




ADVERTIMENT. L'accés als continguts d'aquesta tesi queda condicionat a l'acceptació de les condicions d'ús establertes per la següent llicència Creative Commons:  <https://creativecommons.org/licenses/?lang=ca>

ADVERTENCIA. El acceso a los contenidos de esta tesis queda condicionado a la aceptación de las condiciones de uso establecidas por la siguiente licencia Creative Commons:  <https://creativecommons.org/licenses/?lang=es>

WARNING. The access to the contents of this doctoral thesis it is limited to the acceptance of the use conditions set by the following Creative Commons license:  <https://creativecommons.org/licenses/?lang=en>



**Universitat Autònoma
de Barcelona**

Towards Robustness
in Computer-based
Image Understanding

A dissertation submitted by **Diego Velázquez
Dorta** at Universitat Autònoma de Barcelona to
fulfil the degree of **Doctor of Philosophy**.

Bellaterra, June 15, 2023

Co-Director	Dr. Jordi Gonzalez Sabaté Centre de Visió per Computador Universitat Autònoma de Barcelona
Co-Director	Dr. Josep M. Gonfaus Sitjes Satellogic
Co-Director	Dr. Pau Rodríguez López Apple Research
Thesis committee	Dr. David Vazquez Bermudez ServiceNow Research (Montréal), Canada Dr. Xavier Baró Soler Universitat Oberta de Catalunya (UOC) Dr. Jorge Bernal del Nozal Universitat Autònoma de Barcelona
International evaluators	Dr. Wenjuan Gong China University of Petroleum (East China), China Dr. Alexandre Lacoste ServiceNow Research (Quebec), Canada



This document was typeset by the author using L^AT_EX 2_ε.

The research described in this book was carried out at the Centre de Visió per Computador, Universitat Autònoma de Barcelona. Copyright © 2023 by **Diego Velázquez Dorta**. All rights reserved. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopy, recording, or any information storage and retrieval system, without permission in writing from the author.

ISBN: 978-81-126409-5-3

Printed by Ediciones Gráficas Rey, S.L.

I visualize a time when we will be to robots what dogs are to humans, and I'm
rooting for the machines.
— Claude Shannon

Be thankful that you have a life, and forsake your vain and presumptuous
desires for a second one.
— Richard Dawkins

Para Arnau...

Acknowledgements

It feels like this stage of my life has been going on forever. I suppose that so much excitement, stress, doubts and hard work crammed into so little time can create that illusion. This stage as a PhD student has been the most challenging thing I've ever done and it would have not been possible without the support of many.

I would like to start by thanking Jordi González for being more than a supervisor, a friend, and for giving me the advice I needed to hear the most: "Don't worry about the end results, worry about enjoying the journey instead". Thank you Poal. Thanks to Pau Rodríguez and Josep M. Gonfaus, friends who taught me everything about research when I knew nothing, and without whom this work would definitely not exist. My deepest thanks to Xavier Roca; who from the very beginning and with the utmost patience taught me how to code, for giving me the opportunity to the same for others.

My sincere appreciation the ServiceNow Research team. I especially want to thank Alexandre Lacoste for his generosity and kindness and for giving me the opportunity to work with him. My thanks to all the friends I made there, Alexandre Drouin, Juan Rodriguez, Arjun, Tom, Valentina, Etienne and George for making my stay so enjoyable. Thanks also to Catherine Martin and David Vazquez who made me feel at home.

No words are enough to thank my wife Cristina, who gives me joy when it seems there is none to be found, for supporting me throughout everything and without whom no journey can be enjoyed.

Por último, me gustaría agradecer a mi familia, especialmente a mi madre por estar siempre a mi lado sin importar la distancia a mi hermana por hacer de madre y a mi padre por su continuo sacrificio y estoicismo a lo largo de los años, convirtiéndome en la persona que soy hoy.

