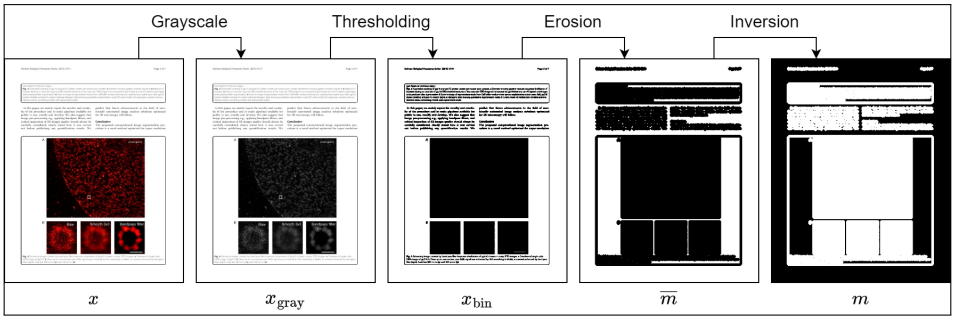


Figure 7.2: **Layout Mask Generation Pipeline.** Starting from an unlabeled document image x, we generate a pseudo-layout mask m through a series of classical image processing steps: grayscale conversion (x_{gray}), thresholding (x_{bin}), morphological erosion ($\overline{m}$), and inversion. The resulting mask m captures layout structure and serves as a self-supervised signal for object localization and representation learning

inspired encoder with dual-branch architecture, (iii) Region-level representation learning via mask pooling, (iv) A layout prediction module for object localization, (v) Downstream fine-tuning with a supervised segmentation model. Figure 7.1 provides a conceptual comparison between SelfDocSeg and multimodal pre-training methods. The full mask generation process is visualized in Figure 7.2.

7.3.1 Problem Formulation

Let $\mathscr{D} = \{x, y\}$ be a dataset of document images $x \in \mathscr{I}^{3\times H\times W}$ and layout annotations $y = \{y_1, \ldots, y_p\}$, where each y_l contains a region mask and class label. In our pre-training phase, we discard y and instead derive a new dataset $\mathscr{D}' = \{x, m\}$, where m is a pseudo-layout mask generated from x (Section 7.3.2). The objective is to pre-train an encoder F_θ using only x and m, such that it learns transferable layout-aware representations for downstream segmentation tasks.

7.3.2 Layout Mask Generation

To generate m from x, we follow these image processing steps:

1. Convert x to grayscale (x_{gray}),
2. Apply global thresholding to obtain a binary image (x_{bin}),
3. Perform erosion to merge visual blobs ($\overline{m}$),
4. Invert $\overline{m}$ to obtain the final layout mask m.

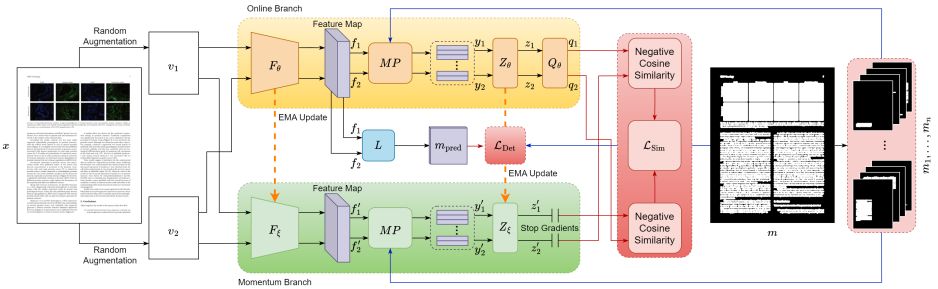


Figure 7.3: **SelfDocSeg Pre-training Framework.** Given an input document image x, two augmented views (v_1, v_2) are processed by an online and a momentum branch. Each branch includes an encoder (F_θ, F_ξ) and mask pooling (MP) guided by the pseudo-mask m and its object-wise splits $m_1, \ldots, m_n$. The online branch also includes projector (Z_θ) and predictor (Q_θ) modules, while the momentum branch uses only Z_ξ. Representations are aligned using similarity loss $\mathscr{L}_{\text{Sim}}$. A layout predictor L learns to localize regions via focal loss $\mathscr{L}_{\text{Det}}$. EMA updates transfer weights from the online to the momentum branch.

This process produces coarse masks that approximate the layout structure without any manual labels (Figure 7.2).

7.3.3 Self-Supervised Pre-Training

Architecture Overview. Given two augmented views v_1 and v_2 of a document image x, we use: (i) An *online branch* with encoder F_θ, projector Z_θ, and predictor Q_θ, (ii) A *momentum branch* with encoder F_ξ and projector Z_ξ, updated via exponential moving average (EMA). Feature maps from both branches are pooled using the layout mask m, and the embeddings are aligned through self-distillation. The architecture is illustrated in Figure 7.3.

Data Augmentation Strategy. Following SimCLR [39], we apply random: (i) Gaussian blurring, (ii) Color jittering, (iii) Color dropping (grayscale) and (iv) Solarization. We exclude cropping and flipping to preserve layout consistency.

Mask Pooling. Let $f \in \mathbb{R}^{c\times h\times w}$ be the feature map output by the encoder. For each layout region m_k, we compute the average pooled representation as shown in eq. 7.1:

$$y^{(k)} = \frac{1}{\sum_{i,j} m_k[i,j]} \sum_{i,j} m_k[i,j] \cdot f[i,j] \tag{7.1}$$

This is done independently on both branches to yield batches of region embeddings.

Representation Learning Objective. Let q be the online branch output (after pre-

dictor), and z' the momentum branch output (after projector). We minimize the cosine distance between corresponding layout embeddings across augmented views as in eq. 7.2:

$$\mathscr{L}_{\text{Sim}} = 4 - 2\left(\frac{\langle q_1, z_2' \rangle}{\|q_1\|_2 \cdot \|z_2'\|_2} + \frac{\langle q_2, z_1' \rangle}{\|q_2\|_2 \cdot \|z_1'\|_2}\right) \tag{7.2}$$

Momentum Update Rule. The momentum encoder parameters ξ are updated from the online encoder θ using exponential moving average as in eq. 7.3:

$$\theta \leftarrow \text{optimizer}(\theta, \nabla_\theta \mathscr{L}_{\text{total}}, \eta), \quad \xi \leftarrow \tau \cdot \xi + (1-\tau) \cdot \theta \tag{7.3}$$

Layout Prediction Module. A layout predictor L receives the feature maps f and outputs a predicted layout mask m_{pred}. It is trained using focal loss:

$$\begin{aligned}\mathscr{L}_{\text{Det}} = -\frac{\alpha}{\sum_{i,j} m[i,j]} \sum_{i,j} \big[& m[i,j](1 - m_{\text{pred}}[i,j])^\gamma \log m_{\text{pred}}[i,j] \\ & + (1 - m[i,j]) m_{\text{pred}}[i,j]^\gamma \log(1 - m_{\text{pred}}[i,j]) \big]\end{aligned} \tag{7.4}$$

Overall Loss. The complete loss function used during pre-training is:

$$\mathscr{L}_{\text{total}} = \mathscr{L}_{\text{Sim}} + \mathscr{L}_{\text{Det}} \tag{7.5}$$

This enables the encoder to jointly learn layout region embeddings and their spatial positions.

7.3.4 Fine-Tuning for Document Layout Segmentation

Once pre-training is complete, the weights of the encoder F_θ are transferred to a Mask R-CNN [95] model equipped with a Feature Pyramid Network (FPN) [160]. The detector is trained on annotated document images for segmentation. The pre-trained features improve performance, especially in low-resource settings, as detailed in the next section.

7.4 Experimental Validation

Datasets. To train and evaluate our proposed SelfDocSeg framework, we leverage a diverse set of document layout analysis datasets. For the self-supervised pre-training

phase, we use only the training split (without annotations) of DocLayNet [197], which comprises 69,375 document images drawn from six different domains and annotated for 11 layout classes. This unlabelled data serves as the basis for extracting pseudo-layout masks and training our encoder as described in Section 7.3. For downstream evaluation of the pre-trained encoder, we employ four datasets: PRImA [**?**] (305 labeled images), Historic Japanese [224] (2,271 documents with 259k labeled layout elements spanning 7 categories), PubLayNet [291] (335k training, 11k validation/test samples with 5 layout categories), and DocLayNet [197] (used here with ground-truth labels).

Implementation Details. We implement SelfDocSeg using the Lightly library built on PyTorch Lightning and PyTorch. Models are trained on NVIDIA RTX A40 GPUs. Pseudo-layout masks are created with OpenCV using a grayscale threshold of 239 (for 8-bit images) and a 5×5 rectangular kernel for erosion. Encoders F_θ and F_ξ are based on ResNet-50 [96], using the final residual block as the feature extractor (output channels $d = 2048$). The projector and predictor MLPs have dimensions $2048 \rightarrow 4096 \rightarrow 256$. The layout predictor L is a 1×1 convolution layer. We train using LARS [281] optimizer with learning rate $\eta = 0.2$, weight decay 5×10^{-4}, cosine decay over 800 epochs, and momentum $\tau = 0.99$ for the target network.

For downstream segmentation, we fine-tune a Mask RCNN [95] (ResNet-50 backbone with FPN [160]) via Detectron2 [269]. Hyperparameters include: 300k iterations, initial learning rate 0.0025, SGD with Nesterov momentum, 64 anchor boxes, batch size 128 per RoI head, NMS threshold 0.4, test threshold 0.6.

7.4.1 Comparative Evaluation

State-of-the-Art Models. Since our contribution focuses on self-supervised pre-training, we evaluate downstream performance after fine-tuning and compare with the following baselines: For *Self-Supervised* model baselines we chose: (i) **LayoutLMv3** [104]: uses masked language/image modeling with OCR supervision and multimodal alignment. (ii) **UDoc** [82]: aligns vision-text-layout embeddings with ROI masking and contrastive loss. (iiI) **DiT** [147]: uses vision-only BEiT-style masked modeling with 42M training samples. And for *Supervised:* baselines, we have: (i) **DocSegTr** [24]: transformer encoder-decoder with convolutional backbone, (ii) **LayoutParser** [225]: CNN-based layout parsing with OCR assistance, (iii) **Biswas et. al.** [26]: modified Mask RCNN with multi-scale features. (iv) **Mask RCNN** [95]: Vanilla instance segmentation model. We also use vanilla BYOL [81] to compare.

Table 7.1 summarizes the mAP performance across datasets. SelfDocSeg outperforms BYOL and vanilla Mask RCNN, while approaching the performance of OCR-guided models despite using only visual cues and far fewer training images.

Table 7.1: Comparison of document object detection performance (mAP) across datasets using different visual (V), layout (L), and textual (T) cues during training.

Type	Method	V	L	T	# Data	DocLayNet	PubLayNet	PRImA	HJ
Supervised	DocSegTr [24]	✓	✗	✗	–	–	90.4	42.5	**83.1**
	LayoutParser [225]	✓	✓	✓	–	–	86.7	**64.7**	81.6
	Biswas *et al.* [26]	✓	✗	✗	–	–	89.3	56.2	82.0
	Mask RCNN [95]	✓	✗	✗	–	72.4	88.6	56.3	80.1
Self-Sup.	LayoutLMv3$_{\text{Base}}$ [104]	✓	✓	✓	11M	–	**95.1**	40.3	82.7
	UDoc [82]	✓	✓	✓	1M	–	93.9	–	–
	DiT$_{\text{Base}}$ [147]	✓	✗	✗	42M	–	93.5	–	–
Proposed	BYOL [81]	✓	✗	✗	81k	63.5	79.0	28.7	59.8
	SelfDocSeg (Ours)	✓	✗	✗	81k	**74.3**	89.2	52.1	78.8

Table 7.2: Semi-supervised fine-tuning on DocLayNet: effect of labeled data quantity.

% Annotations	mAP
10%	41.3
50%	65.1
100%	**74.3**

7.4.2 Performance and Generalization

Our experiments show that SelfDocSeg reaches competitive mAP values compared to models that rely on OCR or large-scale pre-training. Particularly on PRImA and HJ datasets, SelfDocSeg performs comparably or better than supervised transformer models, demonstrating strong generalization. In Fig. 7.4, we visualize qualitative segmentation results. Figure **??** presents qualitative examples illustrating the effectiveness of SelfDocSeg across diverse document types, including *Invoice*, *Advertisement*, *Industrial*, and *Leaflet* layouts. Each row showcases documents with complex structures, varying font styles, dense tabular content, and challenging background artifacts. Despite the absence of manual annotations during training, the model demonstrates precise object localization and layout segmentation. Invoices and industrial documents show consistent detection of tables, headers, and footers. Advertisements, which often contain low-contrast elements and minimal text, are handled effectively through robust visual representations. Leaflets, which blend textual blocks with graphic regions, highlight the model's ability to discern semantically meaningful sections, such as titles and captions. These visual results affirm the model's strong generalization ability across document domains, showcasing the benefits of self-supervised pre-training.

Figure 7.4: Qualitative comparison of predicted layout masks vs. ground-truth on DocLayNet samples (**Left:** predictions, **Right:** GT).

Table 7.3: Ablation study: contribution of loss components in SelfDocSeg pre-training.

Loss Configuration	mAP
w/o $\mathscr{L}_{\text{Sim}}$	39.1
w/o $\mathscr{L}_{\text{Det}}$	69.7
$\mathscr{L}_{\text{Sim}} + \mathscr{L}_{\text{Det}}$	**74.3**

7.4.3 Ablation Analysis

To evaluate the value of our pre-training, we fine-tune using subsets of annotated data. Results in Table 7.2 show graceful degradation even with only 10% of labels, supporting the effectiveness of visual representation learning. In Table 7.3, we analyze each loss term's contribution. The focal loss $\mathscr{L}_{\text{Det}}$ aids object localization, while $\mathscr{L}_{\text{Sim}}$ encourages feature alignment; both are essential for optimal downstream results. Together, these findings confirm that SelfDocSeg can achieve robust layout understanding in a self-supervised manner using only visual data and limited annotations during fine-tuning.

7.5 Conclusion and Future Work

In this chapter, we introduced **SelfDocSeg**, a self-supervised learning framework designed to extract rich visual representations from document images without the need for human-annotated labels. By leveraging a classical image processing pipeline to derive pseudo-layout masks, SelfDocSeg facilitates region-level representation learning that aligns well with document structure. The proposed architecture builds upon a dual-branch BYOL-inspired setup and integrates a novel mask pooling strategy to promote spatially aware feature aggregation. Additionally, a lightweight layout prediction module refines object-level localization using only weak supervision from generated masks.

Through extensive experiments, we demonstrated that SelfDocSeg not only surpasses standard visual pre-training baselines like BYOL but also provides competitive performance compared to supervised counterparts on diverse benchmarks such as DocLayNet, PubLayNet, PRImA, and HJ. The model shows particular promise in domains with limited annotation budgets or highly heterogeneous layouts, validating the importance of structure-aware self-supervision.

Future Directions. While SelfDocSeg makes notable progress in vision-based document representation, several promising avenues remain for exploration:

- **Multimodal Self-Supervision:** Integrating weak textual or OCR-derived cues in a contrastive or masked pretext task to enrich representations.
- **Adaptive Mask Generation:** Learning to generate or refine pseudo-layout masks using data-driven feedback mechanisms instead of fixed morphological rules.
- **Fine-Grained Semantics:** Extending region-based pooling to token-level features, enabling downstream tasks like entity detection or layout-to-structure conversion.
- **Cross-Domain Generalization:** Adapting SelfDocSeg to unseen domains (e.g., historical manuscripts or handwritten forms) via domain-aware pretext tasks.

This concludes **Part II: Representation** of the thesis. Across this section, we explored vision-language and self-supervised strategies to capture the structural and semantic richness of documents. The next part will build upon these representations to tackle **document generation**, where layout-aware decoding becomes central to producing faithful, controllable document images.

Part III

Generation

Chapter 8

DocSynth: Layout-Guided Document Image Synthesis

Form follows function — that has been misunderstood.
Form and function should be one, joined in a spiritual union.
– Steve Jobs

---

*This chapter introduces **DocSynth**, a generative adversarial framework designed to synthesize realistic and diverse document images conditioned solely on layout structures. We formulate the task of document generation as a layout-to-image synthesis problem and present a model that leverages latent object embeddings, spatial reasoning modules, and adversarial training to generate coherent visual documents from structured layout templates. The model is capable of handling complex configurations, supporting layout-based content control, and enabling variability through latent sampling. Although pioneering in its design, this chapter also critically reflects on the inherent limitations of generating documents in pixel space, particularly with respect to semantic coherence and scalability.*

8.1 Introduction

In the broader pursuit of Document AI [46], the ability to not only interpret but also generate structured document images opens up possibilities for layout-driven learning. While the previous parts and chapters in this thesis focused on understanding and representing layout as structure, this chapter pivots toward generative modeling,

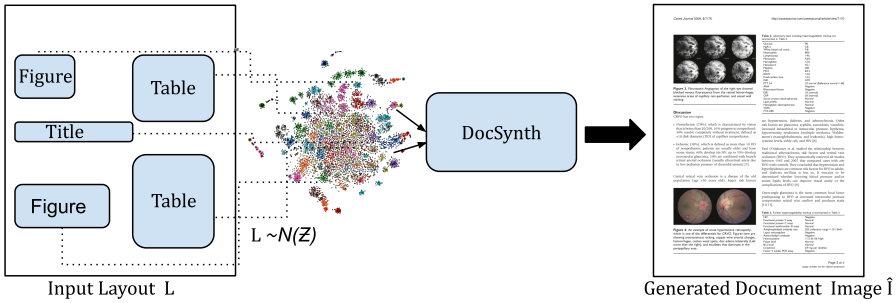


Figure 8.1: **Layout-to-Image Generation with DocSynth:** Given an input layout composed of spatial bounding boxes and class labels, DocSynth generates realistic document images by sampling from latent distributions over both appearance and spatial structure. Multiple diverse samples can be generated per layout.

exploring how layout itself can serve as a guiding signal for document image synthesis. This shift from analysis to synthesis strengthens the thesis's central premise: that layout encodes semantic intent and can be leveraged for both recognition and generation tasks. Modern business workflows increasingly rely on both scanned and digitally-born documents, encompassing a wide range of formats—from invoices and forms to contracts and reports. Extracting information from such documents often requires not only content recognition but also an understanding of their spatial organization. The Office Document Architecture (ODA) framework [102] offers a foundational duality in this regard: *documents are visual images (for rendering and printing) and structured semantic entities (for interpretation)*. Bridging these two views is precisely where layout understanding plays a pivotal role.

However, training models that effectively learn layout-aware reasoning demands large, diverse, and high-quality datasets—resources that are often limited due to annotation costs and privacy concerns. While data augmentation techniques like rotation, scaling, and cropping are widely used, they often fall short in capturing the structural complexity and semantic coherence of real documents. This calls for a more principled solution: **synthetic document generation**, where new samples are created in a controllable, layout-consistent manner.

To this end, we introduce `DocSynth`, a layout-conditioned document image synthesis framework that capitalizes on deep generative models to produce realistic document images guided by a reference layout. As illustrated in Figure 8.1, the model takes an input layout lattice composed of object categories and spatial locations and generates plausible visual renditions, thereby serving both data augmentation and analysis-by-synthesis objectives. This capability proves especially useful in few-shot training scenarios, where real examples are scarce, and synthetic variants can enrich down-

stream tasks such as classification, retrieval, and segmentation. Compared to traditional rendering pipelines, which rely on heuristic composition or manually designed templates [58], DocSynth harnesses neural rendering techniques such as Generative Adversarial Networks (GANs) to learn complex mappings between spatial layouts and their visual realizations. This paradigm shift from rule-based to data-driven synthesis opened the broader evolution in computer vision and reinforced the importance of layout as an actionable prior in document generation.

The contributions of this chapter are the following: (i) We propose **DocSynth**, a novel layout-guided generative model that synthesizes realistic document images from reference layout templates using GANs. (ii) We demonstrate the effectiveness of DocSynth through extensive experiments on the PubLayNet dataset [291], capturing both spatial structure and semantic diversity. (iii) We frame document synthesis as a layout-to-image generation task, introducing a new research direction in Document AI for controllable data augmentation and low-resource learning.

8.2 Related Work

Understanding the structural and spatial organization of documents is a long-standing challenge in Document Analysis and Recognition (DAR). Document layouts encapsulate both physical structure (i.e. the spatial arrangement of elements like text, tables, or graphics) and logical semantics (i.e. the role or meaning these components convey in context e.g., header, signature, logo). Accurate extraction of these layouts is foundational for downstream tasks such as OCR [129], document classification [90], and information extraction [114], as extensively reviewed in [23].

Generative Modeling for Image Synthesis. The advent of Generative Adversarial Networks (GANs) [80] has revolutionized image synthesis, enabling realistic generation across diverse domains—from digits and faces to scenes and handwritten characters. Notably, controlled generation—where specific aspects like object placement or layout are imposed as conditions—has emerged as a promising direction. Early works such as Lake et al. [135] introduced hierarchical generative models that construct characters from strokes, demonstrating compositional learning. Similarly, Layout2Image [288] presented a framework for generating complex scenes from a reference layout of object positions and categories, forming a direct inspiration for our document synthesis task.

Document Layout Generation and Design. Generative modeling for layout structures—particularly in graphics and document design—has also gained momentum. LayoutGAN [146] introduced a GAN-based framework with a wireframe rendering layer, generating layout designs through learned geometric priors. Zheng et al. [290] extended this by incorporating content-aware priors, enabling layout generation conditioned on textual and semantic cues. These approaches, however, operate primarily on abstract layout representations, not full-resolution document images. In response to the limitations of CNN-based decoding, which may ignore low-dimensional geomet-

ric regularities, READ [194] proposed a recursive architecture capable of synthesizing structured 2D layouts with content fidelity. While effective for layout-level generation, it does not translate to pixel-space rendering necessary for tasks like document image classification or visual retrieval. On the other end, GANwriting [123] demonstrated synthetic word-level handwritten image generation, emphasizing how learned structure can facilitate image-level realism. However, their work remains focused on localized text snippets, *lacking support for full-page document synthesis.*

Despite these advances, existing methods remain insufficient for *whole-page document image synthesis* that balances realism, layout fidelity, and semantic diversity. Document images present unique challenges—blending structured graphical layouts with natural language content across heterogeneous templates (e.g., invoices, resumes, reports). Furthermore, the need for controllable generation—particularly from layout specifications—is paramount for scalable data augmentation, semi-supervised learning, and retrieval tasks. In this chapter, we propose DocSynth, a layout-guided document image generation framework that bridges this gap. By synthesizing document images directly from spatial and categorical layout templates, DocSynth offers a milestone step towards fully controllable document synthesis, establishing new avenues for low-resource learning in Document AI.

8.3 The DocSynth Framework

This chapter introduces our layout-driven generative framework for document image synthesis, DocSynth. We begin by formally defining the problem and introducing the key notation. Following that, we detail the proposed architecture, describe each component of the network, outline the training and inference strategies, and conclude with implementation-specific considerations.

8.3.1 Problem Formulation

Let us define X as a fixed-size image canvas (e.g., 128×128) and I as a document image over this canvas. A document layout is represented as $L = \{(\ell_i, \text{bbox}_i)\}_{i=1}^{n}$, where each object instance O_i belongs to a category $\ell_i \in \mathscr{O}$ and is spatially defined by a bounding box $\text{bbox}_i \subset X$. We sample latent appearance features $Z_{obj} = \{\mathbf{z}_{obj_i}\}_{i=1}^{n}$ from a standard Gaussian prior $\mathcal{N}(0,1)$ for each object instance. The goal is to learn a generator function G parameterized by Θ_G that produces a synthetic image $\tilde{I}$ from layout L and the object-wise latent code Z_{obj} as in eq. 8.1:

$$\tilde{I} = G(L, Z_{obj}; \Theta_G) \tag{8.1}$$

The DocSynth model is designed to address **three core challenges**:

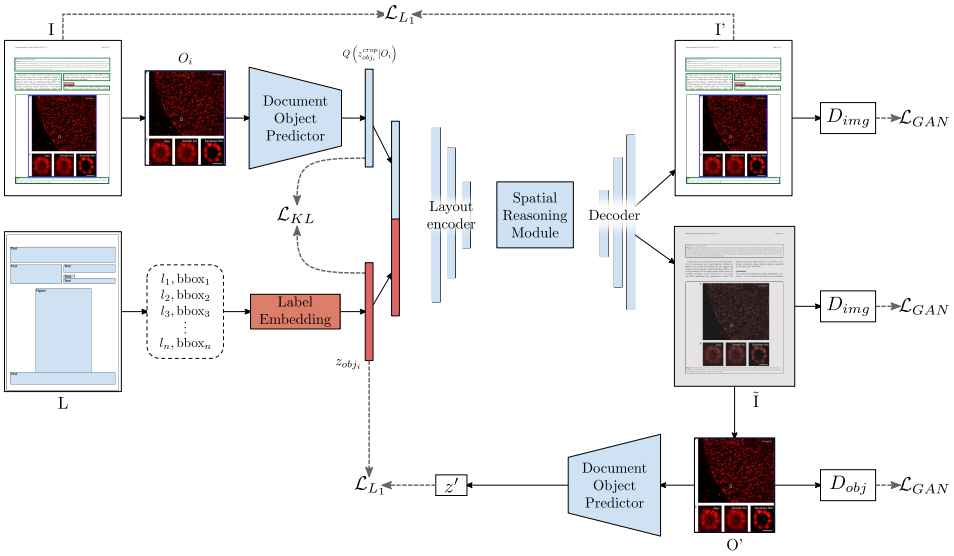


Figure 8.2: **DocSynth Framework Overview:** The model is trained in an adversarial setup with both image- and object-level discriminators. Given a layout with bounding boxes and semantic labels, the generator synthesizes document images guided by spatial configuration and object appearance.

- Can the generator synthesize visually realistic images faithfully reflecting the input layout L?
- Can it generate diverse yet layout-consistent images by varying Z_{obj}?
- Is the model robust to layout alterations, such as modifying object positions or adding/removing elements?

8.3.2 Model Architecture and Training Strategy

Training Phase. As shown in Figure 8.2, training begins with extracting layout annotations L and cropping object instances O_i from ground truth image I. The model embeds object category labels as vectors e_i, and samples appearance latent codes from both a prior $\mathcal{N}(0,1)$ and a posterior $Q(\mathbf{z}_{obj_i}^{crop} \mid O_i)$ predicted by an object encoder E. Two types of synthetic outputs are produced:

1. Reconstructed image I' using posterior samples Z_{obj}^{crop}.
2. Generated image $\tilde{I}$ using randomly sampled Z_{obj}.

A second object encoder E' regresses latent features from generated objects, enforcing alignment between generation and prior encoding. The entire pipeline is optimized

via adversarial learning, using two discriminators (D_{img} and D_{obj}) at the image and object levels, respectively.

Inference Phase. During inference, given a user-defined layout L, the generator samples object appearance codes from $\mathcal{N}(0,1)$ and synthesizes novel document images consistent with the spatial arrangement and object labels.

8.3.3 Architectural Module Description

Object Encoders. The object encoders E and E' extract posterior representations from cropped real and generated object instances. Each encoder outputs the mean and variance vectors for Gaussian sampling and consists of stacked convolutional layers followed by fully connected layers.

Layout Encoding. Each object's layout encoding F_i is constructed by fusing its label embedding e_i, latent code $\mathbf{z}_i$, and bounding box bbox_i. These per-object features are rasterized into feature maps and aggregated by a convolutional layout encoder C.

Spatial Reasoning Module. To model inter-object dependencies, a convolutional LSTM (ConvLSTM) processes the sequence of object feature maps. It outputs a spatially coherent hidden state h that serves as input to the final image decoder.

Image Generator. The decoder K takes the hidden feature map h and reconstructs either the original document image I' or a new variant $\tilde{I}$, depending on the sampled latent codes.

Discriminators Two adversarial discriminators are used: D_{img} evaluates full-page realism, and D_{obj} focuses on individual object quality. The adversarial loss follows the standard GAN objective as in eq. 8.2:

$$\mathcal{L}_{\text{GAN}} = \mathbb{E}_{x\sim p_{\text{real}}}[\log D(x)] + \mathbb{E}_{y\sim p_{\text{fake}}}[\log(1 - D(y))] \tag{8.2}$$

8.3.4 Learning Objectives

The total training loss is a weighted sum of six components:

$$\mathcal{L}_G = \lambda_1 \mathcal{L}_{\text{GAN}}^{\text{img}} + \lambda_2 \mathcal{L}_{\text{GAN}}^{\text{obj}} + \lambda_3 \mathcal{L}_{\text{AC}}^{\text{obj}} + \lambda_4 \mathcal{L}_{\text{KL}} + \lambda_5 \mathcal{L}_1^{\text{img}} + \lambda_6 \mathcal{L}_1^{\text{obj}} \tag{8.3}$$

Where:

- $\mathcal{L}_{\text{GAN}}^{\text{img}}$, $\mathcal{L}_{\text{GAN}}^{\text{obj}}$ are adversarial losses.
- $\mathcal{L}_{\text{KL}}$ ensures regularization of posterior and prior distributions.
- $\mathcal{L}_1^{\text{img}}$, $\mathcal{L}_1^{\text{obj}}$ are pixel-wise reconstruction losses.
- $\mathcal{L}_{\text{AC}}^{\text{obj}}$ is a classification loss to enforce semantic accuracy.

8.3.5 Implementation Details

To stabilize adversarial training, we adopt Spectral Normalization GAN (SN-GAN) [**?**]. Conditional Batch Normalization [**?**] is applied to object encoding layers. The model is implemented in PyTorch and supports image resolutions of 64×64 and 128×128. Hyperparameters are set as follows: $\lambda_1 = 0.01$, $\lambda_2 = 1$, $\lambda_3 = 8$, $\lambda_4 = 1$, $\lambda_5 = 1$, and $\lambda_6 = 1$. Training is conducted using the Adam optimizer [130] with a batch size of 16 over 300,000 iterations. All experiments are reproducible on a single NVIDIA GPU with 24 GB memory.

8.4 Experimental Evaluation

This chapter presents a comprehensive set of experiments designed to evaluate the effectiveness of our proposed *DocSynth* framework for layout-guided document image generation. The experimental analysis comprises qualitative visualizations, quantitative benchmarking using established metrics, and a series of ablation studies to assess the contribution of individual architectural components. All experiments were implemented using the PyTorch library and conducted on a single NVIDIA GPU with 24GB of memory.

8.4.1 Datasets

To validate our approach, we conduct experiments on the **PubLayNet** dataset [291], a large-scale benchmark containing structured document images derived from the PubMed Central digital library. The dataset provides annotated bounding boxes for five key object categories: `text`, `title`, `list`, `table`, and `figure`. For our evaluation, we use the official training split comprising 335,703 images and the validation split with 11,245 images.

8.4.2 Evaluation Metrics

To assess the fidelity and variability of the generated document images, we employ two widely used metrics in generative modeling:

Fréchet Inception Distance (FID) The FID score [99] quantifies the distance between feature distributions of real and generated images in the latent space of an Inception-v3 network [235]. Lower FID values indicate better alignment with real image statistics and more photorealistic outputs.

Diversity Score (LPIPS) To capture perceptual diversity, we use the Learned Perceptual Image Patch Similarity (LPIPS) score [287], computed over pairs of images generated from the same layout. Higher scores indicate greater variation in appearance while

preserving layout structure. We use the AlexNet backbone [134] to extract features for diversity estimation.

8.4.3 Qualitative Results

Visualizing the Synthetic Distribution. We first demonstrate the capability of our model to generate plausible document images across diverse layout patterns. Figure 8.3 shows a 2D t-SNE [246] embedding of synthetic samples, illustrating distinct clusters corresponding to different layout structures. These results highlight our model's ability to learn rich spatial priors and generate visually distinct yet structurally consistent samples.

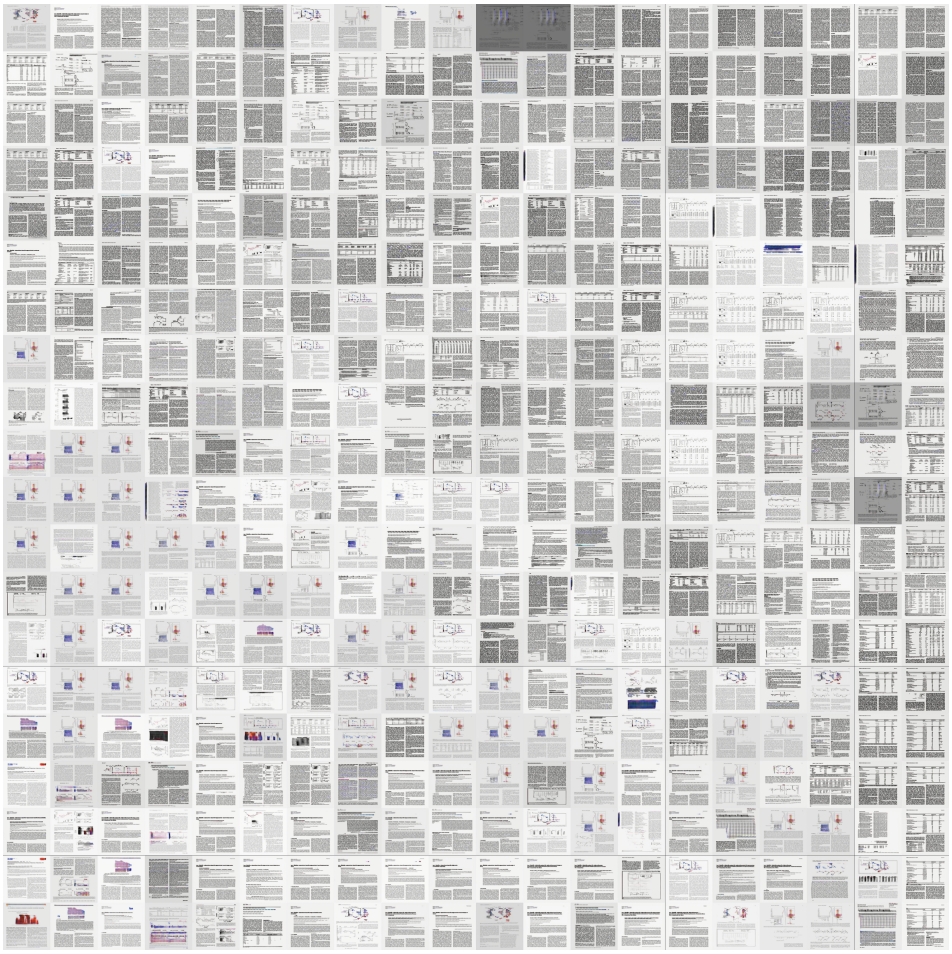


Figure 8.3: t-SNE visualization of the generated synthetic document images

Controllable Synthesis via Layout Conditioning. A major strength of our approach lies in its capacity for layout-conditioned controllable generation. We assess this in two distinct scenarios:

Case 1: Diverse Styles from Fixed Layout. As shown in Figure 8.4, our model generates multiple stylistic variants from a single reference layout while preserving the structure and semantics of layout elements. This supports use cases like dataset augmentation and style transfer.

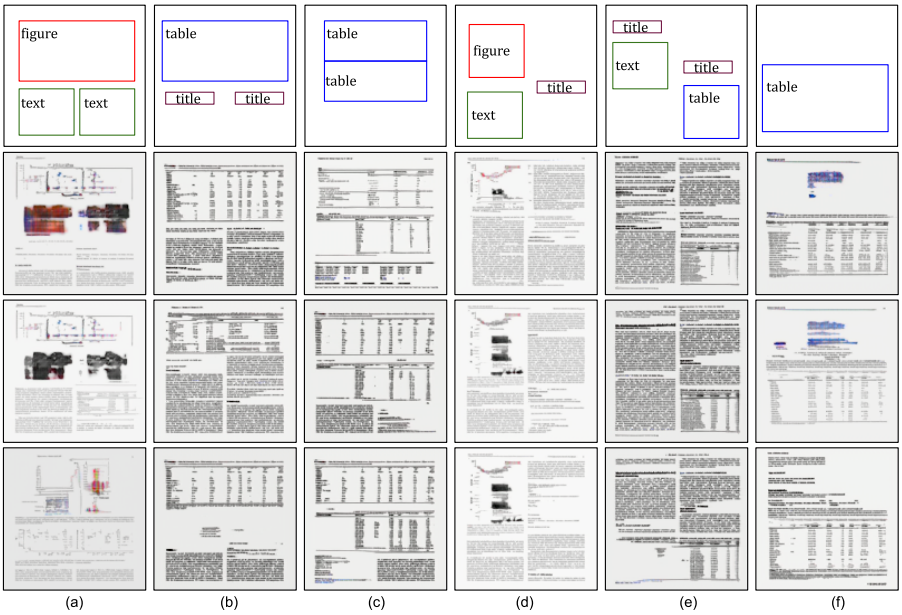


Figure 8.4: Examples of diverse synthesized documents generated from the same layout. The layout structure remains fixed, while the visual style varies across samples.

Case 2: Dynamic Layout Editing. In Figure 8.5, we simulate incremental editing of the layout (adding or removing bounding boxes). The model successfully adapts the generated images in response, maintaining spatial coherence and semantic alignment. This reflects its robustness to layout variations and capacity for layout-guided scene manipulation.

8.5 Quantitative Results

We quantitatively evaluate our generation results using FID and LPIPS-based Diversity Score. Table 8.1 reports scores for both 128×128 and 64×64 resolutions. We observe that generated images achieve comparable FID scores to real samples, while also at-

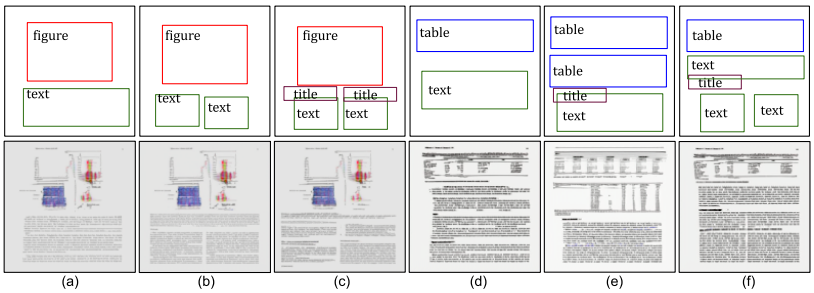


Figure 8.5: Examples of synthesized document images by adding or removing bounding boxes. Top row: incremental addition; Bottom row: object removal.

taining higher diversity, particularly at the 64×64 resolution.

Table 8.1: Performance metrics for real and generated document images

Method	**FID**	**Diversity Score**
Real Images (128×128)	30.23	0.125
DocSynth (128×128)	**33.75**	**0.197**
Real Images (64×64)	25.23	0.115
DocSynth (64×64)	**28.35**	**0.201**

8.6 Ablation Studies

To understand the contribution of architectural choices, we perform controlled ablation experiments focused on the spatial reasoning module of our network. Table 8.2 compares different variants: no LSTM, vanilla LSTM, and conv-LSTM with varying kernel depths k. The results clearly demonstrate the effectiveness of the conv-LSTM structure in preserving layout fidelity and boosting generation quality. Deeper convolutional gates (up to $k = 3$) further improve performance by better capturing spatial dependencies among layout objects.

8.7 Conclusions and Future Directions

This chapter presented **DocSynth**, one of the first comprehensive attempts at *layout-guided synthetic document image generation* at scale. Through a carefully designed

Table 8.2: Impact of spatial reasoning module on FID score

Reasoning Backbone	**FID**
No LSTM	70.61
Vanilla LSTM	75.71
conv-LSTM (k=1)	37.69
conv-LSTM (k=2)	36.42
conv-LSTM (k=3)	**33.75**

generative architecture combining spatial reasoning, conditional priors, and adversarial training, we demonstrated that it is indeed possible to synthesize plausible, diverse, and structurally coherent document images purely from abstract layout representations. Our model enables controllable generation, layout-driven editing, and multimodal sampling—all of which open new possibilities for augmenting datasets, simulating layout scenarios, and training downstream vision models in a low-cost and customizable manner.

Despite its strengths, generating documents directly in the **image space** presents inherent limitations. Visual synthesis is heavily constrained by resolution, difficult to scale to high-fidelity documents, and often fails to generalize across complex domains with diverse font styles, languages, or embedded content (e.g., formulas, tables). Moreover, image-based models do not capture the underlying textual semantics or logical structure of the document, which are essential for downstream tasks like editing, retrieval, or semantic search. In that sense, DocSynth is a *foundational but partial* step toward document generation. It demonstrates the viability of layout-to-image synthesis, but also exposes the bottlenecks of working in pixel space, where learning is limited by visual consistency and lacks interpretability. These limitations motivate a shift in perspective—from visual realism to **structural fidelity**, and from rasterized pixels to **sequential token representations**.

Looking ahead, we build upon the insights from this chapter—particularly the importance of layout priors and object-level semantics—to explore autoregressive models that generate structured documents in a token space. This not only addresses the scalability and fidelity issues faced by image-based methods, but also sets the stage for unifying generation, editing, and understanding in a single language-driven framework. In summary, while DocSynth marks a milestone in layout-to-image synthesis, it also serves as a launching pad for a deeper exploration into *document generation as a language problem*. By blending layout structure with language modeling, the forthcoming chapters chart a path toward models that can *understand*, *generate*, and *edit* documents in a way that is both human-aligned and machine-efficient.

Chapter 9

Towards Autoregressive Vector Document and Sketch Generation

The essence of intelligence is skill in extracting meaning from everyday experience.
– Herbert A. Simon

---

*This chapter presents a paradigm shift in document generation by transitioning from layout-to-image generation to a sequence-based autoregressive modeling. We introduce **DocSynthv2**, a vectorized layout-aware generation model that formulates the document structure comprising both layout and textual elements as a sequential representation. Unlike classical GAN-based methods that operate in the pixel space, our approach models documents as grammars of layout and content, enabling fine-grained control and enhanced generation fidelity. Furthermore, we extend this paradigm with **SketchGPT**, an autoregressive sketch completion model that captures the compositional rules underlying document structure, offering insights into layout grammar learning through **next token prediction**. These models demonstrate robust performance in generating coherent, diverse, and semantically plausible document representations across various document types.*

9.1 Introduction

With growing interest in layout-aware document modeling [108, 133, 86, 107], the need for scalable synthetic document generation has become increasingly relevant. Previous works like DocSynth [27] and SynthTIGER [280] attempted to synthesize document images directly in the pixel space using layout templates. While effective for rendering

style-diverse documents, these approaches are limited by resolution bottlenecks and the inability to preserve high-fidelity textual content.