Abstract

This thesis embarks on an exploratory journey into robustness in deep learning, with a keen focus on the intertwining facets of generalization, explainability, and edge cases within the realm of computer vision. In deep learning, robustness epitomizes a model's resilience and flexibility, grounded on its capacity to generalize across diverse data distributions, explain its predictions transparently, and navigate the intricacies of edge cases effectively. The challenges associated with robust generalization are multifaceted, encompassing the model's performance on unseen data and its defense against out-of-distribution data and adversarial attacks. Bridging this gap, the potential of Embedding Propagation (EP) for improving out-of-distribution generalization is explored. EP is depicted as a powerful tool facilitating manifold smoothing, which in turn fortifies the model's robustness against adversarial onslaughts and bolsters performance in few-shot and self-/semi-supervised learning scenarios. In the labyrinth of deep learning models, the path to robustness often intersects with explainability. As model complexity increases, so does the urgency to decipher their decision-making processes. Acknowledging this, the thesis introduces a robust framework for evaluating and comparing various counterfactual explanation methods, echoing the imperative of explanation quality over quantity and spotlighting the intricacies of diversifying explanations. Simultaneously, the deep learning landscape is fraught with edge cases - anomalies in the form of small objects or rare instances in object detection tasks that defy the norm. Confronting this, the thesis presents an extension of the DETR (DEtection TRansformer) model to enhance small object detection. The devised DETR-FP, embedding the Feature Pyramid technique, demonstrating improvement in small objects detection accuracy, albeit facing challenges like high computational costs. With emergence of foundation models in mind, the thesis unveils EarthView, the largest scale remote sensing dataset to date, built for the self-supervised learning of a robust foundational model for remote sensing. Collectively, these studies contribute to the grand narrative of robustness in deep learning, weaving together the strands of generalization, explainability, and edge case performance. Through these methodological advancements and novel datasets, the thesis calls for continued exploration, innovation, and refinement to fortify the bastion of robust computer vision.

Key words: *computer-vision, self-supervised learning, explainability, adversarial attacks*

Resumen

Esta tesis se embarca en un viaje exploratorio hacia la robustez en el aprendizaje profundo, con un enfoque agudo en las facetas entrelazadas de la generalización, la explicabilidad y los casos extremos dentro del ámbito de la visión por computadora. En el aprendizaje profundo, la robustez epitomiza la resistencia y la flexibilidad de un modelo, basada en su capacidad para generalizar a través de diversas distribuciones de datos, explicar sus predicciones de manera transparente y navegar eficazmente por las complejidades de los casos extremos. Los desafíos asociados con la generalización robusta son multifacéticos, e incluyen el rendimiento del modelo en datos no vistos y su defensa contra datos fuera de distribución y ataques adversarios. Para salvar esta brecha, se explora el potencial de la Propagación de Incrustación (EP) para mejorar la generalización fuera de distribución, que facilita el suavizado del conjunto, y a la vez fortalece la robustez del modelo contra los embates adversarios, mejorando el rendimiento en aprendizaje con pocos ejemplos y auto/supervisado. En el laberinto de los modelos de aprendizaje profundo, el camino hacia la robustez a menudo se cruza con la explicabilidad. Reconociendo esto, la tesis introduce un marco robusto para evaluar y comparar varios métodos de explicación contrafactual, haciendo eco de la imperatividad de la calidad de la explicación sobre la cantidad y destacando las complejidades de diversificar las explicaciones. Simultáneamente, el panorama del aprendizaje profundo está lleno de casos extremos, anomalías en forma de objetos pequeños o instancias raras en tareas de detección que desafían la norma. Así, la tesis presenta una extensión del modelo DETection TRansformers para mejorar la detección de objetos pequeños que incorpora la técnica de la Pirámide de Características, aunque enfrenta desafíos como altos costos computacionales. Con la aparición de los modelos fundacionales en mente, la tesis desvela EarthView, el conjunto de datos de detección remota a mayor escala hasta la fecha, construido para el aprendizaje auto-supervisado de un modelo fundacional robusto para la detección remota. En conjunto, estos estudios contribuyen a la gran narrativa de la robustez en el aprendizaje profundo, entrelazando las hebras de la generalización, la explicabilidad y el rendimiento en los casos extremos. A través de estos avances metodológicos y conjuntos de datos novedosos, la tesis pide una continua exploración, innovación y refinamiento para fortificar el bastión de la visión por computador robusta.

Palabras clave: *visión por computador, aprendizaje auto-supervisado, explicabilidad, ataques adversarios*