To overcome these constraints, DocSynthv2 reformulates the problem as a *layout-to-sequence generation task*. Each document is represented as an ordered sequence of layout tokens, object labels, and corresponding textual snippets. By modeling this sequence autoregressively, the system gains fine-grained control over both the structure and content generation processes. The benefits are multifold: (1) the model supports *high-resolution rendering* since it decouples generation from pixel space; (2) it allows partial or constrained generation (e.g., layout completion); and (3) it integrates naturally with language modeling techniques and text conditioning. To benchmark this task, we introduce PubGenNet, a large-scale document generation dataset curated for layout-text pair modeling. The dataset includes diverse domains (scientific, legal, forms) and enables rigorous evaluation for document completion, conditional generation, and data augmentation. DocSynthv2 demonstrates state-of-the-art results on these tasks, offering a simple yet flexible framework that can generate realistic and semantically coherent document layouts.

In parallel to structured documents, *hand-drawn sketches* represent a fascinating form of sequential visual communication. From architecture [55] to electronics [217] and entertainment [20], sketches serve as expressive tools grounded in spatial-temporal order. Unlike static images, sketches are captured as a stream of pen movements, and thus naturally lend themselves to sequence modeling. SketchGPT proposes a unified autoregressive generative model that treats sketches as *visual sentences* composed of discrete strokes. Inspired by the success of GPT-style models in next-token prediction [203, 30], our approach learns to predict the next drawing primitive conditioned on the sequence of prior strokes. Unlike prior models like SketchRNN [87] or SketchBERT [159], SketchGPT generalizes across multiple tasks—sketch generation, completion, and classification—using a single architecture.

To improve generalization and avoid overfitting, we introduce a `stroke-to-primitive abstraction` [2], which discretizes continuous sketch inputs into a compact lexicon of reusable shapes. This enables efficient learning while preserving structural expressiveness. SketchGPT outperforms prior sketch generation models on multiple datasets and demonstrates strong capabilities in downstream applications. The core contributions of this chapter are fourfold: (i) First, we propose DocSynthv2, an autoregressive vector-based generation framework that models documents as unified sequences of layout and textual tokens, enabling flexible and high-resolution document synthesis. (ii) Second, we introduce SketchGPT, a GPT-inspired generative model that captures the sequential structure of hand-drawn sketches through stroke-level abstraction, supporting tasks such as sketch generation, completion, and classification. (iii) Third, we curate a new benchmark dataset, PubGenNet, specifically designed for layout-text modeling and document generation, facilitating consistent evaluation across multiple settings. (iv) Finally, we conduct qualitative and quantitative experiments spanning both document and sketch domains, demonstrating the broad applicability and effectiveness of autoregressive approaches in structured visual content generation.

9.2 Related Work

Document Layout Generation. The task of document layout generation has seen rapid growth, driven by its importance in applications ranging from automated publishing and report creation to responsive web design. Foundational models like *LayoutGAN* [146] and *LayoutVAE* [120] modeled the spatial distribution of 2D objects and synthesized plausible document-like layouts. Building on this, layout-conditioned generation approaches such as [290] enabled controllable synthesis by conditioning on prompts like input categories or exemplar images. Further, `READ` [194] introduced recursive VAE hierarchies to represent document structures, which was later extended with graph autoencoders in `LayoutGMN` [195] for layout sampling under structural constraints. Of particular relevance to our work is the `Layout Transformer` [86], which employs self-attention [249] and a next-element prediction objective to autoregressively generate document layout tokens comprising class labels and bounding box coordinates. Subsequent approaches such as [8] combine VAE objectives with generative transformers to enhance the diversity and fidelity of generated layouts.

Synthetic Document Generation. Alongside layout modeling, the synthesis of full document images from structured layouts has gained popularity in the computer vision community. Layout-conditioned image generators [288, 98, 117] aim to translate layout maps into high-resolution images, emphasizing photorealism and object placement consistency. Specifically within the document domain, *DocSynth* [27] was the first to offer an end-to-end image synthesis pipeline for creating synthetic documents using layout-to-image translation, primarily to support layout analysis tasks [197, 291]. While effective for data augmentation, such pixel-based techniques often yield low-resolution outputs, with limited control over embedded textual content. Our proposed **DocSynthv2** addresses this limitation by shifting from raster-based to vector-based generation, encoding both layout structure and text as unified sequences in an autoregressive modeling paradigm.

Sketch as a Language. Sketches, like textual language, exhibit an inherent structure, composed of sequential strokes that mirror syntactic and semantic units [74, 178]. This analogy has inspired a class of models treating sketch generation as a form of visual language modeling. One early example is *SketchRNN* [87], which introduced a sequence-to-sequence LSTM model [100] to learn dynamic stroke trajectories for sketch synthesis. Expanding this idea, *SketchBERT* [159] adapted the BERT language model [61] to the vector sketch domain, achieving strong results on recognition and retrieval tasks. More recently, *SketchKnitter* [256] employed a denoising diffusion model to simulate human sketching behaviors, capturing the distribution of stroke points over time. Our approach, **SketchGPT**, draws from these works but extends the modeling scope by leveraging GPT-style next-token prediction to unify sketch generation, completion, and classification under a single autoregressive framework.

Sketch Abstraction via Primitives. A critical challenge in sketch modeling is handling variability in stroke styles and densities. Inspired by theories in cognitive science [21], recent works have proposed abstracting continuous sketches into compact, symbolic

primitives. For instance, Alaniz *et al.* [2] proposed a Primitive Matching Network to map freeform strokes to a finite set of canonical shapes using affine transformations. This abstraction improves generalization by discretizing the input space and minimizing overfitting to fine-grained drawing variations. Our model incorporates a similar primitive-based mapping strategy to reduce complexity and enhance training stability in the autoregressive sketch modeling process.

Applications and Datasets. The release of large-scale sketch datasets, such as TU-Berlin [66], Sketchy [218], and QuickDraw [118], has fueled progress in sketch understanding and creative generation. Early efforts in sketch recognition [219, 155] were built on hand-crafted features, later superseded by deep learning models [229, 282] that now rival human performance. Transformer-based generative frameworks [19, 242, 213] have emerged as state-of-the-art for sketch modeling, exploiting learned tokenization schemes to improve interpretability and performance across diverse sketch tasks. Our work contributes to this evolving landscape by offering a flexible and unified transformer-based model that supports a variety of sketch-related objectives while maintaining a lightweight and task-agnostic design.

9.3 The DocSynthv2 Framework

In this section, we detail the proposed methodology for autoregressive document generation using **DocSynthv2**. We begin by formalizing the representation of document elements, followed by a comprehensive explanation of our model design and training objectives.

9.3.1 Document Representation

Each document $\mathscr{D}$ is modeled as a sequence of layout elements, where each element is characterized by a set of attributes. These include its semantic category c (e.g., paragraph, table, figure), spatial position (x, y), size (w, h), and optional textual content t. Inspired by prior works [86, 8, 109], we quantize the continuous attributes into discrete tokens to enable sequence modeling with autoregressive transformers.

We define a document as a sequence of S elements as in eq. 9.1 :

$$\mathscr{D} = (D_1, D_2, \ldots, D_S), \tag{9.1}$$

where each element D_i is a tuple:

$$D_i = \left\{ d_i^k \mid k \in \mathscr{E} \right\}, \tag{9.2}$$

and $\mathscr{E}$ represents the set of all attribute types. This includes discrete tokens for class label, bounding box coordinates, style information (such as font or weight), and optionally, a set of tokens for associated text content.

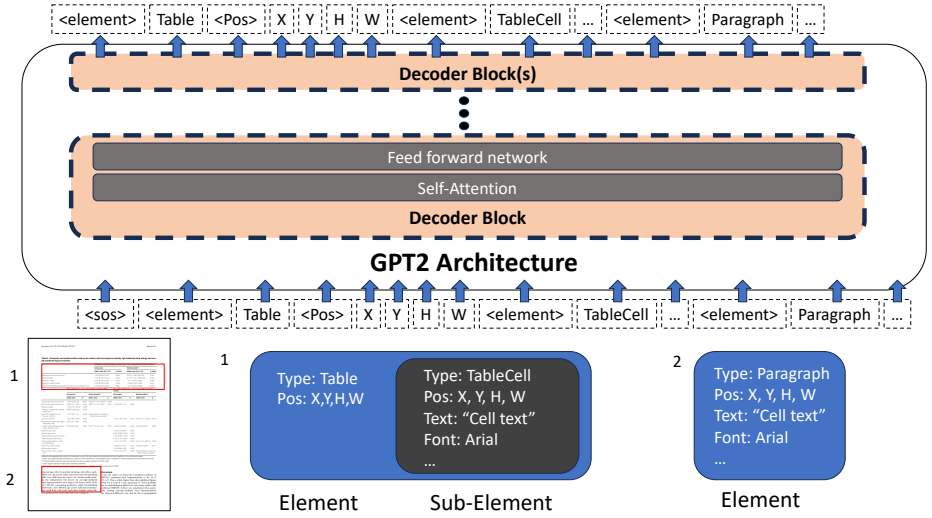


Figure 9.1: **Overall architecture of DocSynthv2, our autoregressive framework for structured document generation.** Each document is represented as a sequential stream of layout-text tokens, encoding hierarchical information from high-level elements (e.g., tables, paragraphs) to nested sub-elements (e.g., table cells). These tokens include type, position (X, Y, H, W), style, and content attributes, which are processed through a stack of GPT2-based decoder blocks with self-attention and feed-forward layers. The model learns to predict the next token conditioned on the prior sequence, capturing both spatial and semantic structure of the document.

To form a full input sequence for the model, we concatenate all element tokens linearly as in eq. 9.3:

$$\mathscr{D} = \langle \text{sos} \rangle \, c_1 \, x_1 \, y_1 \, w_1 \, h_1 \, t_1 \, \ldots \, c_S \, x_S \, y_S \, w_S \, h_S \, t_S \, \langle \text{eos} \rangle \,, \tag{9.3}$$

where ⟨sos⟩ and ⟨eos⟩ are special start-of-sequence and end-of-sequence tokens, respectively. For missing fields (e.g., font in non-text elements), we insert a special `[NULL]` token.

9.3.2 Discrete Modeling and Sequence Learning

Each document is tokenized into a sequence of length m, with each token represented as a latent vector $\boldsymbol{\theta}_j$, where $j = 1, \ldots, m$. The overall generative process is formulated as a chain of conditional probabilities using the autoregressive factorization as in eq. 9.4:

$$p(\boldsymbol{\theta}_{1:m}) = \prod_{j=1}^{m} p(\boldsymbol{\theta}_j \mid \boldsymbol{\theta}_{1:j-1}), \tag{9.4}$$

This formulation enables the model to generate one token at a time, conditioned on all previously generated tokens, capturing both syntactic and layout-dependent dependencies across elements.

9.3.3 Model Architecture

The DocSynthv2 architecture is based on the GPT2 decoder stack [204], tailored to capture the layout-content interplay. It comprises a stack of N masked transformer blocks, each containing:

- A masked multi-head self-attention (MHA) mechanism to preserve autoregressive ordering.
- A position-wise feed-forward network (FFN) to enhance token representation.
- Residual connections and layer normalization to facilitate stable training.

The input to the model is a tokenized sequence of the type shown in Equation 9.3. The model predicts the next token $\boldsymbol{\theta}_j$ at each timestep j using the context $\boldsymbol{\theta}_{1:j-1}$. During training, the model is exposed to full ground truth sequences and optimized using teacher forcing. At inference time, it generates new layouts auto-regressively, conditioned either on a partial layout or a prompt of class tokens (e.g., starting with a Table or Title block).

9.3.4 Learning Objectives

To train DocSynthv2, we use a combination of categorical cross-entropy losses for discrete attributes and optionally, variational regularization for smoother token distribution. The training objective minimizes the negative log-likelihood over the token sequence as in eq. 9.5:

$$\mathscr{L}_{\text{NLL}} = -\sum_{j=1}^{m} \log p_\theta \left(\boldsymbol{\theta}_j \mid \boldsymbol{\theta}_{1:j-1}\right), \tag{9.5}$$

In case of latent sampling (as explored in earlier VAE-based layout models), an additional Kullback–Leibler divergence term may be introduced:

$$\mathscr{L}_{\text{KL}} = \text{KL}(q(z \mid \mathscr{D}) || p(z)), \tag{9.6}$$

The final loss is a weighted combination:

$$\mathscr{L} = \mathscr{L}_{\text{NLL}} + \lambda \mathscr{L}_{\text{KL}}, \tag{9.7}$$

where λ balances the reconstruction and regularization terms.

9.3.5 Inference Strategy

During generation, given an initial sequence of visible tokens $\mathscr{D}_{1:T}$ (layout prompt), the model samples one token at a time to complete the sequence until the end-of-sequence token is predicted or a maximum length is reached. For example, when tasked with generating a scientific article layout with tables and paragraphs, DocSynthv2 can be conditioned on the first few elements and generate plausible layout and content token streams thereafter. Unlike pixel-space generation models [27], this sequence-based generation yields high-resolution editable structures directly useful for downstream applications such as document design, editing, and structure-aware classification.

9.4 SketchGPT: A Generative Transformer for Sketch Completion and Classification

Built upon the foundations of autoregressive GPT-like models, *SketchGPT* is a task-agnostic generative transformer pre-trained on the *QuickDraw* dataset [87], which contains multiple object categories of sketches. The model learns neural representations of sketch data, capturing sequential dependencies among their compounding strokes. The pre-trained representations exhibit adaptability to a wide range of downstream sketch tasks, including sketch completion, generation, and classification. An overview of the framework is shown in Figure 9.2.

9.4.1 Data Preprocessing and Sketch Abstraction

At its core, a sketch is represented as a sequence of time-stamped coordinate points. The QuickDraw dataset stores these sketches using the stroke-3 format, where each point is defined by three values: (x, y, p). Here, x and y represent coordinate offsets, while p denotes the pen state. To ensure data uniformity, min-max normalization is applied such that $x, y \in [0, 1]$.

The *stroke abstraction step* discretizes continuous sketch strokes into a finite dictionary of primitive lines. Inspired by Alaniz *et al.* [2], each stroke is approximated by a straight-line primitive selected from a predefined dictionary. This stroke-to-primitive mapping is computed via cosine similarity between the orientations:

$$\text{sim}(s_i, p_j) = \frac{s_i \cdot p_j}{\|s_i\| \cdot \|p_j\|} \tag{9.8}$$

$$p_i = \arg\max_{p_j \in P} \text{sim}(s_i, p_j) \tag{9.9}$$

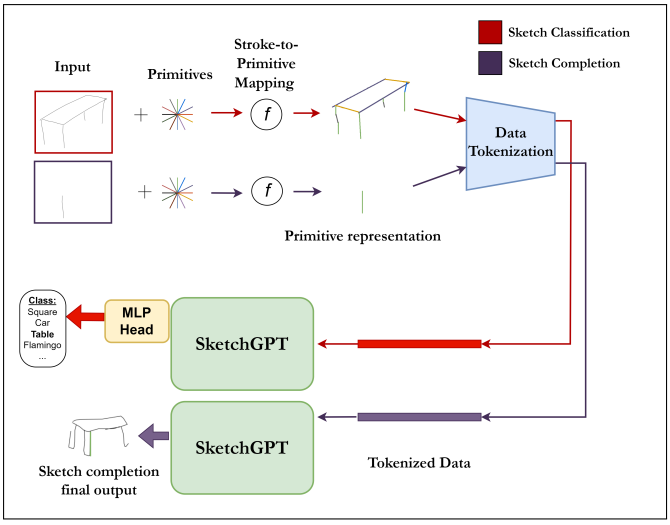


Figure 9.2: Overview of the **SketchGPT** framework for unified sketch understanding and generation. Given an input sketch, each stroke is first mapped to a closest abstract primitive using a *stroke-to-primitive mapping function* to produce a simplified structural representation. This representation is then tokenized and passed through an autoregressive GPT-style model. The model serves a dual purpose: for **sketch classification** (red path), a multi-layer perceptron (MLP) head predicts the object class; for **sketch completion** (purple path), the model continues the sequence to generate plausible remaining strokes. This unified architecture enables multitask learning across sketch domains by modeling stroke sequences as visual language tokens.

To compensate for variable stroke lengths, a scaling factor aligns each primitive with the original stroke:

$$T(p_i, s_i) = \left\lceil \frac{m(s_i)}{m(p_i)} \right\rceil \tag{9.10}$$

The sketch S_i is thus represented as:

$$S_i = \{p_1 \cdot T(p_1, s_1), p_2 \cdot T(p_2, s_2), \ldots, p_n \cdot T(p_n, s_n)\} \tag{9.11}$$

This representation is then tokenized into a sequence using a vocabulary $\mathcal{V}$ including special tokens:

$$\mathcal{T} = [BOS, p_1, \ldots, p_1, SEP, p_2, \ldots, SEP, \ldots, p_n, \ldots, EOS] \tag{9.12}$$

9.4.2 Model Architecture

SketchGPT is based on a decoder-only transformer [204] with causal masked multi-head self-attention. For a sketch sequence X, the masked attention operation is:

$$\text{MaskedAttention}(X) = \text{softmax}\left(\frac{(XW_Q)(XW_K)^T \odot \text{Mask}}{\sqrt{d_k}}\right)(XW_V) \tag{9.13}$$

The multi-head version is expressed as:

$$\text{MultiHead}(X) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)W_O \tag{9.14}$$

where $\text{head}_i = \text{MaskedAttention}_i(X)$. These outputs are passed through an MLP, forming a transformer block. Multiple blocks are stacked to form the model backbone.

9.4.3 Pre-Training SketchGPT

Following GPT [203], SketchGPT is pre-trained with an unsupervised next-token prediction objective over the sketch token corpus:

$$L_{\text{pretrain}} = \sum_{n=i} -\log P(\tau_n | \tau_{n-k}, \ldots, \tau_{n-1}; \Theta) \tag{9.15}$$

This enables the model to learn the structural and semantic patterns inherent in sketch sequences.

9.4.4 Fine-Tuning for Downstream Tasks

After pre-training, SketchGPT is fine-tuned on specific tasks:

Sketch Completion and Generation. The model is trained to complete partial sketches in an autoregressive manner. Conditioned on class context, it predicts missing primitives. Unconditional generation is a special case starting from the `[BOS]` token.

Sketch Classification. Given a tokenized sketch $S_i = [\tau_1, \ldots, \tau_n]$, the final activation is passed through an MLP classifier to predict the class label y_i:

$$L_{\text{classification}} = \sum_{(S,y)} -\log P(y | \tau_1, \ldots, \tau_n) \tag{9.16}$$

These tasks demonstrate the versatility of SketchGPT in adapting a single model architecture for both generative and discriminative tasks within the sketch domain.

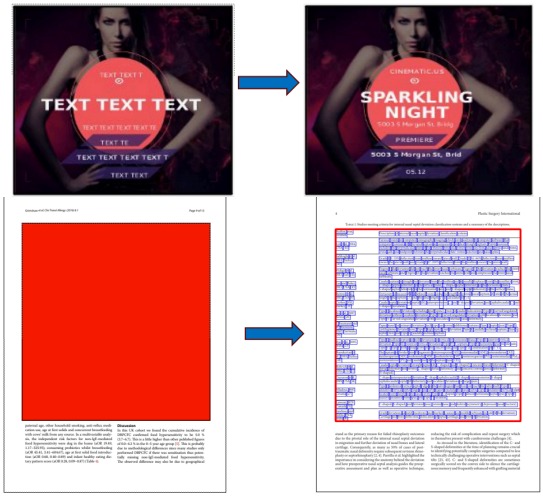


Figure 9.3: **Text Prediction and Document Completion Results using DocSynthv2.** The top row shows an example of text prediction for advertisement design using the Crello dataset, where the model generates realistic and contextually consistent text in a visually guided layout. The bottom row presents document layout and content completion on PubGenNet, where missing content is autoregressively reconstructed, preserving both spatial structure and semantic flow.

9.5 Experimental Evaluation

9.5.1 Tasks in Document Generation

The primary motivations for our model are to address the key aspects of document design and generation. We have selected the evaluation tasks based on: (1) Creating a new document or completing a partially finished one, focusing on maintaining coherence, appearance, and relevance to the intended content. (2) Test the model's ability in layout design, specifically its understanding of spacing, alignment, and the interplay between text and other elements.

Document Completion: This task requires the model to analyze the current layout elements and content within the document (eg. text, title, tables, figures etc.) and logically predict what elements should follow to maintain the coherence and plausible structure of a document.

Single and Multiple Text Box Placement: This task in terms of next element prediction requires the model to identify optimal locations and sizes for text boxes within a document, based on the existing layout and design principles. It assesses the model's capability to seamlessly incorporate new text elements, ensuring they align with the document's structure and visual appeal.

Table 9.1: Quantitative evaluation for Document Completion. Results style: **best**, *second best*. ↑ higher is better and ↓ lower is better.

Model	**mIoU ↑**	**FID ↓**	**Align ↓**	**Over ↓**
LayoutTrans [86]	0.077	14.769	*0.019*	**0.0013**
Layoutformer++ [116]	**0.471**	**10.251**	0.020	0.0022
Ours (w/o txt)	0.315	12.217	0.025	0.0019
Ours (lay+txt)	*0.452*	*10.718*	**0.015**	**0.0013**
Δ	-0.019	+0.467	-0.004	0.000

Table 9.2: Quantitative evaluation for Single and Multiple Box Placement in Crello. Results style: **best**, *second best*. ↑ higher is better and ↓ lower is better.

Model	**Single**		**Multiple**	
	IoU ↑	BDE ↓	IoU ↑	BDE ↓
SmartText [143]	0.047	0.262	0.023	0.300
FlexDM (MM) [109]	**0.357**	**0.098**	**0.110**	0.141
FlexDM (w/o img) [109]	*0.355*	*0.100*	*0.103*	0.156
FlexDM (w/o txt) [109]	0.350	0.106	0.086	0.178
Ours	0.315	0.104	0.105	**0.131**

9.5.2 Quantitative Evaluation for DocSynthv2

Table 9.1 summarizes the performance comparison of DocSynthv2 over the existing SOTA transformer decoder-only models. Our full model (with text attributes) gives us boost in performance over the layout-only model, demonstrating that utilizng the raw text can help guide models for layout generation when avaialble. Although our model is a lightweight decoder-only architecture, it can perform on par with LayoutFormer++ [116] which is an encoder-decoder-based transformer. Our results with high Alignment and Overlap scores also suggest that layout generation and completion models gain substantial improvement when trained on sequences integrating textual content. In Table 9.2, we summarize the performance of Single and Multiple Text Box Placement in the Crello dataset. The results show that the model does worse for text placement in the Single Text box condition, likely due to the weaker multimodal features compared to [109]. However, it performs on par for IoU and outperforms for BDE in the Multiple condition, which may be due to the raw text in our model.

9.5.3 Qualitative Evaluation for DocSynthv2

Figure 9.3 shows example of our applied for text synthesis and document completion on the Crello and PubGenNet datasets. In the Crello Text prediction example, it can

be seen that the text is aligned with the layout showing a plausible flyer title for the heading section followed by an address and date in the sub text fields. For the Document Completion Task, we have the model generate the text within in an existing Table structure. The filled text maintains coherence across the two table columns, filling it with Authors names and reference information on the left and text of the right. In this example the text coherence could likely be improved by LLMs.

9.5.4 Quantitative SketchGPT Evaluation with CNN Classifier

To quantitatively assess the sketch generation quality, we employ a CNN-based evaluation protocol using a pre-trained ResNet34 classifier. The classifier is trained to distinguish between seven categories from the QuickDraw dataset — bus, cat, elephant, flamingo, owl, pig, and sheep. This enables us to measure the recognizability of sketches generated by `SketchGPT` as compared to the baseline `SketchRNN` model.

For each model, we generate 1000 samples per class (7000 total), rasterize them into images, and pass them to the CNN. The classifier achieves a validation top-1 accuracy of 87.92%. As reported in Table 9.3, `SketchGPT` outperforms `SketchRNN` across both top-1 and top-3 accuracy metrics, indicating its superior ability to synthesize human-like, class-consistent sketches.

Table 9.3: CNN-based quantitative evaluation on generated sketches using Top-1 and Top-3 classification accuracy.

Method	**Top-1 Accuracy (%)**	**Top-3 Accuracy (%)**
SketchRNN	44.6	79.1
SketchGPT	**50.4**	**81.7**

9.5.5 SketchGPT Human User Study

We also perform a human evaluation study with 100 participants to assess five qualitative properties of generated sketches: fine detail appreciation, creativity, diversity, human-likeness, and overall preference. Participants were shown five randomly generated sketches per class for both `SketchGPT` and `SketchRNN` across the same seven categories from QuickDraw.

Figure 9.4 summarizes the results, highlighting that `SketchGPT` consistently receives higher scores across all evaluation axes. Notably, the model exhibits stronger diversity and creativity, showcasing its expressive capability in mimicking varied human sketching behavior.

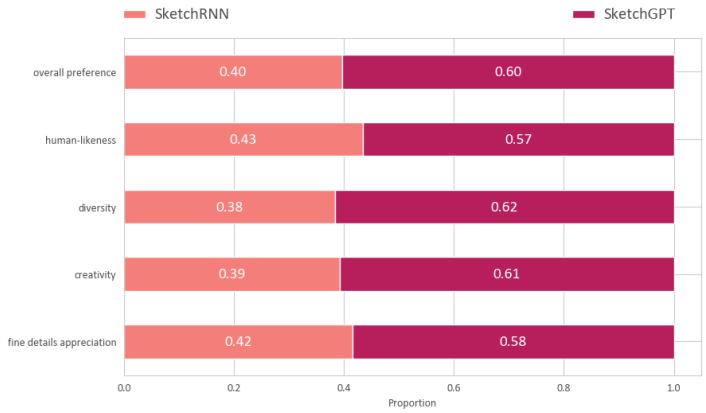


Figure 9.4: Results from the human user study comparing `SketchGPT` and `SketchRNN` across five qualitative dimensions.

9.5.6 Qualitative Evaluation of Sketch Completion

To explore sketch completion capabilities, we input partial sketches into `SketchGPT` and examine the generated completions. Figure 9.5 demonstrates the model's capacity to produce diverse yet coherent sketches for each incomplete input, trained on class-specific datasets. This further reflects the model's robustness in inferring plausible structural continuations.

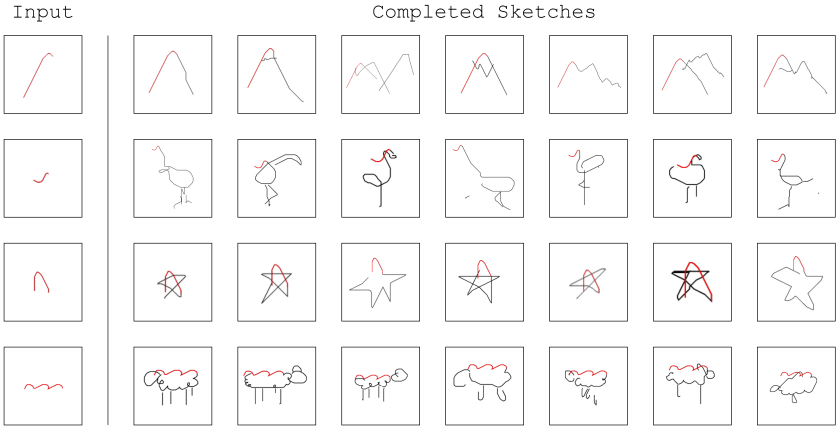


Figure 9.5: Qualitative examples of sketch completion with multiple completions per partial input using class-specific SketchGPT models.

9.5.7 Evaluation for Sketch Recognition

Competitors include traditional models such as HOG-SVM [65] and Ensemble [155], as well as deep models including Sketch-a-Net [282], DSSA [229], ResNet variants [96], SketchBERT [159], and ViT [64]. `SketchGPT` is evaluated both with and without a pre-training phase.

9.5.8 Results and Discussion

As shown in Table 9.4, `SketchGPT` achieves strong performance, outperforming most baselines and only slightly behind `SketchBERT`, which uses substantially more pre-training data. Notably, even without pre-training, `SketchGPT` delivers robust results, validating the effectiveness of its vector-based tokenization and autoregressive training.

Table 9.4: Sketch classification results (Top-1 and Top-5 accuracy) on QuickDraw.

Method	**Top-1 Acc. (%)**	**Top-5 Acc. (%)**
HOG-SVM [65]	52.05	74.50
Ensemble [155]	60.31	80.22
Bi-LSTM [100]	74.68	90.59
Sketch-a-Net [282]	70.64	87.93
DSSA [229]	79.47	92.41
ResNet18 [96]	79.67	91.71
ResNet34 [96]	82.02	93.48
SketchBERT [159]	**88.30**	**97.82**
ViT [64]	47.69	68.73
SketchGPT (w/o Pretrain)	81.42	91.81
SketchGPT (w/ Pretrain)	<u>83.58</u>	<u>93.65</u>

9.6 Ablation Studies

To better understand the behavior and sensitivity of `SketchGPT`, we conduct several ablation studies.

9.6.1 Effect of Temperature on Generation Quality

We study how temperature scaling affects sketch generation by varying the temperature t from 0.6 to 2.0 on the sword class. As shown in Figure 9.6, low temperatures

produce overly simplistic sketches, while very high values lead to erratic outputs. The range between 1.0 and 1.4 offers an optimal trade-off between fidelity and diversity.

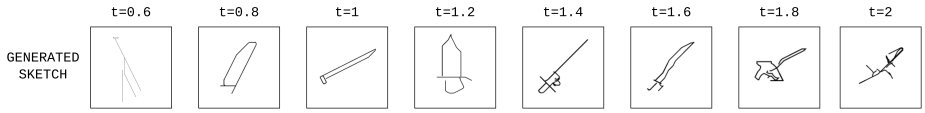


Figure 9.6: Impact of temperature parameter on the quality of generated sketches for the "sword" class.

9.6.2 Impact of Number of Classes on Classification Accuracy

We analyze model performance when trained with increasing class counts (25, 50, 100, and 200 classes), each with 5K samples per class. Results in Table 9.5 indicate a gradual performance drop with more classes, though the degradation is modest up to 100 classes. This suggests that extending training data could mitigate losses at larger scales.

Table 9.5: Classification accuracy of SketchGPT with varying class counts.

# Classes	Top-1 Acc. (%)	Top-5 Acc. (%)
25 Classes	86.3	96.8
50 Classes	85.2	95.4
100 Classes	83.6	93.6
200 Classes	78.9	91.2

9.7 Conclusions and Future Directions

In this final part of the thesis, we explored the challenge of document generation with an explicit focus on layout guidance and content conditioning. We introduced and evaluated two distinct frameworks: `DocSynthv2`, which models layout and textual content autoregressively, and `SketchGPT`, a vector-based generative model designed to handle sketch-like document structures with flexibility and compositional creativity. Through DocSynthv2, we showcased how sequence modeling can enable controllable document synthesis tasks, such as layout completion and text generation, evaluated over benchmarks like PubLayNet and Crello. Our experiments highlight the model's ability to retain structure-content alignment while producing visually diverse and semantically coherent outputs. The framework also allowed us to analyze trade-offs between layout fidelity, semantic preservation, and style diversity. With SketchGPT, we

pushed the boundaries further by treating sketches as structured sequences and applying language modeling techniques to vectorized strokes. Our empirical analysis shows its advantage not only in sketch generation and completion, but also in classification and stylistic variation.

Together, these contributions illustrate the power and versatility of autoregressive models when applied to layout-aware generation tasks. They emphasize that structured tokenization, layout semantics, and vector representation form the foundation for controllable and high-quality document generation. This part of the thesis lays important groundwork for future research at the intersection of document understanding, multimodal generation, and creative AI.

While the presented methods in this part demonstrate strong potential in layout-aware document generation, several promising directions remain open for future exploration:

- **Joint Multimodal Pretraining.** Current approaches treat layout and content sequentially but independently. An exciting avenue lies in jointly pretraining models across image, layout, and text modalities using large-scale multimodal corpora. Such a model could enable more nuanced and semantically grounded generation by aligning visual, spatial, and linguistic representations.
- **Sketch-Conditioned Document Generation.** Leveraging sketch input as a guidance signal for document synthesis such as drafting layouts or stylistic strokes can enable more intuitive human-in-the-loop generation systems. Conditioning DocSynthv2 type models with vectorized sketches or human-drawn wireframes would open new creative and design applications.
- **Graph-Augmented Autoregressive Models.** Given the inherently structured nature of documents, integrating graph-based reasoning into autoregressive models could further improve coherence, especially for hierarchical layouts (e.g., nested tables, multi-column formats). Recent advances in graph neural networks and relational transformers may serve as key building blocks for this integration.
- **Task-Specific Adaptation.** Extending the generation frameworks to task-specific settings such as scientific poster design, invoice templating, or multi-language form generation could help evaluate generalization under practical constraints. Fine-tuning on low-resource document types would assess robustness and transferability.
- **Human-Centric Evaluation Metrics.** Beyond traditional metrics like FID or IoU, designing human-centered evaluation frameworks including usability studies, cognitive load assessments, or layout aesthetics scoring could offer richer insights into the quality and impact of generated documents and sketches.

By pursuing these directions, we anticipate significant progress toward making document generation systems not only more controllable and expressive, but also more aligned with human needs, creativity, and real-world utility.

Chapter 10

Conclusion and Future Work

"There is no real ending. It's just the place where you stop the story."
— Frank Herbert, Dune (1965)

In this chapter, we summarize the contributions of this thesis to the field of Document AI by exploring layout as a foundational visual language. We revisit how the thesis navigated three core research axes—understanding, representation, and generation/editing—and conclude by reflecting on open questions and promising research directions.

10.1 Bridging the Axes: Layout as Visual Grammar

This thesis proposed to treat document layout not just as structural metadata, but as a semantic language—one that carries meaning, guides visual attention, and enables reasoning across downstream tasks. Across its trajectory, the thesis explored this idea through three interconnected research axes:

- **Understanding:** Can we teach models to perceive layout like humans do i.e. detecting and segmenting visual elements as coherent units?
- **Representation:** How can we encode layout to make it useful for other downstream tasks, particularly when explicit supervision is unavailable?
- **Generation & Editing:** Can we reverse the process—synthesize or modify layouts while preserving their semantic and structural intent?

By traversing these axes, the thesis demonstrates that layout-aware models can move beyond mere perception—towards understanding, reasoning, and creative generation of documents. This holistic view positions layout not only as an interpretive tool, but also as a generative prior for richer, more context-aware document intelligence.

While the research presented here makes optimal progress across these axes, the journey towards truly *layout-literate* AI systems remains ongoing. Many of the methods and findings in this thesis open the door to new opportunities, and reveal challenges that merit further investigation. The following sections will provide an insight of the thesis contributions summary, outline further open challenges and potential future directions, with the aim of guiding subsequent research at the intersection of layout understanding, representation, and generation.

10.2 Summary of Thesis Contributions

This thesis presents a unified exploration of the hypothesis that *document layouts function as a latent language*, capturing both spatial syntax and semantic structure. By organizing our work across three thematic axes — **Interpretation**, **Representation**, and **Generation** — we contribute a diverse set of models and frameworks that collectively advance the field of Document AI. The core contributions are summarized below:

10.2.1 Interpretation: Layout-Aware Document Segmentation

We address the challenge of layout segmentation beyond bounding-box detection, proposing pixel-level, transformer-based, and semi-supervised methods that model the structural grammar of documents:

- **Mask R-CNN-based segmentation** tailored for complex structured layouts in scientific PDFs and historical archives.
- **DocSegTr**, a twin-attention transformer with interpretability features and a novel inverse focal loss, targeting better recall for small but critical layout elements.
- **SwinDocSegmenter**, a unified, modular Swin Transformer-based framework demonstrating high performance and robustness across domains.
- **SemiDocSeg**, a semi-supervised extension that introduces co-occurrence priors and support sets to boost generalization in low-label settings.

These models demonstrate how interpreting layout as structure improves instance-level understanding and segmentation accuracy across diverse document types.

10.2.2 Representation: Learning Layout Semantics through Self-Supervised and Graph-Based Models

This axis focuses on learning document representations that capture structural and relational information using self-supervised and graph-based approaches:

- **SelfDocSeg**, a vision-only self-supervised framework that leverages layout-aware augmentations and BYOL-style contrastive learning to pre-train models without labeled data.
- **Doc2GraphFormer**, a graph-augmented lightweight document transformer model that integrates layout, visual and textual cues into a task-agnostic document understanding pipeline.

These contributions establish that document representations can be effectively learnt by leveraging layout structure itself — even in the absence of explicit annotation — enabling scalable and generalizable understanding.

10.2.3 Generation: Layout-Controlled Document Synthesis and Grammar Modeling

In the final axis, we investigate how layout can guide generative modeling, treating it as a prior for controllable and structured document generation:

- **DocSynth**, a layout-guided synthesis pipeline that disentangles layout and content for controllable document image generation.
- **DocSynthv2**, an autoregressive modeling approach that encodes layout and text as sequences, enabling text completion and structured layout-text generation on real datasets.
- **SketchGPT**, a sequence-based sketch generation framework modeling layout primitives as tokens, supporting sketch classification and completion through transformer-based decoding.

Together, these works push the boundary of document generation by viewing layout as a *generative grammar* — where both structure and content can be modeled, edited, and synthesized in a coherent and controllable manner.

10.3 Limitations and Future Work

While the thesis presents a coherent progression from interpreting document layouts to generating them, each part introduces its own set of assumptions, design choices,

and open questions. In this section, we critically reflect on the methodological decisions and practical limitations across the three research axes.

10.3.1 Interpretation: Scope of Supervision and Generalization

The first part of the thesis focused on transitioning from bounding-box detection to fine-grained, instance-level segmentation. Although the proposed models (DocSegTr, SwinDocSegmenter, and SemiDocSeg) demonstrate strong performance on standard benchmarks, several challenges remain:

- **Domain shift and visual bias:** Although domain-adaptive components improve robustness, the models remain susceptible to degradation under distribution shifts. Background artifacts, unconventional spatial arrangements, and atypical element shapes—common in handwritten, historical, or visually degraded documents—still cause performance drops. This points to a need for architectures that can disentangle true semantic structure from incidental visual noise.
- **Annotation cost and scalability:** High-quality pixel-level masks remain the gold standard for supervised training, yet they are prohibitively costly at scale. The semi-supervised extension (*SemiDocSeg*) mitigates this to some degree, but its performance is sensitive to the quality and representativeness of the chosen support sets, indicating that scalability still depends on informed curation.
- **Interpretability and transparency:** While transformer-based architectures provide flexibility and accuracy, they largely operate as opaque systems. Attention maps offer a partial window into model reasoning, but they do not guarantee faithful interpretability. Without deeper semantic explainability, it is challenging to build trust in critical domains such as legal or medical document processing.

Addressing these limitations will require coordinated efforts from the AI research community, spanning both methodological and infrastructural developments. In particular, we identify the following priorities:

- **Beyond benchmark-centric evaluation**: Current benchmarks reward accuracy on well-curated, clean datasets. To foster true generalization, the community should adopt evaluation suites that explicitly *target domain shift—e.g., cross-domain validation* across printed, handwritten, historical, and design-heavy layouts, with controlled degradations. RoDLA [42] is a viable approach towards this direction.
- **Reducing annotation dependency at scale**: While semi-supervised and weakly supervised methods show promise, progress will depend on leveraging large-scale synthetic or procedurally generated datasets, along with self-supervised pretraining objectives that are explicitly layout-aware rather than text- or image-only. *Instruction-tuned datasets* for training multimodal models in this regard can be more powerful and generalizable as shown in BigDocs [215].

- **Towards interpretable layout reasoning**: Rather than treating attention maps as a post-hoc diagnostic, future work should integrate *intrinsically interpretable mechanisms into model design*—such as modular reasoning pipelines, structured scene graphs, or symbolic-layout hybrids—to make decision pathways explicit and auditable. DocXVQA [232] is a potential approach that could be exploited towards having such level of grounded information for spatial, visual and textual modalities.
- **From perception to functional understanding**: The community should move beyond "where" elements are located to "why" they are arranged that way. This requires incorporating multimodal cues (visual, textual, and structural) and explicit reading order information (eg. LayoutReader [263]) into models so that layout understanding aligns with document intent and usage context.
- **Agentic AI for structure–content extraction beyond OCR:** A promising frontier is the development of agent-based document understanding systems that can iteratively plan, reason, and adapt their extraction strategy depending on document complexity. Such systems would move beyond static OCR pipelines to actively parse layout, infer hierarchical relationships, and extract structured content while validating and correcting errors in context—mirroring the way human analysts interact with documents.
- **Sustainable model deployment:** Domain-adaptive training pipelines should be designed with efficiency in mind to reduce compute costs, enabling on-device inference, and making models more accessible to low-resource organizations that handle specialized document types.

Taken together, these directions emphasize that progress in document layout interpretation is not just a matter of incremental accuracy gains, but of building systems that are robust, transparent, and truly useful across the diverse realities of document collections in the wild.

10.3.2 Representation: Structural Biases and Transferability

Part II of this thesis introduced self-supervised and graph-based models for capturing layout semantics without relying on manual annotations. Approaches such as *SelfDocSeg* and *Doc2GraphFormer* demonstrated that rich structural priors can be learned and transferred effectively to a variety of downstream tasks. However, several critical limitations and open questions remain:

- **Layout-centric bias:** Current representation methods [6, 7, 284, 82] prioritize structural regularities, sometimes at the expense of semantic or textual cues. In scenarios where meaning is embedded in subtle language patterns, visual style, or multimodal interplay, layout-dominant embeddings may fail to capture essential context. Bridging this gap calls for more balanced architectures that

can integrate visual, structural, and semantic signals without overfitting to one modality. A good starting point in this direction could be the usage of GlobalDoc [11] framework that uses only visual and textual cues and learns a shared embedding space for capturing "implicit" document structure. Extending this to integrate a more balanced spatial modality can be the next potential step.

- **Graph formulation sensitivity:** The performance of graph-based models is tightly coupled to the choice of node features and edge definitions. In *Doc2Graph*-style systems [78, 79, 22, 183], these are often manually engineered and task-specific, limiting adaptability and generalization. End-to-end learnable graph construction or agentic systems capable of dynamically revising the graph topology based on task feedback could help to alleviate this issue.
- **Downstream task alignment:** Although the learned representations transfer well to segmentation and classification, their utility for more complex objectives (e.g., entity linking, hierarchical reasoning, logical ordering) is underexplored. Representation learning should move towards task-aware adaptation, ensuring that embeddings capture the specific relational and semantic properties demanded by the target application.

From a broader perspective, the community's next steps in this space should include:

- Designing **multi-view pretraining objectives** that jointly optimize for structural fidelity and semantic depth, enabling representations that work across visually and semantically diverse document types.
- Exploring **dynamic graph induction** via neural or agentic approaches, where graph structure evolves adaptively as the model processes the document, rather than being fixed at preprocessing time.
- Investigating **cross-task, cross-domain transfer** benchmarks to measure how well learned representations generalize to entirely new document genres and task families.
- Integrating **agentic AI planning loops** into representation learning—where an AI agent could, for example as in a framework like LATIN-Prompt [257], iteratively query different parts of the document, update its graph representation, and refine embeddings based on reasoning goals rather than static input.