Resum

Aquesta tesi s'embarca en un viatge exploratori cap a la robustesa en l'aprenentatge profund, amb un enfocament agut en les facetes entrelaçades de la generalització, l'explicabilitat i els casos límit dins de l'àmbit de la visió per ordinador. En l'aprenentatge profund, la robustesa epitomiza la resistència i la flexibilitat d'un model, basada en la seva capacitat per generalitzar a través de diverses distribucions de dades, explicar les seves prediccions de manera transparent i navegar eficaçment per les complexitats dels casos límit. Els desafiaments associats amb la generalització robusta són multifacètics, incloent-hi el rendiment del model en dades no vistes i la seva defensa contra dades fora de distribució i atacs adversaris. Per salvar aquesta bretxa, s'explora el potencial de la Propagació d'Incrustació (EP) per millorar la generalització fora de distribució, que al seu torn enforteix la robustesa del model contra els embats adversaris i millora el rendiment en aprenentatge amb pocs exemples i auto/supervisat. Dins el laberint dels models d'aprenentatge profund, el camí cap a la robustesa sovint es creua amb l'explicabilitat. A mesura que augmenta la complexitat del model, també augmenta la urgència de desxifrar els seus processos de presa de decisions. Reconèixer això, la tesi introdueix un marc robust per avaluar i comparar diversos mètodes d'explicació contrafactual, fent ressò de la imperativitat de la qualitat de l'explicació sobre la quantitat i destacant les complexitats de diversificar les explicacions. Simultàniament, el panorama de l'aprenentatge profund està ple de casos límit, anomalies en forma de petits objectes o instàncies rares en tasques de detecció d'objectes que desafien la norma. Enfrontant això, la tesi presenta una extensió del model DETection TRansformer per millorar la detecció de petits objectes que incorpora la tècnica de la Piràmide de Característiques, tot i que es troba desafiaments com ara alts costos computacionals. Amb l'aparició dels models fonamentals en ment, la tesi desvetlla EarthView, el conjunt de dades de detecció remota a major escala fins ara, construït per a l'aprenentatge auto-supervisat d'un model fonamental robust per a la detecció remota. Col·lectivament, aquests estudis contribueixen a la gran narrativa de la robustesa en l'aprenentatge profund, entrelaçant les fils de la generalització, l'explicabilitat i el rendiment en els casos límit. A través d'aquests avanços metodològics i conjunts de dades novells, la tesi fa una crida a la continuada exploració, innovació i refinament per enfortir el bastió de la visió per computador robusta.

Paraules clau: *visió per computador, aprenentatge auto-supervisat, explicabilitat, atacs adversaris*

Contents

Abstract (English/Spanish/Catalan)	iii
List of figures	xiii
List of tables	xvii
1 Introduction	1
1.1 Goals and Contributions	3
1.1.1 Improving Small Object Detection	3
1.1.2 Enhancing Out-of-Distribution Performance	4
1.1.3 Evaluating Counterfactual Explanations	5
1.1.4 Foundation Model for Remote Sensing	5
1.2 Structure of the dissertation	6
2 Towards small logo detection with no priors	9
2.1 Related Work	9
2.1.1 Object Detection with Priors	9
2.1.2 Set Prediction with Priors	11
2.1.3 Detection Transformers	11

Contents

2.1.4	Towards Small Objects: Logo Detection	12
2.2	Methodology	13
2.2.1	Prediction of sets	13
2.2.2	The core DETR architecture	14
2.2.3	Feature Pyramid Networks + DETR	15
2.3	Experiments and Results	16
2.3.1	DETR vs. Faster-RCNN	18
2.3.2	Hyperparameter search	20
2.3.3	Decomposing the performance	23
2.3.4	DETR-FP	24
2.4	Discussion	26
3	Embedding Propagation for Manifold Smoothing	29
3.1	Related Work	29
3.2	Methodology	32
3.3	Experiments and Results	35
3.3.1	Datasets	36
3.3.2	Manifold smoothness	36
3.3.3	Adversarial robustness	39
3.3.4	Self-supervised Learning	41
3.3.5	Few-Shot and Semi-supervised Learning	44
3.4	Discussion	47

4	A principled Benchmark for Visual Counterfactual Explainers	49
4.1	Related Work	49
4.2	Methodology	51
4.2.1	Data generation	51
4.2.2	Counterfactual generation	52
4.2.3	Optimal Classifier	53
4.2.4	Evaluating Counterfactual Explanations	54
4.3	Experiments and Results	57
4.3.1	Datasets	58
4.3.2	Metric Evaluation	59
4.3.3	Limitations	63
4.4	Discussion	65
5	EarthView: A Large Scale Remote Sensing Dataset	67
5.1	Related Work	67
5.1.1	Dataset for training	67
5.1.2	Learning from unlabelled data	69
5.2	Proposed Dataset	70
5.2.1	Sentinel	70
5.2.2	Satellogic	72
5.2.3	NEON	72
5.2.4	Hosting and Storage	74

Contents

5.3	Proposed Model	74
5.3.1	EarthMAE	75
5.3.2	Training Paradigm	76
5.4	Experiments and Results	76
5.5	Discussion	79
6	Conclusions and Future work	81
6.1	Conclusions	81
6.2	Future Perspective	82
6.3	Scientific Articles	83
6.3.1	Journals	84
6.4	Contributed Code and Datasets	84
6.4.1	In the Media	85
	Bibliography	112