Addressing these challenges will be essential to move from general-purpose document embeddings toward *goal-aware, task-adaptive representations* that can serve as a foundation for the next generation of layout-literate AI systems.

10.3.3 Generation: Grammar Modeling vs. Visual Fidelity

The final part of this thesis investigated layout-guided and autoregressive approaches for document generation, with *DocSynthv2* [25] and *SketchGPT* [243] demonstrating

promising capabilities for controllable synthesis. These models illustrate how layout can serve as both a conditioning signal and a generative prior. However, several unresolved tensions and methodological constraints remain:

- **Sequence modeling constraints:** Treating documents as linear sequences (whether tokens, strokes, or layout primitives) imposes an ordering that may be misaligned with inherently non-linear structures such as multi-column formats, tables, or graph-like layouts. This can limit a model's ability to capture parallel reading flows or interlinked visual relationships.
- **Visual vs. semantic trade-offs:** Generating visually compelling layouts with high fidelity (as in *DocSynth*) can conflict with preserving semantic integrity, particularly when text content is synthesized alongside graphics. Conversely, focusing on semantic accuracy may lead to visually unnatural layouts. Achieving a balanced optimization of both still remains a core open challenge.
- **Data and tokenization limitations:** Both *SketchGPT* and *DocSynthv2* depend on specific datasets and handcrafted tokenization schemes. These may fail to generalize to multilingual settings, handwritten documents, or hybrid text–graphic compositions, where tokenization rules become more ambiguous and domain-specific.

To advance the field of layout-aware document generation, we identify several future priorities:

- **Hybrid structural representations:** Develop models capable of seamlessly combining sequence-based and graph-based representations [139], enabling the capture of both linear narrative flow and complex spatial relationships.
- **Dual-objective training regimes:** Introduce multi-task or multi-objective optimization frameworks that jointly maximize visual realism and semantic faithfulness, potentially leveraging disentangled latent spaces for structure and content.
- **Multilingual and multimodal generalization:** Curate diverse training corpora spanning multiple languages, scripts, and media types, along with adaptive tokenization strategies that can evolve during training.
- **Agentic generation workflows:** Extend document generation into iterative, agent-driven pipelines where the system can plan, assess, and revise its own output—e.g., generating a draft layout, running semantic validation, and refining specific regions until both structure and content meet task requirements.
- **Reward models for layout-aware RL:** Train reward function models that explicitly score generated documents on multiple axes (visual quality, structural alignment, semantic accuracy, and controllability) and use these signals to guide reinforcement learning policies. Such reward-driven optimization could enable agentic systems to learn iterative improvement strategies, rather than relying solely on static supervised loss functions.

- **Evaluation beyond FID and Alignment metrics:** Move toward holistic evaluation metrics that jointly assess visual quality, structural alignment, semantic accuracy, and user-controllable attributes, allowing fairer comparison of different generation paradigms.

By addressing these gaps, the community can move toward generative systems that treat layout not merely as a canvas for rendering, but as an active, manipulable grammar—capable of producing documents that are both visually compelling and semantically coherent across a wide range of real-world scenarios.

10.4 Success Stories and Real-World Impact

Beyond academic exploration and benchmark performance, several components developed in this thesis have been integrated into real-world systems, demonstrating their value in enterprise-grade document intelligence pipelines. These deployments validate not only the technical soundness of the methods, but also their operational reliability under the messy, unpredictable conditions of production environments.

Industrial Deployment of SWINDOCSEGMENTER: A key example comes from an industrial collaboration where the *SwinDocSegmenter* framework [13] was deployed as the primary layout interpretation module within a large-scale, AI-powered document processing system used for real-time information extraction and summarization. In industrial settings, incoming documents often deviate significantly from curated research datasets: contracts scanned at non-uniform angles, administrative forms containing stamps and handwritten notes, archival material with degraded print quality, and a wide variety of legacy templates. Accurate layout parsing in such contexts is essential, as it forms the structural foundation for any downstream analysis. The deployment leveraged *SwinDocSegmenter* for:

- **Robust multi-domain segmentation:** Handling both machine-printed and scanned forms without retraining, the model demonstrated resilience to background noise, unusual spatial arrangements, and visual artifacts.
- **Seamless modular integration:** Thanks to its domain-adaptive design, the model was integrated into the existing pipeline without requiring major architectural changes, replacing brittle rule-based systems and enabling rapid onboarding of new document types.
- **Structural blueprint for downstream AI:** The segmentation maps became a structural scaffold for subsequent tasks, such as:
  - Automated table extraction and normalization.
  - Key–value pairing for contractual and administrative data.
  - Entity detection and change tracking across document versions.

- **Operational efficiency at scale:** In production, the system processed hundreds of documents per minute under strict latency constraints, enabling near-real-time querying and summarization of large repositories.

This industrial deployment underscores the thesis's emphasis on *domain-adaptive architecture* and *robust representation learning*—key factors that allowed a research model to transition smoothly into an enterprise environment, reduce manual review workload, and accelerate decision-making on a large scale.

From Research to Creative Prototypes: Layout-Conditioned Generation in Action: The generative frameworks developed in Part III of this thesis—*DocSynth*, *DocSynthv2* and *SketchGPT*—have inspired exploratory prototypes that illustrate how layout-conditioned generation can move beyond the lab and into creative, interactive, and resource-critical workflows. These demonstrations served as proof-of-concept systems, bridging the gap between theoretical modeling and real-world application.

- **DocSynth for synthetic data generation:** Leveraging its controllable layout-to-document synthesis capabilities, *DocSynth* was applied to a prototype document augmentation tool for creating large-scale synthetic datasets for a French Bank Corporation. This was particularly valuable in such low-resource domains, such as specialized forms and niche administrative templates in French, where collecting annotated real data is prohibitively costly. The generated documents maintained structural and stylistic diversity while preserving semantic plausibility—making them suitable for downstream training of OCR engines, key–value extraction models, and form-understanding systems.
- **DocSynthv2 for customizable template generation (Adobe Internship Project)**: During a research internship with a big giant design industry, *DocSynthv2* was extended into an advanced layout-aware generation tool capable of creating and modifying templates with user-specified text content. This system allowed fine-grained control over both structural arrangement and semantic composition, enabling rapid prototyping of document templates for marketing, publishing, and creative design workflows. Its sequence-based modeling of layout and text facilitated seamless adaptation to diverse template styles while preserving design coherence.
- **SketchGPT for creative sketch-based prototyping (Research Demo Platform):** *SketchGPT* was showcased as an interactive platform for sketch-based document creation, where users could draw rough layout strokes and receive AI-driven completions. This tool proved valuable for rapid ideation in UI/UX layout design, educational demonstrations, and creative prototyping, lowering the entry barrier for structured design generation. By combining free-form human input with structured generative modeling, it demonstrated a human–in-the-loop approach to layout creation.

Together, these prototypes demonstrate that modeling layout as language is not merely a conceptual framework but a practical enabler across domains—from synthetic data

generation for enterprise OCR systems to customizable template creation for industry partners, and creative sketch-based design for research and education. They highlight the adaptability of the proposed generative models to both production-driven and exploratory creative settings, reinforcing the thesis's vision of layout-aware AI as a bridge between document understanding and document creation.

10.5 Grand Challenge: Multimodal Reasoning in Layout Understanding

As Document AI advances, a key unresolved frontier is the ability to *reason jointly across modalities*—text, diagrams, equations, and layout—particularly in unstructured and handwritten material. This thesis has laid the foundation for modeling layout as a language across interpretation, representation, and generation. The natural next step is to ask: *Can machines reason with layout in the wild, as humans do when navigating scientific notes or answering open-ended questions?*

Why this is hard: Unlike clean, typeset documents where text lines, headings, and figures follow predictable formatting rules, handwritten scientific notes are inherently irregular and often idiosyncratic to the author's style. They exhibit:

- **Non-linear writing flows:** Scientific note-taking rarely follows a strict left-to-right, top-to-bottom order. Equations may be inserted mid-sentence, diagrams may interrupt paragraphs, and annotations can refer to distant parts of the page. This breaks the assumptions of sequence-based reading and requires models to dynamically re-order and link related elements.

- **Domain-specific symbols and notations:** From chemical structure diagrams to Feynman diagrams in physics, shorthand in biology, or custom mathematical symbols, these visual tokens often carry meaning that is not explicitly explained within the note itself. Recognizing and interpreting them demands both visual pattern recognition and domain-specific semantic grounding.

- **Informal sketches and multi-domain elements:** Hand-drawn flowcharts, quick conceptual diagrams, and schematic representations frequently coexist with text. These are often incomplete or abstract, relying on the reader's prior knowledge to fill in missing details—making them challenging for models trained on clean, fully specified figures.

- **Sparse or implicit layout cues:** Many handwritten notes lack consistent spacing, alignment, or clear bounding regions for different content types. Instead, semantic relationships are implied through proximity, arrows, underlines, or color cues. Such implicit structure forces models to infer relationships that are not explicitly marked.

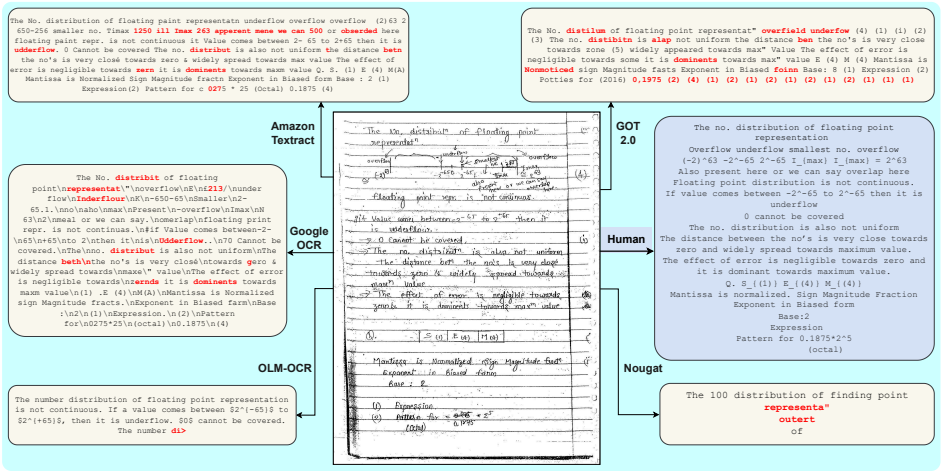


Figure 10.1: **Comparison of OCR and Vision-Language Model outputs on a handwritten scientific note.** The central image shows a real student-authored note with equations and prose. Surrounding boxes show outputs from commercial OCR tools (Amazon Textract, Google OCR), vision-language-based OCRs (OLM-OCR, Nougat, GOT 2.0), and a human annotation. The image highlights critical failures in structure, symbol transcription (e.g., math notations), and semantic understanding across models, underlining the need for multimodal reasoning beyond plain OCR. Figure adapted from the NoTeS-BANK benchmark

These characteristics collectively undermine the effectiveness of current OCR- and layout-only approaches, which typically assume a single reading order and clearly defined content boundaries. Overcoming them will require systems that can *jointly* interpret multiple modalities—visual, textual, symbolic, and spatial—while grounding their reasoning in both the document's visual evidence and the domain-specific context of its content. As illustrated in Figure 10.1 on a scientific note sample, we see how it's interpretation challenges the current paradigm of layout parsing, and demand reasoning that is not only multimodal but also grounded in visual evidence and domain-specific context.

Curation of a New Reasoning Benchmark: To push beyond the limitations of OCR- and layout-only methods, we are developing **NOTES-BANK**, a benchmark for *evidence-grounded, multimodal question answering* over scientific handwritten notes. These notes present highly irregular structures—non-linear writing flows, domain-specific symbols, informal sketches, and sparse layout cues—that demand models capable of joint symbolic, spatial, and semantic reasoning. NOTES-BANK introduces two tasks: (1) *Evidence-Based VQA*, requiring answers to be returned with bounding-box visual evidence, and (2) *Open-Domain VQA*, which combines domain classification, retrieval of relevant notes, and cross-modal grounded answer generation. This represents a shift toward systems that can "*read like a student*"—localizing, connecting, and syn-

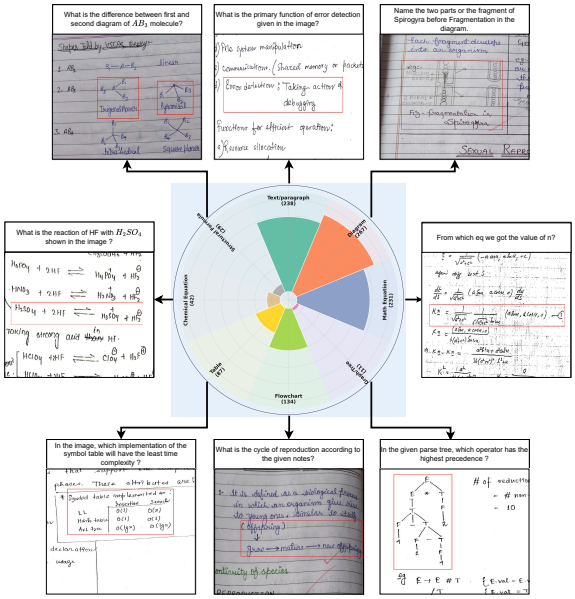


Figure 10.2: **Illustration of diverse answer types and visual object categories within the NOTES-BANK benchmark.** The central radial chart shows the distribution of annotated regions across semantic categories such as text, equations, diagrams, chemical structures, and flowcharts. Surrounding examples depict questions that require grounding answers to specific visual cues, such as boxed formulae, underlined labels, structural fragments, or parse trees. These instances emphasize the role of **layout grammar**—the implicit spatial organization of symbols, annotations, and multimodal components—in enabling human-like reasoning. Understanding such unstructured scientific notes necessitates vision-language models that can infer semantic roles from layout context, symbol types, and spatial alignment.

thesizing dispersed, symbol-rich content. The benchmark builds on this thesis's vision by treating layout as a semantic cue for evidence attribution, extending layout-aware generation into layout-aware reasoning, and opening a new axis of *retrieval-based, visually grounded reasoning* that could leverage architectures such as *Doc2Graph* and autoregressive layout modeling. Achieving this will require vision-language pretraining on noisy, handwritten, diagram-rich data, cross-modal fusion strategies that preserve spatial anchoring, and layout-aware prompting and retrieval methods. As shown in Figure 10.2, real-world handwritten scientific notes involve diverse object types such as formulas, parse trees, diagrams, and equations, all situated within informal layouts. Modeling these layouts as latent grammar is crucial for enabling grounded question answering over unstructured visual content. Ultimately, such advancements would enable machines to not only "read" but to "reason with layout"—unlocking intelligent AI systems that can tutor, summarize, or collaborate with humans over visually complex documents.

10.6 Epilogue

When this journey began, the challenge seemed deceptively simple: documents are everywhere, yet machines still fail to read them with the nuance and adaptability of humans. A contract, a handwritten note, a research paper—each communicates not only through words, but through its spatial composition, its visual rhythm, its layout. Over the course of this work, that intuition crystallized into three interconnected research axes.

The first, *Understanding*, asked whether machines could learn to *see* documents as we do—identifying coherent units, segmenting them with precision, and remaining resilient to the countless variations found in the wild. With *SwinDocSegmenter* and its extensions, we saw this capability deployed in real industrial pipelines, replacing brittle heuristics with learned, domain-adaptive structure.

The second, *Representation*, sought to move from seeing to *knowing*—to encode layout in a form that could travel across tasks and domains, enabling new applications without starting from scratch. Models like *SelfDocSeg* and *Doc2GraphFormer* proved that structure can be learned without labels, but also revealed the subtle biases and design choices that shape what gets preserved—and what gets lost—in these representations.

The third, *Generation*, reversed the perspective entirely: could machines *create* layouts with intent, preserving both their visual form and their semantic meaning? Through *DocSynth*, *DocSynthv2*, and *SketchGPT*, this work showed that layout-aware generation could serve both pragmatic ends—like synthetic data for OCR—and creative ones, from design prototyping to sketch-based ideation.

These ideas did not remain confined to academic benchmarks. In industry collaborations, layout parsing became a production-ready enabler of large-scale document intelligence. Generative prototypes found use in low-resource training pipelines and creative workflows. These deployments affirmed that "layout as language" is not just a theoretical lens—it is a practical foundation for systems operating in messy, high-stakes environments.

And yet, the story is far from over. The next grand challenge—multimodal reasoning—demands systems that can think across text, diagrams, equations, and layout, navigating handwritten notes and symbol-rich pages as a human student or researcher might. The NOTES-BANK benchmark is a first step, inviting models to read with evidence, reason across modalities, and engage in iterative, agentic understanding.

In the end, this thesis is less a conclusion than a bridge: from perception, to knowledge, to creation—and onward to reasoning. It is a reminder that the limits of how we teach machines to read and reason are, in a sense, the limits of the worlds they can inhabit. As Wittgenstein once observed, *"The limits of my language mean the limits of my world."*

List of Key Contributions

Simple things should be simple, complex things should be possible.
– Alan Key

Topics

The central theme of this dissertation is the development of more effective **layout-aware document understanding systems**. The core contributions revolve around modeling document layout as a latent language to support both the *interpretation* and *generation* of document structure. However, the doctoral journey also gave rise to a number of complementary research directions within the broader field of Document AI. These were intentionally excluded from the main narrative to maintain thematic coherence, yet they represent valuable by-products of the research process and contribute to the field in their own right. Key among these are:

- **Document Image Enhancement:** Development of deep learning frameworks for restoring degraded document images through denoising, deblurring, and binarization. These approaches as in *DocEnTr* [230] and *Text-DIAE* [231], often guided by perceptual and task-specific loss functions, were designed to improve both human readability and OCR performance under challenging acquisition conditions.
- **Scene Text Spotting:** Exploration of text spotting in natural scenes and noisy, real-world environments, including multilingual and domain-specific contexts. A couple of works related to domain adaptive text spotting [49, 50] addressed the combined challenges of detection and recognition in visually cluttered, low-quality, or stylistically varied imagery.
- **User-Guided Document and Scene Text Editing:** Early prototypes of interactive, layout-aware, and font-agnostic text editing systems for both documents and scene images. These systems [51] sought to integrate semantic understanding with visual realism, enabling fine-grained, human-in-the-loop modifications

while preserving overall design and layout integrity. In addition, this body of work also included the early development of **DocEdit** [182], a redefined document editing framework [265] aimed at structured, layout-aware modifications with precise semantic and visual consistency.

While peripheral to the main focus, these contributions reflect the breadth of challenges encountered when pushing toward more robust, flexible, and user-adaptive document understanding technologies. They also illustrate the potential for cross-pollination between core layout reasoning research and adjacent areas such as document restoration, scene text analysis, and interactive visual editing.

International Journals

- Pau Torras, **Sanket Biswas***, and Alicia Fornés. "A unified representation framework for the evaluation of Optical Music Recognition systems." *International Journal on Document Analysis and Recognition (IJDAR)* 27, no. 3 (2024): 379-393.

- Ayan Banerjee, **Sanket Biswas***, Josep Lladós, and Umapada Pal. "SemiDocSeg: harnessing semi-supervised learning for document layout analysis." *International Journal on Document Analysis and Recognition (IJDAR)* 27, no. 3 (2024): 317-334.

- **Sanket Biswas***, Pau Riba, Josep Lladós, and Umapada Pal. "Beyond document object detection: instance-level segmentation of complex layouts." *International Journal on Document Analysis and Recognition (IJDAR)* 24, no. 3 (2021): 269-281.

Selected International Conferences

- Alloy Das, **Sanket Biswas***, Prasun Roy, Subhankar Ghosh, Umapada Pal, Michael Blumenstein, Josep Lladós, and Saumik Bhattacharya. "FASTER: A Font-Agnostic Scene Text Editing and Rendering Framework." In *Winter Conference on Applications of Computer Vision (WACV)*, pp. 1944-1954. IEEE, 2025. **(*Spotlight, Equal Contribution)**

- Souhail Bakkali, **Sanket Biswas***, Zuheng Ming, Mickaël Coustaty, Marçal Rusiñol, Oriol Ramos Terrades, and Josep Lladós. "GlobalDoc: A Cross-Modal Vision-Language Framework for Real-World Document Image Retrieval and Classification." In Winter Conference on Applications of Computer Vision (WACV), pp. 1436-1446. IEEE, 2025. **(*Spotlight)**

- **Sanket Biswas***, Rajiv Jain, Vlad I. Morariu, Jiuxiang Gu, Puneet Mathur, Curtis Wigington, Tong Sun, and Josep Lladós. "Docsynthv2: A practical autoregressive

modeling for document generation." In Proceedings of the IEEE/CVF Conference on *Computer Vision and Pattern Recognition*, pp. 8148-8153. 2024. **(*Spotlight)**

- Ayan Banerjee, **Sanket Biswas***, Josep Lladós, and Umapada Pal. "Graphkd: Exploring knowledge distillation towards document object detection with structured graph creation." In *International Conference on Document Analysis and Recognition*, pp. 354-373. Cham: Springer Nature Switzerland, 2024. **(*Best Student Paper ICDAR 2024)**

- Alloy Das, **Sanket Biswas***, Umapada Pal, and Josep Lladós. "Diving into the depths of spotting text in multi-domain noisy scenes." In 2024 IEEE *International Conference on Robotics and Automation (ICRA)*, pp. 410-417. IEEE, 2024. **(*Oral, Equal Contribution)**

- Alloy Das, **Sanket Biswas***, Ayan Banerjee, Josep Lladós, Umapada Pal, and Saumik Bhattacharya. "Harnessing the power of multi-lingual datasets for pre-training: Towards enhancing text spotting performance." In *Winter Conference on Applications of Computer Vision,* pp. 718-728. 2024.

- Ayan Banerjee, **Sanket Biswas***, Josep Lladós, and Umapada Pal. "Swindocsegmenter: An end-to-end unified domain adaptive transformer for document instance segmentation." In *International Conference on Document Analysis and Recognition*, pp. 307-325. Cham: Springer Nature Switzerland, 2023. **(*Oral, Equal Contribution)**

- Adarsh Tiwari, **Sanket Biswas***, and Josep Lladós. "Sketchgpt: Autoregressive modeling for sketch generation and recognition." In *International Conference on Document Analysis and Recognition*, pp. 421-438. Cham: Springer Nature Switzerland, 2024. **(*Oral)**

- Subhajit Maity, **Sanket Biswas***, Siladittya Manna, Ayan Banerjee, Josep Lladós, Saumik Bhattacharya, and Umapada Pal. "Selfdocseg: A self-supervised vision-based approach towards document segmentation." In *International Conference on Document Analysis and Recognition*, pp. 342-360. Cham: Springer Nature Switzerland, 2023. **(*Oral, Equal Contribution)**

- Andrea Gemelli, **Sanket Biswas***, Enrico Civitelli, Josep Lladós, and Simone Marinai. "Doc2graph: a task agnostic document understanding framework based on graph neural networks." In *European Conference on Computer Vision*, pp. 329-344. Cham: Springer Nature Switzerland, 2022.

- Mohamed Ali Souibgui, **Sanket Biswas***, Andres Mafla, Ali Furkan Biten, Alicia Fornés, Yousri Kessentini, Josep Lladós, Lluis Gomez, and Dimosthenis Karatzas. "Text-DIAE: a self-supervised degradation invariant autoencoder for text recognition and document enhancement." In the *AAAI conference on artificial intelligence*, vol. 37, no. 2, pp. 2330-2338. 2023. **(*Equal Contribution)**

- **Sanket Biswas***, Pau Riba, Josep Lladós, and Umapada Pal. "Docsynth: a layout guided approach for controllable document image synthesis." In *International Conference on Document Analysis and Recognition*, pp. 555-568. Cham: Springer International Publishing, 2021.

- Nil Biescas, Carlos Boned, Josep Lladós, and **Sanket Biswas***. "Geocontrastnet: Contrastive key-value edge learning for language-agnostic document understanding." In *International Conference on Document Analysis and Recognition*, pp. 294-310. Cham: Springer Nature Switzerland, 2024. **(*Oral)**

- Jordy Van Landeghem, **Sanket Biswas***, Matthew Blaschko, and Marie-Francine Moens. "Beyond document page classification: design, datasets, and challenges." In IEEE/CVF *Winter Conference on Applications of Computer Vision*, pp. 2962-2972. 2024. **(*Oral)**

arXiv

- **Sanket Biswas***, Ayan Banerjee, Josep Lladós, and Umapada Pal. "DocSegTr: an instance-level end-to-end document image segmentation transformer." arXiv preprint arXiv:2201.11438 (2022).

International Dataset Collaborations

In addition to methodological and system-level contributions, this doctoral work was also shaped by active participation in collaborative dataset creation efforts within the Document AI research community. These initiatives addressed gaps in existing resources by providing large-scale, diverse, and task-specific datasets that enable robust benchmarking and foster reproducible research.

- **DUDE [247]**: A comprehensive benchmark for *document understanding in diverse environments*, combining multiple domains, layouts, and modalities to support evaluation across a wide range of real-world scenarios. *Role:* Contributed to dataset design, curated subsets for layout-rich domains, and coordinated annotation guidelines and paper writing.

- **BigDocs [215]**: A large-scale corpus of document images designed for multimodal pretraining and evaluation for newly introduced document reasoning tasks, with a focus on scaling layout-aware models to handle millions of pages across industries and formats. *Role:* Contributed to dataset building strategy and conducted complete literature survey for getting relevant data sources to ensure diversity in layout and domain coverage.

- **NOTES-BANK [193]**: A benchmark for *evidence-grounded, multimodal reasoning* over handwritten scientific notes, introducing tasks such as evidence-based VQA and open-domain VQA with visual grounding. *Role:* Conceived the benchmark design, defined task specifications, supervised data annotation workflows, and developed evaluation protocols for multimodal reasoning.

These collaborative efforts not only provided critical resources for the experiments presented in this thesis but also enriched the broader research ecosystem—enabling new research directions, facilitating fair comparisons, and establishing stronger baselines for layout-aware document understanding. In particular, NOTES-BANK directly connects to the grand challenge outlined in the following section, serving as a foundational step toward multimodal reasoning systems that can navigate the rich, unstructured landscapes of handwritten and symbol-heavy documents.

International Workshops and Competitions Organized

In addition to technical and collaborative research contributions, this doctoral work has also involved active community-building through the organization of international workshops and competitions in the Document AI field.

- **ICDAR 2023 – DUDE Competition:** Served as a primary organizer for the *Document Understanding in Diverse Environments (DUDE)* competition, hosted at the International Conference on Document Analysis and Recognition (ICDAR) 2023. The competition introduced a challenging multi-domain, multi-paged benchmark to evaluate document understanding systems under diverse layout and modality conditions, attracting global participation from both academia and industry.
- **ICCV 2025 - GDUG Workshop:** I am currently serving as a lead organizer for the Workshop on Graphic Design Understanding and Generation (GDUG), to be held October 19, 2025, in Honolulu, in conjunction with ICCV 2025. The workshop brings together researchers, creators, and practitioners to explore how AI can bridge the gap between generative approaches and the realities of structured graphic design workflows where compositions are built from layers, typography, styles, and visual grammar—rather than pixel-based painting. The discussion will include topics such as multimodal document understanding and generation, layout modeling, typography analysis, perceptual evaluation of design, and AI-assisted creative workflows

Bibliography

[1] Mudit Agrawal and David Doermann. Voronoi++: A dynamic page segmentation approach based on voronoi and docstrum features. In *Proceedings of the International Conference on Document Analysis and Recognition*, pages 1011–1015, 2009.

[2] Stephan Alaniz, Massimiliano Mancini, Anjan Dutta, Diego Marcos, and Zeynep Akata. Abstracting sketches through simple primitives. In *eccv*, pages 396–412. Springer, 2022.

[3] Fawaz Khaled Alarfaj, Iqra Malik, Hikmat Ullah Khan, Naif Almusallam, Muhammad Ramzan, and Muzamil Ahmed. Credit card fraud detection using state-of-the-art machine learning and deep learning algorithms. *Ieee Access*, 10:39700–39715, 2022.

[4] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022.

[5] Apostolos Antonacopoulos, David Bridson, Christos Papadopoulos, and Stefan Pletschacher. A realistic dataset for performance evaluation of document layout analysis. In *2009 10th International Conference on Document Analysis and Recognition*, pages 296–300. Ieee, 2009.

[6] Srikar Appalaraju, Bhavan Jasani, Bhargava Urala Kota, Yusheng Xie, and R Manmatha. Docformer: End-to-end transformer for document understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 993–1003, 2021.

[7] Srikar Appalaraju, Peng Tang, Qi Dong, Nishant Sankaran, Yichu Zhou, and R Manmatha. DocFormerv2: Local Features for Document Understanding. *arXiv preprint arXiv:2306.01733*, 2023.

[8] Diego Martin Arroyo, Janis Postels, and Federico Tombari. Variational transformer networks for layout generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13642–13652, 2021.

[9] Abedelkadir Asi, Rafi Cohen, Klara Kedem, and Jihad El-Sana. Simplifying the reading of historical manuscripts. In *Proceedings of the International Conference on Document Analysis and Recognition*, pages 826–830. IEEE, 2015.

[10] Micheal Baechler, Marcus Liwicki, and Rolf Ingold. Text line extraction using dmlp classifiers for historical manuscripts. In *2013 12th International Conference on Document Analysis and Recognition*, pages 1029–1033. IEEE, 2013.

[11] Souhail Bakkali, Sanket Biswas, Zuheng Ming, Mickaël Coustaty, Marçal Rusiñol, Oriol Ramos Terrades, and Josep Lladós. Globaldoc: A cross-modal vision-language framework for real-world document image retrieval and classification. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1436–1446. IEEE, 2025.

[12] Souhail Bakkali, Zuheng Ming, Mickael Coustaty, Marçal Rusiñol, and Oriol Ramos Terrades. VLCDoC: Vision-Language contrastive pre-training model for cross-Modal document classification. *Pattern Recognition*, 139:109419, 2023.

[13] Ayan Banerjee, Sanket Biswas, Josep Lladós, and Umapada Pal. Swindocsegmenter: an end-to-end unified domain adaptive transformer for document instance segmentation. In *International Conference on Document Analysis and Recognition*, pages 307–325. Springer, 2023.

[14] Ayan Banerjee, Sanket Biswas, Josep Lladós, and Umapada Pal. Graphkd: Exploring knowledge distillation towards document object detection with structured graph creation. In *International Conference on Document Analysis and Recognition*, pages 354–373. Springer, 2024.

[15] Ayan Banerjee, Sanket Biswas, Josep Lladós, and Umapada Pal. Semidocseg: harnessing semi-supervised learning for document layout analysis. *International Journal on Document Analysis and Recognition (IJDAR)*, 27(3):317–334, 2024.

[16] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. BEiT: BERT Pre-Training of Image Transformers. In *International Conference on Learning Representations*, 2022.

[17] Amir Bar, Xin Wang, Vadim Kantorov, Colorado J Reed, Roei Herzig, Gal Chechik, Anna Rohrbach, Trevor Darrell, and Amir Globerson. Detreg: Unsupervised pretraining with region priors for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[18] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. *Advances in neural information processing systems*, 32, 2019.

[19] Ankan Kumar Bhunia, Salman Khan, Hisham Cholakkal, Rao Muhammad Anwer, Fahad Shahbaz Khan, Jorma Laaksonen, and Michael Felsberg. Doodleformer: Creative sketch drawing with transformers. In *European Conference on Computer Vision*, pages 338–355. Springer, 2022.

[20] Ayan Kumar Bhunia, Ayan Das, Umar Riaz Muhammad, Yongxin Yang, Timothy M Hospedales, Tao Xiang, Yulia Gryaditskaya, and Yi-Zhe Song. Pixelor: A competitive sketching ai agent. so you think you can sketch? *ACM Transactions on Graphics (TOG)*, 39(6):1–15, 2020.

[21] Irving Biederman. Recognition-by-components: a theory of human image understanding. *Psychological review*, 94(2):115, 1987.

[22] Nil Biescas, Carlos Boned, Josep Lladós, and Sanket Biswas. Geocontrastnet: Contrastive key-value edge learning for language-agnostic document understanding. In *International Conference on Document Analysis and Recognition*, pages 294–310. Springer, 2024.

[23] Galal M Binmakhashen and Sabri A Mahmoud. Document layout analysis: A comprehensive survey. *ACM Computing Surveys*, 52(6):1–36, 2019.

[24] Sanket Biswas, Ayan Banerjee, Josep Lladós, and Umapada Pal. Docsegtr: an instance-level end-to-end document image segmentation transformer. *arXiv preprint arXiv:2201.11438*, 2022.

[25] Sanket Biswas, Rajiv Jain, Vlad I Morariu, Jiuxiang Gu, Puneet Mathur, Curtis Wigington, Tong Sun, and Josep Lladós. Docsynthv2: A practical autoregressive modeling for document generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8148–8153, 2024.

[26] Sanket Biswas, Pau Riba, Josep Lladós, and Umapada Pal. Beyond document object detection: instance-level segmentation of complex layouts. *International Journal on Document Analysis and Recognition (IJDAR)*, 24(3):269–281, 2021.

[27] Sanket Biswas, Pau Riba, Josep Lladós, and Umapada Pal. Docsynth: a layout guided approach for controllable document image synthesis. In *International Conference on Document Analysis and Recognition*, pages 555–568. Springer, 2021.

[28] Lukas Blecher, Guillem Cucurull, Thomas Scialom, and Robert Stojnic. Nougat: Neural optical understanding for academic documents. *arXiv preprint arXiv:2308.13418*, 2023.

[29] Łukasz Borchmann, Michał Pietruszka, Tomasz Stanislawek, Dawid Jurkiewicz, Michał Turski, Karolina Szyndler, and Filip Graliński. DUE: End-to-End Document Understanding Benchmark. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.

[30] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[31] Yunning Cao, Ye Ma, Min Zhou, Chuanbin Liu, Hongtao Xie, Tiezheng Ge, and Yuning Jiang. Geometry aligned variational transformer for image-conditioned layout generation. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 1561–1571, 2022.

[32] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.

[33] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in neural information processing systems*, 2020.

[34] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, 2021.

[35] Shang Chai, Liansheng Zhuang, and Fengying Yan. Layoutdm: Transformer-based diffusion model for layout generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18349–18358, 2023.

[36] Arindam Chaudhuri, Krupa Mandaviya, Pratixa Badelia, Soumya K Ghosh, Arindam Chaudhuri, Krupa Mandaviya, Pratixa Badelia, and Soumya K Ghosh. *Optical character recognition systems*. Springer, 2017.

[37] Jian Chen, Ruiyi Zhang, Yufan Zhou, Rajiv Jain, Zhiqiang Xu, Ryan Rossi, and Changyou Chen. Towards aligned layout generation via diffusion model with aesthetic constraints. *ICLR*, 2024.

[38] Jin Chen and Daniel Lopresti. Table detection in noisy off-line handwritten documents. In *Proceedings of the International Conference on Document Analysis and Recognition*, pages 399–403, 2011.

[39] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning*, 2020.

[40] Xi Chen, Xiao Wang, Soravit Changpinyo, Anthony J Piergiovanni, Piotr Padlewski, Daniel Salz, Sebastian Goodman, Adam Grycner, Basil Mustafa, Lucas Beyer, et al. Pali: A jointly-scaled multilingual language-image model. *arXiv preprint arXiv:2209.06794*, 2022.

[41] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021.

[42] Yufan Chen, Jiaming Zhang, Kunyu Peng, Junwei Zheng, Ruiping Liu, Philip Torr, and Rainer Stiefelhagen. Rodla: Benchmarking the robustness of document layout analysis models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15556–15566, 2024.

[43] Christian Clausner, Apostolos Antonacopoulos, and Stefan Pletschacher. Icdar2019 competition on recognition of documents with complex layouts-rdcl2019. In *Proceedings of the International Conference on Document Analysis and Recognition*, pages 1521–1526, 2019.

[44] Neil Cohn. Navigating comics: An empirical and theoretical approach to strategies of reading comic page layouts. *Frontiers in psychology*, 4:186, 2013.

[45] Patricia Craja, Alisa Kim, and Stefan Lessmann. Deep learning for detecting financial statement fraud. *Decision Support Systems*, 139:113421, 2020.

[46] Lei Cui, Yiheng Xu, Tengchao Lv, and Furu Wei. Document ai: Benchmarks, models and applications. *arXiv preprint arXiv:2111.08609*, 2021.

[47] Cheng Da, Chuwei Luo, Qi Zheng, and Cong Yao. Vision Grid Transformer for Document Layout Analysis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19462–19472, 2023.

[48] Zhigang Dai, Bolun Cai, Yugeng Lin, and Junying Chen. Up-detr: Unsupervised pre-training for object detection with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021.

[49] Alloy Das, Sanket Biswas, Ayan Banerjee, Josep Lladós, Umapada Pal, and Saumik Bhattacharya. Harnessing the power of multi-lingual datasets for pre-training: Towards enhancing text spotting performance. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 718–728, 2024.

[50] Alloy Das, Sanket Biswas, Umapada Pal, and Josep Lladós. Diving into the depths of spotting text in multi-domain noisy scenes. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 410–417. IEEE, 2024.

[51] Alloy Das, Sanket Biswas, Prasun Roy, Subhankar Ghosh, Umapada Pal, Michael Blumenstein, Josep Lladós, and Saumik Bhattacharya. Faster: A font-agnostic scene text editing and rendering framework. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1944–1954. IEEE, 2025.

[52] Brian Davis, Bryan Morse, Scott Cohen, Brian Price, and Chris Tensmeyer. Deep visual template-free form parsing. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 134–141. IEEE, 2019.

[53] Brian Davis, Bryan Morse, Brian Price, Chris Tensmeyer, Curtis Wigington, and Vlad Morariu. End-to-end document recognition and understanding with dessurt. In *Computer Vision–ECCV 2022 Workshops: Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part IV*, pages 280–296. Springer, 2023.

[54] Brian Davis, Bryan Morse, Brian Price, Chris Tensmeyer, and Curtis Wiginton. Visual fudge: Form understanding via dynamic graph editing. In *International Conference on Document Analysis and Recognition*, pages 416–431. Springer, 2021.

[55] Lluís-Pere de las Heras, Oriol Ramos Terrades, Sergi Robles, and Gemma Sánchez. Cvc-fp and sgt: a new database for structural floor plan analysis and its groundtruthing tool. *International Journal on Document Analysis and Recognition (IJDAR)*, 18:15–30, 2015.

[56] Axel De Nardin, Silvia Zottin, Matteo Paier, Gian Luca Foresti, Emanuela Colombi, and Claudio Piciarelli. Efficient few-shot learning for pixel-precise handwritten document layout analysis. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3680–3688, 2023.

[57] Diana Dee-Lucas and Jill H Larkin. Learning from electronic texts: Effects of interactive overviews for information access. *Cognition and instruction*, 13(3):431–468, 1995.

[58] Mathieu Delalandre, Ernest Valveny, Tony Pridmore, and Dimosthenis Karatzas. Generation of synthetic documents for performance evaluation of symbol recognition & spotting systems. *International Journal on Document Analysis and Recognition (IJDAR)*, 13(3):187–207, 2010.

[59] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[60] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Naacl-hlt (1)*, January 2019.

[61] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[62] Yihao Ding, Zhe Huang, Runlin Wang, YanHang Zhang, Xianru Chen, Yuzhong Ma, Hyunsuk Chung, and Soyeon Caren Han. V-Doc: Visual questions answers with Documents. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21492–21498, 2022.

[63] David Doermann. The indexing and retrieval of document images: A survey. *Computer vision and image understanding*, 70(3):287–298, 1998.

[64] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[65] Mathias Eitz, James Hays, and Marc Alexa. How do humans sketch objects? *ACM Transactions on graphics (TOG)*, 31(4):1–10, 2012.

[66] Mathias Eitz, Kristian Hildebrand, Tamy Boubekeur, and Marc Alexa. Sketch-based image retrieval: Benchmark and bag-of-features descriptors. *IEEE transactions on visualization and computer graphics*, 17(11):1624–1636, 2010.

[67] Vebjørn Ekroll, Bilge Sayim, and Johan Wagemans. The other side of magic: The psychology of perceiving hidden things. *Perspectives on Psychological Science*, 12(1):91–106, 2017.

[68] Jeffrey L Elman. Finding structure in time. *Cognitive science*, 14(2):179–211, 1990.

[69] Qi Fan, Wei Zhuo, Chi-Keung Tang, and Yu-Wing Tai. Few-shot object detection with attention-rpn and multi-relation detector. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4013–4022, 2020.

[70] Jing Fang, Liangcai Gao, Kun Bai, Ruiheng Qiu, Xin Tao, and Zhi Tang. A table detection method for multipage pdf documents via visual seperators and tabular structures. In *Proceedings of the International Conference on Document Analysis and Recognition*, pages 779–783, 2011.

[71] Tahani Fennir, Bart Lamiroy, and Jean-Charles Lamirel. Using gans for domain adaptive high resolution synthetic document generation. In *International Conference on Document Analysis and Recognition*, pages 49–61. Springer, 2023.

[72] Tsu-Jui Fu, William Yang Wang, Daniel McDuff, and Yale Song. Doc2ppt: Automatic presentation slides generation from scientific documents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 634–642, 2022.

[73] Akshay Gadi Patil, Omri Ben-Eliezer, Or Perel, and Hadar Averbuch-Elor. Read: Recursive autoencoders for document layout generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 544–545, 2020.

[74] Yaroslav Ganin, Sergey Bartunov, Yujia Li, Ethan Keller, and Stefano Saliceti. Computer-aided design as language. *Advances in Neural Information Processing Systems*, 34:5885–5897, 2021.

[75] Łukasz Garncarek, Rafał Powalski, Tomasz Stanisławek, Bartosz Topolski, Piotr Halama, Michał Turski, and Filip Graliński. Lambert: Layout-aware language modeling for information extraction. In *International Conference on Document Analysis and Recognition*, pages 532–547. Springer, 2021.

[76] Łukasz Garncarek, Rafał Powalski, Tomasz Stanisławek, Bartosz Topolski, Piotr Halama, Michał Turski, and Filip Graliński. Lambert: Layout-aware language modeling for information extraction. In *International conference on document analysis and recognition*, pages 532–547. Springer, 2021.

[77] Cheryl Geisler. Textual objects: Accounting for the role of texts in the everyday life of complex organizations. *Written communication*, 18(3):296–325, 2001.

[78] Andrea Gemelli, Sanket Biswas, Enrico Civitelli, Josep Lladós, and Simone Marinai. Doc2graph: a task agnostic document understanding framework based on graph neural networks. In *European Conference on Computer Vision*, pages 329–344. Springer, 2022.