List of Figures

2.1	The DETR-FP approach	16
2.2	DETR attention on MS-COCO	20
2.3	DETR attention on OpenLogo	21
2.4	Random search parameters	22
2.5	Parameter importance	22
2.6	Performance decomposition	24
2.7	DETR-FP attention on OpenLogo	25
2.8	DETR-FP qualitative results	26
3.1	Embedding propagation method	33
3.2	Overview of the EPNet training procedure across different tasks	34
3.3	Comparison of the class embedding manifold	37
3.4	Interpolation of embedding pairs	38
3.5	Alignment and Uniformity	41
3.6	Performance across query sizes	46
4.1	Causal graph	53
4.2	Generator interpolations	59

List of Figures

4.3	Average attribute perturbation	59
4.4	Explainer sensitivity	62
4.5	Causal Flip counterfactuals	64
5.1	Masking schemas	69
5.2	Samples from the dataset	70
5.3	Spatial Coverage	73
5.4	Temporal distribution	73
5.5	EarthMae architecture	75
5.6	Masking and time ablation	77
5.7	Different dataset size performance	78

List of Tables

2.1	Object detection baselines	19
2.2	Models weights ablation	19
2.3	Quantitative results on OpenLogo	24
3.1	Adversarial attacks results	39
3.2	Self-supervised results	41
3.3	Few-show classification results	43
3.4	Semi-supervised learning results	46
4.1	Explainers comparison	51
4.2	Score and trivial counterfactuals	60
4.3	Percentage of Estimator Flips	61
5.1	Dataset overview	74

1 Introduction

The domain of computer vision has seen significant advances in recent years, but these developments have also unearthed numerous challenges related to the robustness and generalizability of these models. In this dissertation, we investigate four key areas of computer vision, each presenting unique obstacles and opportunities, with a central theme of enhancing model robustness.

In the context of deep learning, robustness refers to the ability of a model to maintain its performance when faced with various forms of perturbations. These perturbations could be in the form of input noise, adversarial attacks, or changes in the data distribution. A robust model, therefore, is one that can effectively generalize from its training data to unseen data, maintain its performance under different operating conditions, and resist manipulation by adversarial inputs.

Improving edge cases, such as the detection of small objects in images, contributes to robustness by enhancing the model's ability to handle a wider range of scenarios. In real-world applications, objects of interest can vary significantly in size and may often be small relative to the image size. By improving the model's performance on these edge cases, we ensure that the model's performance is not overly dependent on the size of the objects, thereby enhancing its robustness.

Out-of-distribution performance is another critical aspect of robustness. In real-world applications, the data that a model encounters may not always follow the same distribution as the training data. Improving a model's out-of-distribution performance ensures that it can maintain its performance even when faced with such data, thereby enhancing its robustness.

Explainability in deep learning models is a crucial factor in robustness. While deep learning models are often seen as "black boxes" due to their complex, layered architectures, efforts to improve their explainability can lead to more robust models. By understanding how a model makes its decisions, we can identify potential weaknesses or biases in the model and address them, thereby enhancing its robustness.

Finally, the creation of foundation models can also contribute to robustness. Foundation models are large-scale models trained on diverse data, intended to serve as a starting point for more specific models. By starting with a foundation model, we can leverage the broad generalization capabilities that these models

have learned. This can help in creating more robust models, as the foundation model has already learned to handle a wide range of data variations.

In this dissertation, we delve into each of these areas, exploring the challenges and opportunities they present, and proposing novel solutions to enhance the robustness of computer vision models. Through our investigations, we aim to push the boundaries of what is currently achievable in computer vision, paving the way for more reliable and robust applications in the future. Small object detection, an integral part of computer vision, is a critical tool for several applications, ranging from surveillance to autonomous vehicles. In recent years the field has evolved rapidly, marked by the proliferation of models such as R-CNN [53], YOLO [151, 152, 153], CenterNet [233], and many others. These models exhibit impressive performance on various benchmarks; however, they are typically grounded in the same principle—addressing object detection as a supervised classification problem on proposed regions. While this approach has proved successful, it also carries inherent priors that can jeopardize the robustness of the resulting models.

The embedding propagation (EP) procedure has been shown to improve few-shot learning performance. The main hypothesis is that EP smooths the class embedding manifold, acting as a regularizer. In fact, smooth class embedding manifolds are a known requisite for semi-supervised learning [27], and to improve adversarial robustness [201]. In our second contribution, we provided additional quantitative and qualitative insights showing that EP yields smoother classification surface as measured with the Laplacian. In addition, we extended [164] showing that besides few-shot classification, EP also improves adversarial robustness and self/semi-supervised learning performance.

DiVE [161] and DiCE [126] propose metrics that allow researchers to evaluate the quality of an explanation. These metrics evaluate the proximity of explanations to their original sample, and how diverse these are. Unfortunately, they are easy to game. For example, an explainer could maximize diversity by always modifying the same counterfactual attribute but randomly perturbing other non-counterfactual attributes to produce new redundant explanations. We propose a more general, harder to game metric that allows us to evaluate a set of explainers in order to identify their strengths and weaknesses through fair comparisons. Further, the set of attributes of a dataset can influence the evaluation of the explainability methods. CelebA [115] is a common dataset used for generating counterfactual explanations [38, 161], and it is labeled with a series of attributes, such as "Attractive", that fail to fully describe the true underlying factors that generated the images (e.g, illumination, occlusions, contrast, etc). Likewise, there is no guarantee that unsupervised disentanglement methods such as VAEs identify the true factors of variations without

making strong assumptions [2]. We sidestep these problems by evaluating all explainers in a common latent space with known attributes that fully describe the samples. Recently [142] published a benchmark (CARLA) with an extensive comparison of several counterfactual explanation methods across 3 different tabular datasets.

Finally, as the urgency to address climate change, environmental predicaments, and natural disasters intensifies, the pivotal role of Earth monitoring has gained increased recognition [165]. Moreover, this vital tool is progressively influential in arenas like agriculture [86] and city planning [127]. Traditional vision models have already rendered substantial contributions across numerous applications [98, 148, 162]. With the advent of foundation models [95, 133, 134], we foresee an evolution in the capabilities of Earth monitoring, with these models serving as a robust bedrock for future developments.

1.1 Goals and Contributions

1.1.1 Improving Small Object Detection

Despite recent advances, these models often struggle with the detection of small objects, which can drastically limit their usability in real-world scenarios. Our first paper focuses on this issue by extending the DETR model using a Feature Pyramid Network technique to enhance the detection of small objects. These models exhibit impressive performance on various benchmarks; however, they are typically grounded in the same principle—addressing object detection as a supervised classification problem on proposed regions. While this approach has proved successful, it also carries inherent limitations that can jeopardize the robustness of the resulting models. These models contain numerous hyperparameters, require multiple post-processing steps, and are often complex to handle. This complexity can be attributed to the treatment of object detection as a box-classification problem, which brings about several challenges that have been solved by introducing prior geometrical knowledge to be fed into the model. Whether it be anchor boxes, their ratio and size, or a grid of possible object centers [191, 233]. This hand-crafted, prior information has a severe impact on model performance as shown in [231]. To remove this priors entirely DETR [22] proposes to treat the object detection problem as a set prediction problem approach to object detection, treating it as a set prediction problem directly and thereby bypassing the surrogate tasks intrinsic to traditional models. Adopting an encoder-decoder architecture based on transformers [199], DETR enables the incorporation of global context in its

predictions.

Despite its advantages, DETR struggles with detecting small objects due to the lack of low-level features fed into the transformer. To bolster the model’s robustness with small object detection, we introduce a Feature Pyramid (FP) Network to the DETR model. This enhanced model, termed DETR-FP, captures low-level feature maps and feeds them into the model at higher resolution.

1.1.2 Enhancing Out-of-Distribution Performance

Another critical aspect of robustness in computer vision relates to the model’s out-of-distribution (OOD) performance. Adversarial attacks, few-shot learning scenarios, and self-/semi-supervised learning tasks pose significant challenges for conventional machine learning models. In the second contribution, a thorough investigation of the utilization of Embedding Propagation (EP) for manifold smoothing, a non-parametric method that improves out-of-distribution (OOD) performance in machine learning tasks. The overarching goal is to cultivate a higher level of model robustness against various disruptive factors, particularly in the contexts of few-shot learning classification, adversarial attacks, and self-/semi-supervised learning scenarios. EP constructs interpolations from the output features of a neural network based on their similarity in a constructed graph. In turn, this procedure regularizes the manifold for both training and testing, effectively creating smoother class embedding manifolds and strengthening the robustness of the model against noise. Chapter 3 explores the remarkable benefits of employing Ensemble Propagation (EP) in machine learning algorithms. EP contributes to an increased smoothness of the classification surface, hence fortifying defenses against adversarial attacks that leverage the model’s susceptibilities towards erroneous high-confidence predictions on perturbed inputs. Further, EP extends its utility to self- and semi-supervised learning algorithms, serving as an effective hard negative mining method, thereby heightening their performance. Our focus is on the particular impact of EP in self-supervised learning scenarios with unlabeled samples, where our results evidence a substantial boost in accuracy and a marked decline in performance when EP is eliminated. The research encompasses various datasets such as miniImagenet, tieredImagenet, CIFAR10, CIFAR100, MNIST, Fashion-MNIST, and STL-10. In these diverse contexts, our experiments consistently validate that EP not only bolsters performance in few-shot learning scenarios but also significantly reinforces model resilience against iterative perturbations.