[79] Andrea Gemelli, Emanuele Vivoli, and Simone Marinai. Graph neural networks and representation embedding for table extraction in PDF documents. In *accepted for publication at ICPR22*, 2022.

[80] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Networks. In *NeurIPS*, pages 2672—-2680, 2014.

[81] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020.

[82] Jiuxiang Gu, Jason Kuen, Vlad I Morariu, Handong Zhao, Rajiv Jain, Nikolaos Barmpalios, Ani Nenkova, and Tong Sun. Unidoc: Unified pretraining framework for document understanding. *Advances in Neural Information Processing Systems*, 34:39–50, 2021.

[83] Jiuxiang Gu, Xiangxi Shi, Jason Kuen, Lu Qi, Ruiyi Zhang, Anqi Liu, Ani Nenkova, and Tong Sun. Adopd: A large-scale document page decomposition dataset. In *The Twelfth International Conference on Learning Representations*, 2024.

[84] Zhangxuan Gu, Changhua Meng, Ke Wang, Jun Lan, Weiqiang Wang, Ming Gu, and Liqing Zhang. XYLayoutLM: Towards Layout-Aware Multimodal Networks for Visually-Rich Document Understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4583–4592, June 2022.

[85] Ruohao Guo, Dantong Niu, Liao Qu, and Zhenbo Li. Sotr: Segmenting objects with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7157–7166, 2021.

[86] Kamal Gupta, Justin Lazarow, Alessandro Achille, Larry S Davis, Vijay Mahadevan, and Abhinav Shrivastava. Layouttransformer: Layout generation and completion with self-attention. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1004–1014, 2021.

[87] David Ha and Douglas Eck. A neural representation of sketch drawings. In *ICLR 2018*, 2018. 2018.

[88] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30, 2017.

[89] Haralick. Document image understanding: Geometric and logical layout. In *1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 385–390. Ieee, 1994.

[90] Adam W Harley, Alex Ufkes, and Konstantinos G Derpanis. Evaluation of deep convolutional nets for document image classification and retrieval. In *International Conference on Document Analysis and Recognition (ICDAR)*.

[91] Adam W Harley, Alex Ufkes, and Konstantinos G Derpanis. Evaluation of deep convolutional nets for document image classification and retrieval. In *Proceedings of the International Conference on Document Analysis and Recognition*, pages 991–995, 2015.

[92] Dafang He, Scott Cohen, Brian Price, Daniel Kifer, and C Lee Giles. Multi-scale multi-task fcn for semantic page segmentation and table detection. In *Proceedings of the International Conference on Document Analysis and Recognition*, volume 1, pages 254–261, 2017.

[93] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[94] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020.

[95] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[96] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[97] Liu He, Yijuan Lu, John Corring, Dinei Florencio, and Cha Zhang. Diffusion-based document layout generation. In *International Conference on Document Analysis and Recognition*, pages 361–378. Springer, 2023.

[98] Sen He, Wentong Liao, Michael Ying Yang, Yongxin Yang, Yi-Zhe Song, Bodo Rosenhahn, and Tao Xiang. Context-aware layout to image generation with enhanced object appearance. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15049–15058, 2021.

[99] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *arXiv preprint arXiv:1706.08500*, 2017.

[100] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[101] Teakgyu Hong, Donghyun Kim, Mingi Ji, Wonseok Hwang, Daehyun Nam, and Sungrae Park. Bros: A pre-trained language model focusing on text and layout for better key information extraction from documents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 10767–10775, 2022.

[102] Wolfgang Horak. Office document architecture and office document interchange formats: Current status of international standardization. *Computer*, 18(10):50–60, 1985.

[103] Xin Huang, Zheng Ge, Zequn Jie, and Osamu Yoshie. Nms by representative region: Towards crowded pedestrian detection by proposal pairing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10750–10759, 2020.

[104] Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, and Furu Wei. LayoutLMv3: Pre-training for Document AI with Unified Text and Image Masking. *ACM International Conference on Multimedia*, pages 4083–4091, 2022.

[105] Zhaojin Huang, Lichao Huang, Yongchao Gong, Chang Huang, and Xinggang Wang. Mask scoring r-cnn. In *CVPR*, pages 6409–6418, 2019.

[106] Zheng Huang, Kai Chen, Jianhua He, Xiang Bai, Dimosthenis Karatzas, Shijian Lu, and CV Jawahar. Icdar2019 competition on scanned receipt ocr and information extraction. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1516–1520. Ieee, 2019.

[107] Mude Hui, Zhizheng Zhang, Xiaoyi Zhang, Wenxuan Xie, Yuwang Wang, and Yan Lu. Unifying layout generation with a decoupled diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1942–1951, 2023.

[108] Naoto Inoue, Kotaro Kikuchi, Edgar Simo-Serra, Mayu Otani, and Kota Yamaguchi. Layoutdm: Discrete diffusion model for controllable layout generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10167–10176, 2023.

[109] Naoto Inoue, Kotaro Kikuchi, Edgar Simo-Serra, Mayu Otani, and Kota Yamaguchi. Towards flexible multi-modal document models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14287–14296, 2023.

[110] Mohit Iyyer, Varun Manjunatha, Anupam Guha, Yogarshi Vyas, Jordan Boyd-Graber, Hal Daume, and Larry S Davis. The amazing mysteries of the gutter: Drawing inferences between panels in comic book narratives. In *Proceedings of the IEEE Conference on Computer Vision and Pattern recognition*, pages 7186–7195, 2017.

[111] Ajay Jain, Amber Xie, and Pieter Abbeel. Vectorfusion: Text-to-svg by abstracting pixel-based diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1911–1920, 2023.

[112] Rajiv Jain and Curtis Wigington. Multimodal document image classification. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 71–77. Ieee, 2019.

[113] Guillaume Jaume, Hazim Kemal Ekenel, and Jean-Philippe Thiran. Funsd: A dataset for form understanding in noisy scanned documents. In *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, volume 2, pages 1–6. Ieee, 2019.

[114] Guillaume Jaume, Hazim Kemal Ekenel, and Jean-Philippe Thiran. Funsd: A dataset for form understanding in noisy scanned documents. In *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, volume 2, pages 1–6. IEEE, 2019.

[115] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V Le, Yunhsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. *arXiv preprint arXiv:2102.05918*, 2021.

[116] Zhaoyun Jiang, Jiaqi Guo, Shizhao Sun, Huayu Deng, Zhongkai Wu, Vuksan Mijovic, Zijiang James Yang, Jian-Guang Lou, and Dongmei Zhang. Layoutformer++: Conditional graphic layout generation via constraint serialization and decoding space restriction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18403–18412, 2023.

[117] Justin Johnson, Agrim Gupta, and Li Fei-Fei. Image generation from scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1219–1228, 2018.

[118] Jonas Jongejan, Henry Rowley, Takashi Kawashima, Jongmin Kim, and Nick Fox-Gieg. The quick, draw!-ai experiment. *Mount View, CA, accessed Feb*, 17(2018):4, 2016.

[119] Nicholas Journet, Véronique Eglin, Jean-Yves Ramel, and Rémy Mullot. Text/graphic labelling of ancient printed documents. In *Proceedings of the International Conference on Document Analysis and Recognition*, pages 1010–1014, 2005.

[120] Akash Abdu Jyothi, Thibaut Durand, Jiawei He, Leonid Sigal, and Greg Mori. Layoutvae: Stochastic scene layout generation from a label set. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9895–9904, 2019.

[121] Bingyi Kang, Zhuang Liu, Xin Wang, Fisher Yu, Jiashi Feng, and Trevor Darrell. Few-shot object detection via feature reweighting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8420–8429, 2019.

[122] Le Kang, Jayant Kumar, Peng Ye, Yi Li, and David Doermann. Convolutional neural networks for document image classification. In *2014 22nd international conference on pattern recognition*, pages 3168–3172. Ieee, 2014.

[123] Lei Kang, Pau Riba, Yaxing Wang, Marçal Rusiñol, Alicia Fornés, and Mauricio Villegas. Ganwriting: Content-conditioned generation of styled handwritten word images. In *ECCV*, pages 273–289, 2020.

[124] Dimosthenis Karatzas, Lluis Gomez-Bigorda, Anguelos Nicolaou, Suman Ghosh, Andrew Bagdanov, Masakazu Iwamura, Jiri Matas, Lukas Neumann, Vijay Ramaseshan Chandrasekhar, Shijian Lu, et al. ICDAR 2015 competition on robust reading. In *Proc. of the IEEE International Conference on Document Analysis and Recognition*, pages 1156–1160, 2015.

[125] Rangachar Kasturi, Lawrence O'gorman, and Venu Govindaraju. Document image analysis: A primer. *Sadhana*, 27:3–22, 2002.

[126] Kotaro Kikuchi, Edgar Simo-Serra, Mayu Otani, and Kota Yamaguchi. Constrained graphic layout generation via latent optimization. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 88–96, 2021.

[127] Beomyoung Kim, Joonhyun Jeong, Dongyoon Han, and Sung Ju Hwang. The devil is in the points: Weakly semi-supervised instance segmentation via point-guided mask representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11360–11370, 2023.

[128] Dong-Jin Kim, Xiao Sun, Jinsoo Choi, Stephen Lin, and In So Kweon. Detecting human-object interactions with action co-occurrence priors. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXI 16*, pages 718–736. Springer, 2020.

[129] Geewook Kim, Teakgyu Hong, Moonbin Yim, JeongYeon Nam, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. Ocr-free document understanding transformer. In *European Conference on Computer Vision*, pages 498–517. Springer, 2022.

[130] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[131] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *ICLR*, 2016.

[132] Koichi Kise, Akinori Sato, and Motoi Iwata. Segmentation of page images using the area voronoi diagram. *Computer Vision and Image Understanding*, 70(3):370–382, 1998.

[133] Xiang Kong, Lu Jiang, Huiwen Chang, Han Zhang, Yuan Hao, Haifeng Gong, and Irfan Essa. Blt: Bidirectional layout transformer for controllable layout generation. In *European Conference on Computer Vision*, pages 474–490. Springer, 2022.

[134] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NeurIPS*, pages 1097–1105, 2012.

[135] Brenden M Lake, Ruslan Salakhutdinov, and Joshua B Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.

[136] Marcel Lamott, Yves-Noel Weweler, Adrian Ulges, Faisal Shafait, Dirk Krechel, and Darko Obradovic. Lapdoc: Layout-aware prompting for documents. In *International Conference on Document Analysis and Recognition*, pages 142–159. Springer, 2024.

[137] Chen-Yu Lee, Chun-Liang Li, Timothy Dozat, Vincent Perot, Guolong Su, Nan Hua, Joshua Ainslie, Renshen Wang, Yasuhisa Fujii, and Tomas Pfister. FormNet: Structural Encoding beyond Sequential Modeling in Form Document Information Extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3735–3754, 2022.

[138] Chen-Yu Lee, Chun-Liang Li, Hao Zhang, Timothy Dozat, Vincent Perot, Guolong Su, Xiang Zhang, Kihyuk Sohn, Nikolai Glushnev, Renshen Wang, et al. Formnetv2: Multimodal graph contrastive learning for form document information extraction. *arXiv preprint arXiv:2305.02549*, 2023.

[139] Hsin-Ying Lee, Lu Jiang, Irfan Essa, Phuong B Le, Haifeng Gong, Ming-Hsuan Yang, and Weilong Yang. Neural design network: Graphic layout generation with constraints. In *European conference on computer vision*, pages 491–506. Springer, 2020.

[140] Kenton Lee, Mandar Joshi, Iulia Raluca Turc, Hexiang Hu, Fangyu Liu, Julian Martin Eisenschlos, Urvashi Khandelwal, Peter Shaw, Ming-Wei Chang, and Kristina Toutanova. Pix2struct: Screenshot parsing as pretraining for visual language understanding. In *International Conference on Machine Learning*, pages 18893–18912. PMLR, 2023.

[141] David M Levy. *Scrolling forward: Making sense of documents in the digital age*. Simon and Schuster, 2016.

[142] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*, 2019.

[143] Chenhui Li, Peiying Zhang, and Changbo Wang. Harmonious textual layout generation over natural images via deep aesthetics learning. *IEEE Transactions on Multimedia*, 24:3416–3428, 2021.

[144] Feng Li, Hao Zhang, Huaizhe Xu, Shilong Liu, Lei Zhang, Lionel M Ni, and Heung-Yeung Shum. Mask dino: Towards a unified transformer-based framework for object detection and segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3041–3050, 2023.

[145] Jianan Li, Yunchao Wei, Xiaodan Liang, Jian Dong, Tingfa Xu, Jiashi Feng, and Shuicheng Yan. Attentive contexts for object detection. *IEEE Transactions on Multimedia*, 19(5):944–954, 2016.

[146] Jianan Li, Jimei Yang, Aaron Hertzmann, Jianming Zhang, and Tingfa Xu. Layoutgan: Generating graphic layouts with wireframe discriminators. *arXiv preprint arXiv:1901.06767*, 2019.

[147] Junlong Li, Yiheng Xu, Tengchao Lv, Lei Cui, Cha Zhang, and Furu Wei. Dit: Self-supervised pre-training for document image transformer. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 3530–3539, 2022.

[148] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pages 12888–12900. Pmlr, 2022.

[149] Kai Li, Curtis Wigington, Chris Tensmeyer, Handong Zhao, Nikolaos Barmpalios, Vlad I. Morariu, Varun Manjunatha, Tong Sun, and Yun Fu. Cross-domain document object detection: Benchmark suite and method. In *CVPR*, 2020.

[150] Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, Ming Zhou, and Zhoujun Li. Tablebank: Table benchmark for image-based table detection and recognition. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 1918–1925, 2020.

[151] Minghao Li, Yiheng Xu, Lei Cui, Shaohan Huang, Furu Wei, Zhoujun Li, and Ming Zhou. DocBank: A benchmark dataset for document layout analysis. *arXiv preprint arXiv:2006.01038*, 2020.

[152] Peizhao Li, Jiuxiang Gu, Jason Kuen, Vlad I Morariu, Handong Zhao, Rajiv Jain, Varun Manjunatha, and Hongfu Liu. Selfdoc: Self-supervised document representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5652–5660, 2021.

[153] Xiao-Hui Li, Fei Yin, and Cheng-Lin Liu. Page object detection from pdf document images by deep structured prediction and supervised clustering. In *ICPR*, pages 3627–3632, 2018.

[154] Xin Li, Mingming Gong, Yunfei Wu, Jianxin Dai, Antai Guo, Xinghua Jiang, Haoyu Cao, Yinsong Liu, Deqiang Jiang, and Xing Sun. Dream: Document reconstruction via end-to-end autoregressive model. *arXiv preprint arXiv:2507.05805*, 2025.

[155] Yi Li, Yi-Zhe Song, Shaogang Gong, et al. Sketch recognition by ensemble matching of structured features. In *BMVC*, volume 1, page 2, 2013.

[156] Yujie Li, Pengfei Zhang, Xing Xu, Yi Lai, Fumin Shen, Lijiang Chen, and Pengxiang Gao. Few-shot prototype alignment regularization network for document image layout segementation. *Pattern Recognition*, 115:107882, 2021.

[157] Yulin Li, Yuxi Qian, Yuechen Yu, Xiameng Qin, Chengquan Zhang, Yan Liu, Kun Yao, Junyu Han, Jingtuo Liu, and Errui Ding. Structext: Structured text understanding with multi-modal transformers. In *Proceedings of the 29th ACM international conference on multimedia*, pages 1912–1920, 2021.

[158] Haofu Liao, Aruni RoyChowdhury, Weijian Li, Ankan Bansal, Yuting Zhang, Zhuowen Tu, Ravi Kumar Satzoda, R Manmatha, and Vijay Mahadevan. DocTr: Document transformer for structured information extraction in documents. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19584–19594, 2023.

[159] Hangyu Lin, Yanwei Fu, Xiangyang Xue, and Yu-Gang Jiang. Sketch-bert: Learning sketch bidirectional encoder representation from transformers by self-supervised learning of sketch gestalt. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6758–6767, 2020.

[160] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[161] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[162] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *Proc. of the European Conference on Computer Vision*, pages 740–755. Springer, 2014.

[163] Zening Lin, Jiapeng Wang, Teng Li, Wenhui Liao, Dayi Huang, Longfei Xiong, and Lianwen Jin. Peneo: unifying line extraction, line grouping, and entity linking for end-to-end document pair extraction. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 5171–5180, 2024.

[164] Zening Lin, Jiapeng Wang, Teng Li, Wenhui Liao, Dayi Huang, Longfei Xiong, and Lianwen Jin. Peneo: Unifying line extraction, line grouping, and entity linking for end-to-end document pair extraction. In *Proceedings of the 32nd ACM International Conference on Multimedia*, MM '24, page 5171–5180, New York, NY, USA, 2024. Association for Computing Machinery.

[165] Li Liu, Zhiyu Wang, Taorong Qiu, Qiu Chen, Yue Lu, and Ching Y Suen. Document image classification: Progress over two decades. *Neurocomputing*, 453:223–240, 2021.

[166] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. SSD: Single shot multibox detector. In *Proc. of the European Conference on Computer Vision*, pages 21–37. Springer, 2016.

[167] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[168] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12009–12019, 2022.

[169] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, 2021.

[170] Nikolaos Livathinos, Christoph Auer, Maksym Lysak, Ahmed Nassar, Michele Dolfi, Panos Vagenas, Cesar Berrospi Ramis, Matteo Omenetti, Kasper Dinkla, Yusik Kim, et al. Docling: An efficient open-source toolkit for ai-driven document conversion. *arXiv preprint arXiv:2501.17887*, 2025.

[171] Josep Lladós, Ernest Valveny, Gemma Sánchez, and Enric Marti. Symbol recognition: Current advances and perspectives. In *International Workshop on Graphics Recognition*, pages 104–128. Springer, 2001.

[172] Chuwei Luo, Changxu Cheng, Qi Zheng, and Cong Yao. GeoLayoutLM: Geometric Pre-training for Visual Information Extraction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7092–7101, 2023.

[173] Tengchao Lv, Yupan Huang, Jingye Chen, Yuzhong Zhao, Yilin Jia, Lei Cui, Shuming Ma, Yaoyao Chang, Shaohan Huang, Wenhui Wang, et al. Kosmos-2.5: A multimodal literate model. *arXiv preprint arXiv:2309.11419*, 2023.

[174] Subhajit Maity, Sanket Biswas, Siladittya Manna, Ayan Banerjee, Josep Lladós, Saumik Bhattacharya, and Umapada Pal. Selfdocseg: A self-supervised vision-based approach towards document segmentation. In *International Conference on Document Analysis and Recognition*, pages 342–360. Springer, 2023.

[175] Simone Marinai, Marco Gori, and Giovanni Soda. Artificial neural networks for document analysis and recognition. *IEEE Transactions on pattern analysis and machine intelligence*, 27(1):23–35, 2005.

[176] Logan Markewich, Hao Zhang, Yubin Xing, Navid Lambert-Shirzad, Zhexin Jiang, Roy Ka-Wei Lee, Zhi Li, and Seok-Bum Ko. Segmentation for document layout analysis: not dead yet. *International Journal on Document Analysis and Recognition (IJDAR)*, 25(2):67–77, 2022.

[177] U-V Marti and Horst Bunke. The iam-database: an english sentence database for offline handwriting recognition. *International journal on document analysis and recognition*, 5:39–46, 2002.

[178] Joan Mas, Josep Lladós, Gemma Sánchez, and Joaquim Armando Pires Jorge. A syntactic approach based on distortion-tolerant adjacency grammars and a spatial-directed parser to interpret sketched diagrams. *Pattern Recognition*, 43(12):4148–4164, 2010.

[179] Minesh Mathew, Viraj Bagal, Rubèn Tito, Dimosthenis Karatzas, Ernest Valveny, and CV Jawahar. InfographicVQA. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1697–1706, 2022.

[180] Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. Docvqa: A dataset for vqa on document images. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 2200–2209, 2021.

[181] Minesh Mathew, Ruben Tito, Dimosthenis Karatzas, R Manmatha, and CV Jawahar. Document visual question answering challenge 2020. *arXiv preprint arXiv:2008.08899*, 2020.

[182] Puneet Mathur, Rajiv Jain, Jiuxiang Gu, Franck Dernoncourt, Dinesh Manocha, and Vlad I Morariu. DocEdit: language-guided document editing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 1914–1922, 2023.

[183] Souparni Mazumder, Sanket Biswas, Alloy Das, and Josep Lladós. Doc2graph-x: A multilingual graph-based framework for form understanding. In *International Workshop on Graph-Based Representations in Pattern Recognition*, pages 257–266. Springer, 2025.

[184] Ahmed Nassar, Andres Marafioti, Matteo Omenetti, Maksym Lysak, Nikolaos Livathinos, Christoph Auer, Lucas Morin, Rafael Teixeira de Lima, Yusik Kim, A Said Gurbuz, et al. Smoldocling: An ultra-compact vision-language model for end-to-end multi-modal document conversion. *International Conference on Computer Vision*, 2025.

[185] Nibal Nayef, Fei Yin, Imen Bizid, Hyunsoo Choi, Yuan Feng, Dimosthenis Karatzas, Zhenbo Luo, Umapada Pal, Christophe Rigaud, Joseph Chazalon, et al. Icdar2017 robust reading challenge on multi-lingual scene text detection and script identification-rrc-mlt. In *2017 14th IAPR international conference on document analysis and recognition (ICDAR)*, volume 1, pages 1454–1459. IEEE, 2017.

[186] Konstantina Nikolaidou, Mathias Seuret, Hamam Mokayed, and Marcus Liwicki. A survey of historical document image datasets. *International Journal on Document Analysis and Recognition (IJDAR)*, 25(4):305–338, 2022.

[187] Peter O'Donovan, Aseem Agarwala, and Aaron Hertzmann. Designscape: Design with interactive layout suggestions. In *Proceedings of the 33rd annual ACM conference on human factors in computing systems*, pages 1221–1224, 2015.

[188] Lawrence O'Gorman. The document spectrum for bottom-up page layout analysis. In *Advances in structural and syntactic pattern recognition*, pages 270–279. 1992.

[189] Lawrence O'Gorman and Rangachar Kasturi. *Document image analysis*, volume 39. IEEE Computer Society Press Los Alamitos, 1995.

[190] Sofia Ares Oliveira, Benoit Seguin, and Frederic Kaplan. dhsegment: A generic deep-learning approach for document segmentation. In *2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 7–12. IEEE, 2018.

[191] R OpenAI. GPT-4 technical report. *arXiv*, pages 2303–08774, 2023.

[192] Peter O'Donovan, Aseem Agarwala, and Aaron Hertzmann. Learning layouts for single-pagegraphic designs. *IEEE transactions on visualization and computer graphics*, 20(8):1200–1213, 2014.

[193] Aniket Pal, Sanket Biswas, Alloy Das, Ayush Lodh, Priyanka Banerjee, Soumitri Chattopadhyay, Dimosthenis Karatzas, Josep Llados, and CV Jawahar. Notesbank: Benchmarking neural transcription and search for scientific notes understanding. *arXiv preprint arXiv:2504.09249*, 2025.

[194] Akshay Gadi Patil, Omri Ben-Eliezer, Or Perel, and Hadar Averbuch-Elor. Read: Recursive autoencoders for document layout generation. In *CVPRW*, pages 544–545, 2020.

[195] Akshay Gadi Patil, Manyi Li, Matthew Fisher, Manolis Savva, and Hao Zhang. Layoutgmn: Neural graph matching for structural layout similarity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11048–11057, 2021.

[196] Qiming Peng, Yinxu Pan, Wenjin Wang, Bin Luo, Zhenyu Zhang, Zhengjie Huang, Teng Hu, Weichong Yin, Yongfeng Chen, Yin Zhang, et al. Ernie-layout: Layout knowledge enhanced pre-training for visually-rich document understanding. *arXiv preprint arXiv:2210.06155*, 2022.

[197] Birgit Pfitzmann, Christoph Auer, Michele Dolfi, Ahmed S Nassar, and Peter Staar. DocLayNet: A Large Human-Annotated Dataset for Document-Layout Segmentation. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3743–3751, 2022.

[198] Réjean Plamondon and Sargur N Srihari. Online and off-line handwriting recognition: a comprehensive survey. *IEEE Transactions on pattern analysis and machine intelligence*, 22(1):63–84, 2002.

[199] Rafał Powalski, Łukasz Borchmann, Dawid Jurkiewicz, Tomasz Dwojak, Michał Pietruszka, and Gabriela Pałka. Going full-tilt boogie on document understanding with text-image-layout transformer. In *Document Analysis and Recognition–ICDAR 2021: 16th International Conference, Lausanne, Switzerland, September 5–10, 2021, Proceedings, Part II 16*, pages 732–747. Springer, 2021.

[200] Ioannis Pratikakis, Konstantinos Zagoris, George Barlas, and Basilis Gatos. Icdar2017 competition on document image binarization (dibco 2017). In *2017 14Th IAPR international conference on document analysis and recognition (ICDAR)*, volume 1, pages 1395–1403. IEEE, 2017.

[201] Shah Rukh Qasim, Hassan Mahmood, and Faisal Shafait. Rethinking table recognition using graph neural networks. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 142–147. IEEE, 2019.

[202] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

[203] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. 2018.

[204] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[205] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020.

[206] Jean-Yves Ramel, Stéphane Leriche, Marie-Luce Demonet, and Sébastien Busson. User-driven page layout analysis of historical printed books. *International Journal of Document Analysis and Recognition (IJDAR)*, 9(2-4):243–261, 2007.

[207] Joseph Redmon and Ali Farhadi. YOLO9000: better, faster, stronger. *arXiv preprint arXiv:1612.08242*, 2016.

[208] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017.

[209] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019.

[210] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Proc. of the International Conference on Neural Information Processing Systems*, pages 91–99, 2015.

[211] Pau Riba, Anjan Dutta, Lutz Goldmann, Alicia Fornés, Oriol Ramos, and Josep Lladós. Table detection in invoice documents by graph neural networks. In *Proceedings of the International Conference on Document Analysis and Recognition*, 2019.

[212] Pau Riba, Lutz Goldmann, Oriol Ramos Terrades, Diede Rusticus, Alicia Fornés, and Josep Lladós. Table detection in business document images by message passing networks. *Pattern Recognition*, 127:108641, 2022.

[213] Leo Sampaio Ferraz Ribeiro, Tu Bui, John Collomosse, and Moacir Ponti. Sketchformer: Transformer-based representation for sketched structure. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14153–14162, 2020.

[214] Richard J Roberts. Pubmed central: The genbank of the published literature, 2001.

[215] Juan Rodriguez, Xiangru Jian, Siba Smarak Panigrahi, Tianyu Zhang, Aarash Feizi, Abhay Puri, Akshay Kalkunte, François Savard, Ahmed Masry, Shravan Nayak, et al. Bigdocs: An open dataset for training multimodal models on document and code tasks. *arXiv preprint arXiv:2412.04626*, 2024.

[216] Juan A Rodriguez, Shubham Agarwal, Issam H Laradji, Pau Rodriguez, David Vazquez, Christopher Pal, and Marco Pedersoli. Starvector: Generating scalable vector graphics code from images. *arXiv preprint arXiv:2312.11556*, 2023.

[217] Marçal Rusiñol, Agnés Borràs, and Josep Lladós. Relational indexing of vectorial primitives for symbol spotting in line-drawing images. *Pattern Recognition Letters*, 31(3):188–201, 2010.

[218] Patsorn Sangkloy, Nathan Burnell, Cusuh Ham, and James Hays. The sketchy database: learning to retrieve badly drawn bunnies. *ACM Transactions on Graphics (TOG)*, 35(4):1–12, 2016.

[219] Rosália G Schneider and Tinne Tuytelaars. Sketch classification and classification-driven analysis using fisher vectors. *ACM Transactions on graphics (TOG)*, 33(6):1–9, 2014.

[220] Sebastian Schreiber, Stefan Agne, Ivo Wolf, Andreas Dengel, and Sheraz Ahmed. Deepdesrt: Deep learning for detection and structure recognition of tables in document images. In *Proceedings of the International Conference on Document Analysis and Recognition*, volume 1, pages 1162–1167, 2017.

[221] Adriana Schulz, Ariel Shamir, David IW Levin, Pitchaya Sitthi-Amorn, and Wojciech Matusik. Design and fabrication by example. *ACM Transactions on Graphics (TOG)*, 33(4):1–11, 2014.

[222] Anna Scius-Bertrand, Atefeh Fakhari, Lars Vögtlin, Daniel Ribeiro Cabral, and Andreas Fischer. Are layout analysis and ocr still useful for document information extraction using foundation models? In *International Conference on Document Analysis and Recognition*, pages 175–191. Springer, 2024.

[223] Faisal Shafait and Ray Smith. Table detection in heterogeneous documents. In *Proceedings of the International Workshop on Document Analysis Systems*, pages 65–72, 2010.

[224] Zejiang Shen, Kaixuan Zhang, and Melissa Dell. A large dataset of historical japanese documents with complex layouts. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 548–549, 2020.

[225] Zejiang Shen, Ruochen Zhang, Melissa Dell, Benjamin Charles Germain Lee, Jacob Carlson, and Weining Li. Layoutparser: A unified toolkit for deep learning based document image analysis. In *International Conference on Document Analysis and Recognition*, pages 131–146. Springer, 2021.

[226] Štěpán Šimsa, Milan Šulc, Michal Uřičář, Yash Patel, Ahmed Hamdi, Matěj Kocián, Matyáš Skalickỳ, Jiří Matas, Antoine Doucet, Mickaël Coustaty, et al. DocILE Benchmark for Document Information Localization and Extraction. *arXiv preprint arXiv:2302.05658*, 2023.

[227] Ray Smith. An overview of the tesseract ocr engine. In *Ninth international conference on document analysis and recognition (ICDAR 2007)*, volume 2, pages 629–633. IEEE, 2007.

[228] Richard Socher, Cliff C Lin, Chris Manning, and Andrew Y Ng. Parsing natural scenes and natural language with recursive neural networks. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 129–136, 2011.

[229] Jifei Song, Qian Yu, Yi-Zhe Song, Tao Xiang, and Timothy M Hospedales. Deep spatial-semantic attention for fine-grained sketch-based image retrieval. In *Proceedings of the IEEE international conference on computer vision*, pages 5551–5560, 2017.

[230] Mohamed Ali Souibgui, Sanket Biswas, Sana Khamekhem Jemni, Yousri Kessentini, Alicia Fornés, Josep Lladós, and Umapada Pal. Docentr: An end-to-end document image enhancement transformer. In *2022 26th International Conference on Pattern Recognition (ICPR)*, pages 1699–1705. IEEE, 2022.

[231] Mohamed Ali Souibgui, Sanket Biswas, Andres Mafla, Ali Furkan Biten, Alicia Fornés, Yousri Kessentini, Josep Lladós, Lluis Gomez, and Dimosthenis Karatzas. Text-diae: a self-supervised degradation invariant autoencoder for text recognition and document enhancement. In *proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 2330–2338, 2023.

[232] Mohamed Ali Souibgui, Changkyu Choi, Andrey Barsky, Kangsoo Jung, Ernest Valveny, and Dimosthenis Karatzas. Docvxqa: Context-aware visual explanations for document question answering. *International Conference on Machine Learning*, 2025.

[233] Tomasz Stanislawek, Filip Gralinski, Anna Wróblewska, Dawid Lipinski, Agnieszka Kaliska, Paulina Rosalska, Bartosz Topolski, and Przemyslaw Biecek. Kleister: Key Information Extraction Datasets Involving Long Documents with Complex Layouts. In *Icdar*, volume 12821 of *Lecture Notes in Computer Science*, pages 564–579. Springer, 2021.

[234] John Sweller. Cognitive load during problem solving: Effects on learning. *Cognitive science*, 12(2):257–285, 1988.

[235] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *CVPR*, pages 2818–2826, 2016.

[236] Ryota Tanaka, Kyosuke Nishida, and Sen Yoshida. Visualmrc: Machine reading comprehension on document images. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 13878–13888, 2021.

[237] Yuan Y Tang, Seong-Whan Lee, and Ching Y Suen. Automatic document processing: a survey. *Pattern recognition*, 29(12):1931–1952, 1996.

[238] Zineng Tang, Ziyi Yang, Guoxin Wang, Yuwei Fang, Yang Liu, Chenguang Zhu, Michael Zeng, Cha Zhang, and Mohit Bansal. Unifying vision, text, and layout for universal document processing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19254–19264, 2023.

[239] Benjamin W Tatler, Roland J Baddeley, and Iain D Gilchrist. Visual correlates of fixation selection: Effects of scale and time. *Vision research*, 45(5):643–659, 2005.

[240] Rubèn Tito, Dimosthenis Karatzas, and Ernest Valveny. Document collection visual question answering. In *Document Analysis and Recognition–ICDAR 2021: 16th International Conference, Lausanne, Switzerland, September 5–10, 2021, Proceedings, Part II 16*, pages 778–792. Springer, 2021.

[241] Rubèn Tito, Minesh Mathew, CV Jawahar, Ernest Valveny, and Dimosthenis Karatzas. Icdar 2021 competition on document visual question answering. In *International Conference on Document Analysis and Recognition*, pages 635–649. Springer, 2021.

[242] Adarsh Tiwari, Sanket Biswas, and Josep Lladós. Can pre-trained language models help in understanding handwritten symbols? In *International Conference on Document Analysis and Recognition*, pages 199–211. Springer, 2023.

[243] Adarsh Tiwari, Sanket Biswas, and Josep Lladós. Sketchgpt: Autoregressive modeling for sketch generation and recognition. In *International Conference on Document Analysis and Recognition*, pages 421–438. Springer, 2024.

[244] Tuan Anh Tran, In-Seop Na, and Soo-Hyung Kim. Hybrid page segmentation using multilevel homogeneity structure. In *Proceedings of the 9th International Conference on Ubiquitous Information Management and Communication*, pages 1–6, 2015.

[245] Scott Tupaj, Zhongwen Shi, C Hwa Chang, and Hassan Alam. Extracting tabular information from text files. *EECS Department, Tufts University, Medford, USA*, 1996.

[246] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(11), 2008.

[247] Jordy Van Landeghem, Lukasz Borchmann, Rubèn Tito, Michał Pietruszka, Dawid Jurkiewicz, Rafał Powalski, Paweł Józiak, Sanket Biswas, Mickaël Coustaty, and Tomasz Stanisławek. ICDAR 2023 Competition on Document UnderstanDing of Everything (DUDE). In *Proceedings of the ICDAR 2023*, pages 420–434. Springer, 2023.

[248] Jordy Van Landeghem, Subhajit Maity, Ayan Banerjee, Matthew Blaschko, Marie-Francine Moens, Josep Lladós, and Sanket Biswas. Distildoc: Knowledge distillation for visually-rich document applications. In *International Conference on Document Analysis and Recognition*, pages 195–217. Springer, 2024.

[249] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017.

[250] Emanuele Vivoli, Mohamed Ali Souibgui, Andrey Barsky, Artemis LLabres, Marco Bertini, and Dimosthenis Karatzas. One missing piece in vision and language: A survey on comics understanding. *arXiv preprint arXiv:2409.09502*, 2024.

[251] Bhanu Prakash Voutharoja, Lizhen Qu, and Fatemeh Shiri. Language independent neuro-symbolic semantic parsing for form understanding. *arXiv preprint arXiv:2305.04460*, 2023.

[252] Jiapeng Wang, Lianwen Jin, and Kai Ding. LiLT: A Simple yet Effective Language-Independent Layout Transformer for Structured Document Understanding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7747–7757, 2022.

[253] Jilin Wang, Michael Krumdick, Baojia Tong, Hamima Halim, Maxim Sokolov, Vadym Barda, Delphine Vendryes, and Chris Tanner. A graphical approach to document layout analysis. In Gernot A. Fink, Rajiv Jain, Koichi Kise, and Richard Zanibbi, editors, *Document Analysis and Recognition - ICDAR 2023*, pages 53–69, Cham, 2023. Springer Nature Switzerland.

[254] Lu Wang, Chaoli Wang, Zhanquan Sun, and Sheng Chen. An improved dice loss for pneumothorax segmentation by mining the information of negative areas. *IEEE Access*, 8:167939–167949, 2020.

[255] Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. Ofa: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *International conference on machine learning*, pages 23318–23340. PMLR, 2022.

[256] Qiang Wang, Haoge Deng, Yonggang Qi, Da Li, and Yi-Zhe Song. Sketchknitter: Vectorized sketch generation with diffusion models. In *The Eleventh International Conference on Learning Representations*, 2022.

[257] Wenjin Wang, Yunhao Li, Yixin Ou, and Yin Zhang. Layout and Task Aware Instruction Prompt for Zero-shot Document Image Question Answering. *arXiv preprint arXiv:2306.00526*, 2023.

[258] Xin Wang, Thomas E Huang, Trevor Darrell, Joseph E Gonzalez, and Fisher Yu. Frustratingly simple few-shot object detection. *arXiv preprint arXiv:2003.06957*, 2020.

[259] Xinlong Wang, Rufeng Zhang, Tao Kong, Lei Li, and Chunhua Shen. Solov2: Dynamic and fast instance segmentation. *Advances in Neural information processing systems*, 33:17721–17732, 2020.

[260] Xudong Wang, Rohit Girdhar, Stella X Yu, and Ishan Misra. Cut and learn for unsupervised object detection and instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3124–3134, 2023.

[261] Yizhi Wang, Guo Pu, Wenhan Luo, Yexin Wang, Pengfei Xiong, Hongwen Kang, and Zhouhui Lian. Aesthetic text logo synthesis via content-aware layout inferring. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2436–2445, 2022.

[262] Zhenyu Wang, Yali Li, and Shengjin Wang. Noisy boundaries: Lemon or lemonade for semi-supervised instance segmentation? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16826–16835, 2022.

[263] Zilong Wang, Yiheng Xu, Lei Cui, Jingbo Shang, and Furu Wei. Layoutreader: Pre-training of text and layout for reading order detection. *arXiv preprint arXiv:2108.11591*, 2021.

[264] Zilong Wang, Mingjie Zhan, Houxing Ren, Zhaohui Hou, Yuwei Wu, Xingyan Zhang, and Ding Liang. Grouplink: An end-to-end multitask method for word grouping and relation extraction in form understanding. *arXiv preprint arXiv:2105.04650*, 2021.

[265] Muhammad Waseem, Sanket Biswas, and Josep Lladós. Docedit redefined: In-context learning for multimodal document editing. In *Proceedings of the Winter Conference on Applications of Computer Vision*, pages 1497–1501, 2025.

[266] Max Wertheimer. Laws of organization in perceptual forms. 1938.

[267] Geoffrey F Woodman, Shaun P Vecera, and Steven J Luck. Perceptual organization influences visual working memory. *Psychonomic bulletin & review*, 10(1):80–87, 2003.

[268] Sijin Wu, Dan Zhang, Teng Hu, and Shikun Feng. Docprompt: Large-scale continue pretrain for zero-shot and few-shot document question answering. *arXiv preprint arXiv:2308.10959*, 2023.

[269] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2, 2019.

[270] Zhuofan Xia, Xuran Pan, Shiji Song, Li Erran Li, and Gao Huang. Vision transformer with deformable attention. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4794–4803, 2022.

[271] Yi Xiao and Hong Yan. Location of title and author regions in document images based on the delaunay triangulation. *Image and Vision Computing*, 22(4):319–329, 2004.

[272] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *CVPR*, pages 1492–1500, 2017.

[273] Yang Xu, Yiheng Xu, Tengchao Lv, Lei Cui, Furu Wei, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, Wanxiang Che, et al. Layoutlmv2: Multimodal pre-training for visually-rich document understanding. *arXiv preprint arXiv:2012.14740*, 2020.

[274] Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. Layoutlm: Pre-training of text and layout for document image understanding. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1192–1200, 2020.

[275] Yiheng Xu, Tengchao Lv, Lei Cui, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, and Furu Wei. Layoutxlm: Multimodal pre-training for multilingual visually-rich document understanding. *arXiv preprint arXiv:2104.08836*, 2021.

[276] Yiheng Xu, Tengchao Lv, Lei Cui, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, and Furu Wei. Xfund: A benchmark dataset for multilingual visually rich form understanding. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3214–3224, 2022.

[277] Kota Yamaguchi. Canvasvae: Learning to generate vector graphic documents. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5481–5489, 2021.

[278] Huichen Yang and William Hsu. Transformer-based approach for document layout understanding. In *2022 IEEE International Conference on Image Processing (ICIP)*, pages 4043–4047. IEEE, 2022.

[279] Qixiang Ye and David Doermann. Text detection and recognition in imagery: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 37(7):1480–1500, 2014.

[280] Moonbin Yim, Yoonsik Kim, Han-Cheol Cho, and Sungrae Park. Synthtiger: Synthetic text image generator towards better text recognition models. In *International conference on document analysis and recognition*, pages 109–124. Springer, 2021.

[281] Yang You, Igor Gitman, and Boris Ginsburg. Large batch training of convolutional networks. *arXiv preprint arXiv:1708.03888*, 2017.

[282] Qian Yu, Feng Liu, Yi-Zhe Song, Tao Xiang, Timothy M Hospedales, and Chen-Change Loy. Sketch me that shoe. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 799–807, 2016.

[283] Seonghoon Yu, Paul Hongsuck Seo, and Jeany Son. Zero-shot referring image segmentation with global-local context features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19456–19465, 2023.

[284] Yuechen Yu, Yulin Li, Chengquan Zhang, Xiaoqiang Zhang, Zengyuan Guo, Xiameng Qin, Kun Yao, Junyu Han, Errui Ding, and Jingdong Wang. Structextv2: Masked visual-textual prediction for document image pre-training. *International Conference on Learning Representations*, 2023.

[285] Gongjie Zhang, Zhipeng Luo, Kaiwen Cui, Shijian Lu, and Eric P Xing. Meta-detr: Image-level few-shot detection with inter-class correlation exploitation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[286] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv preprint arXiv:2203.03605*, 2022.

[287] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, pages 586–595, 2018.

[288] Bo Zhao, Weidong Yin, Lili Meng, and Leonid Sigal. Layout2image: Image generation from layout. *IJCV*, 128:2418–2435, 2020.

[289] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 2023.

[290] Xinru Zheng, Xiaotian Qiao, Ying Cao, and Rynson WH Lau. Content-aware generative modeling of graphic design layouts. *ACM TOG*, 38(4):1–15, 2019.

[291] Xu Zhong, Jianbin Tang, and Antonio Jimeno Yepes. Publaynet: largest dataset ever for document layout analysis. In *Proceedings of the International Conference on Document Analysis and Recognition*, pages 1015–1022, 2019.

[292] Zhi-Hua Zhou and Zhi-Hua Zhou. Semi-supervised learning. *Machine Learning*, pages 315–341, 2021.

[293] Chuhang Zou, Alex Colburn, Qi Shan, and Derek Hoiem. Layoutnet: Reconstructing the 3d room layout from a single rgb image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2051–2059, 2018.