1.1.3 Evaluating Counterfactual Explanations

The interpretability of deep learning models has become increasingly important as they become more complex and ubiquitous. Counterfactual explanations offer a promising avenue for improving explainability, but their evaluation lacks robustness. Given the complex nature of deep learning, the researchers emphasize the need for explainability and the insights that can be derived from counterfactual explanations. Our third paper introduces a comprehensive framework for evaluating counterfactual explanation methods, addressing current gaps in evaluation metrics and methodologies, thereby fostering a more principled approach to explanation. Acknowledging the issues present in current approaches—like the lack of adequate metrics and poor dataset comparisons—the we propose a comprehensive solution. The proposed framework relies on principled evaluation methods and the use of an annotated synthetic dataset. The dataset is generated using Structural Causal Models (SCM) and ‘Synbols’ [94], a tool for generating images with fully annotated attributes. For counterfactual explanation generation, the perturbation process is described in the learned latent space leading to classifier sensitivity. The paper goes on to demonstrate the application of this framework with various methods such as Latent-CF [5], DiCE [126], xGEM [82], DiVE [161], GS [99], StyleEx [97], on the Synbols dataset. The research findings underscore the complexity of diversifying explanations and question the value of quantity over quality in counterfactual explanations.

1.1.4 Foundation Model for Remote Sensing

Despite the data abundance, a significant proportion is inaccessible due to prohibitive paywalls. Even though free data sources such as Sentinel-1 and Sentinel-2 are available, they come with their own set of challenges, including a **low spatial resolution** with a 10m ground sample distance (GSD), and **download difficulties** due to bandwidth limitations on Google Earth Engine and the associated expenses with AWS for large-scale downloads. In order to alleviate these issues and contribute towards the robustness of Earth monitoring, we have collaborated with Satellogic and NEON to release a substantial 22 tera pixel dataset, specifically curated for the large-scale self-supervised learning of foundational models in Earth monitoring. Accessible via Hugging Face, this dataset is efficiently divided into subsets for easier processing. This robust dataset encompasses structured data gathered from three distinct sources:

- Satellogic: Supplies RGB and near-infrared data at a 1m GSD, complemented with temporal revisits and planet coverage.

- NEON: Contributes 369 bands of hyperspectral data at a 1m GSD, augmented by RGB data at a 0.1m GSD and elevation data at a 1m GSD from multiple US forests.
- Sentinel: We have compiled a substantial structured subset of Sentinel-1 and 2, amalgamating multi-spectral, synthetic aperture radar (SAR), and temporality.

This work is fundamentally underpinned by two key contributions:

- The provision of a large-scale, multi-modal dataset specifically crafted for robust, self-supervised learning of foundation models in Earth monitoring.
- The creation of a comprehensive masked auto-encoder, trained through various self-supervision schemas, demonstrating an enhanced performance in diverse Earth monitoring tasks.

The robustness of this initiative is amplified by the extensive and diverse data sources, meticulously designed to enhance the pre-training of a foundational model for Earth monitoring. This large-scale self-supervised training aims to fortify the generalization capabilities of these models,

thereby overcoming common limitations and augmenting their reliability and effectiveness in real-world applications.

1.2 Structure of the dissertation

Through this body of work, we aim to explore and address some key challenges in the pursuit of robustness in computer vision, and in doing so, offer new perspectives and approaches for tackling this multifaceted problem.

- In our first contribution, we extend DETR, a new fully differentiable end-to-end solution for object detection that requires no geometric priors and no post processing, for small object detection. We improved its performance by feeding multi level information using a Feature Pyramid (FP) and compared its results with a strong Faster-RCNN baseline in the MS-COCO and OpenLogo benchmarks where we obtain up to a 30% relative improvement. There is, however much room for improvement in the DETR-FP approach. It's considerable computational cost prevents us from using the lower levels of the feature pyramid. Furthermore the introduction of different feature maps for the object queries to analyse, increases the amount of duplicate detections, since the same object can

be detected in different levels of the pyramid. The direction of our future work moves towards addressing these issues in order to make DETR-FP performance a total improvement over the strong baselines that make use of several geometric priors.

Publication: **Velazquez, D**, Gonfaus, J. M., Rodriguez, P., Roca, F. X., Ozawa, S., and Gonzalez, J. (2021). Logo Detection With No Priors. IEEE Access, 9, 106998-107011

- In chapter 3 we demonstrate that the use of EP increases the smoothness of the classification surface, thereby enhancing robustness against adversarial attacks. These attacks frequently exploit the model’s tendency to make high-confidence incorrect predictions when presented with perturbed inputs. Beyond adversarial attacks, the paper also establishes how EP can further enhance robustness within self- and semi-supervised learning algorithms. It functions as a natural hard negative mining method, improving the performance of such learning models. Particularly in self-supervised learning scenarios with unlabeled samples, our results indicate a significant increase in accuracy and a notable detriment in performance upon EP removal. The research is conducted using various datasets, including miniImagenet, tieredImagenet, CIFAR10, CIFAR100, MNIST, Fashion-MNIST, and STL-10. Our experiments consistently show that EP not only improves the performance in few-shot learning scenarios, but also significantly enhances the model’s robustness against iterative perturbations.

Publication: **Velazquez, D**, Rodríguez, P., Gonfaus, J. M., Roca, F. X., and González, J. (2022). A closer look at embedding propagation for manifold smoothing. The Journal of Machine Learning Research, 23(1), 11447-11473.

- Chapter 4 introduces a benchmark serving as a comprehensive platform for assessing a variety of counterfactual explanation methods. A distinguishing feature of this benchmark is its synthetic image base, where every image is fully characterized by annotated attributes. These attributes are made accessible to the explainers through a differentiable generator. This unique structure allows for a more nuanced and fair evaluation of different counterfactual explanation methods, particularly as it relates to their effectiveness and utility in practical applications. Our findings underscore the limitations of relying solely on an increase in

the number of counterfactuals to boost performance. Instead, the quality of explanations and the ability of explainers to deliver non-trivial and diverse explanations emerge as critical factors for successful outcomes. It also emphasizes the importance of a fair evaluation setup and highlights potential biases introduced by an over-reliance on optimal classifiers. The proposed benchmark and the accompanying findings are anticipated to serve as a catalyst for future research in this area. The focus is not just on creating more advanced counterfactual explanation methods, but also on building fairer and more rigorous evaluation mechanisms. The goal is to pave the way for the development of more effective, reliable, and interpretable deep learning models, enhancing their potential to contribute meaningfully to a variety of real-world applications.

Publication: **Velazquez, D**, Rodriguez, P., Lacoste, A., Laradji, I. H., Roca, X., and González, J. (2023). Explaining Visual Counterfactual Explainers. Transactions on Machine Learning Research.

- Earth Monitoring

Lastly, chapter 4 5 presents our ongoing work to provide a foundation model for earth monitoring. We introduce a unique, expansive remote sensing dataset, comprising over 22 trillion pixels - the most extensive of its kind. This multi-sensor, multi-data type dataset will significantly advance remote sensing research. While we present EarthMAE, a model tailored for our dataset, the dataset itself holds the greatest potential. Its diversity allows examination of varied sensor types and data structures, while enabling efficient self-supervised learning scenarios across multiple GPUs. Our study explores different masking strategies and uses timestamp metadata to enhance model accuracy across tasks. The value of our dataset, however, reaches beyond our work. It opens opportunities for future self-supervised learning and remote sensing applications. We expect our efforts to catalyze further research, driving innovative solutions to complex challenges.

Publication: Ongoing.

2 Towards small logo detection with no priors

In recent years, top referred methods on object detection like R-CNN have implemented this task as a combination of proposal region generation and supervised classification on the proposed bounding boxes. Although this pipeline has achieved state-of-the-art results in multiple datasets, it has inherent limitations that make object detection a very complex and inefficient task in computational terms. Instead of considering this standard strategy, in this paper we enhance Detection Transformers (DETR) which tackles object detection as a set-prediction problem directly in an end-to-end fully differentiable pipeline without requiring priors. In particular, we incorporate Feature Pyramids (FP) to the DETR architecture and demonstrate the effectiveness of the resulting DETR-FP approach on improving logo detection results thanks to the improved detection of small logos. So, without requiring any domain specific prior to be fed to the model, DETR-FP obtains competitive results on the OpenLogo and MS-COCO datasets offering a relative improvement of up to 30%, when compared to a Faster R-CNN baseline which strongly depends on hand-designed priors.

2.1 Related Work

The work presented in this paper proposes a pure end-to-end solution to object detection using transformers [22] expanding on previous work by incorporating a Feature Pyramid (FP) network to the DETR architecture and benefiting from bipartite matching losses for set prediction, encoder-decoder architectures based on the transformer, parallel decoding, and other contributions from relevant object detection methods as described next.

2.1.1 Object Detection with Priors

Standard object detection methods use deep learning to generate regions proposals and subsequently classify them, although unifying these two tasks would be more efficient.

The first deep learning method approaching region proposal and classification was R-CNN[53], which uses a selective search to generate the object

proposals and a CNN on top to extract relevant features to be later classified by an SVM. Improvements of this architecture lead to the appearance of Fast-RCNN[54], where region proposals were extracted from features map of a CNN by applying the ROI Pooling operation and then classify each region with a fully connected network. Later, Faster-RCNN[157] solved the main draw-back of previous architectures by substituting the selective search by a dedicated CNN called Region Proposal Network, which learned, given a predefined set of anchor boxes, where objects are located in an image and their shape. Finally, Mask-RCNN [65], replaced the ROI Pooling with the ROI Align operation which removes the quantization present in the former, improving both segmentation and detection results. This family of RCNN detectors has been incrementally improved with recent works such as [209, 223] which guide the region anchors by learning where objects are likely to exist in an image. [20] improves the Faster-RCNN detector using multi-scale convolution feature fusion to make the feature map contain more information, improving the detection of small objects.

All of the aforementioned methods are considered two-stage detectors, the first stage being the region proposal stage and the second one the classification and box refinement stage. One-stage detectors merge these two stages into one by directly predicting predicting class probabilities and box position on image grid cells, given predefined anchors.

Notice that whether the method is one or two stage, it requires prior geometrical knowledge to be fed into the model. Whether it be anchor boxes, their ratio and size, or a grid of possible object centers [191, 233]. This hand-crafted, prior information has a severe impact on model performance as shown in [231].

The closest works to the DETR approach but using priors are end-to-end set predictions for object detections [181] and instance segmentation [138, 156, 168, 174]. They also use bipartite-matching losses with decode-encoder architectures, however they are also based on autoregressive models (RNNs).

Previous work using bipartite matching loss [45, 113, 151] modeled the relation between different predictions using convolutional or fully connected layers with a hand-designed NMS as a post processing step to improve their performance. Some other, more recent work [109, 157, 233] use non-unique assignment rules between ground truth and predictions together with an NMS.

Learnable NMS methods [16, 69] and relation networks [71] use attention to model the relationship between predictions, by using direct set losses they do not require any post-processing steps. They do, however, require for hand-crafted prior information to be fed into the model (e.g., box proposals, anchors). DETR,

in contrast, removes the need for these geometric priors or post processing steps by treating object detection as a set prediction problem.

2.1.2 Set Prediction with Priors

There is no canonical deep learning model to directly predict sets (of detected objects). The basic set prediction task is multi-label classification some examples of this for computer vision can be found in [158], [159]. However the one-vs-rest approach proposed in the aforementioned papers is not suitable for object detection due to the existing underlying structure between elements (i.e near identical boxes), which gives rise to near-duplicate detections.

Most current detectors use geometric prior knowledge to apply some sort of post-processing step, usually non maximum suppression to remove these near-duplicates, however set prediction is post-processing free. For constant size set prediction one could use dense fully connected networks [45], but given that the number of objects varies across images the problem cannot be approached in such a way. A general approach, suitable for variable length sets, is to use auto-regressive sequence models such as recurrent neural networks [205]. Whenever making set predictions the loss should be invariant to permutation in the predictions. A good solution is to design a loss based on the Hungarian algorithm [91] to match predictions and ground truth, in a one-to-one fashion, ensuring permutation-invariance.

2.1.3 Detection Transformers

Transformers were introduced by Vaswani et al [199] as a new attention-based building block for machine translation. Attention mechanisms [4] are neural network layers that aggregate information from the entire input sequence. Transformers implemented layers of self-attention which search and update every item of a sequence by aggregating information from the entire sequence. One of the major advantages of attention-based models is their global computations and perfect memory, making them more suitable on long sequences than RNNs. In many problems transformers now replace RNNs in natural language processing, speech processing and computer vision [39, 120, 139, 149, 188].

Transformers were first used in auto-regressive models, following early sequence-to-sequence models [187], generating output tokens one by one. Due to the prohibitive inference cost (proportional to output length, and hard to batch) development of parallel sequence generation have been proposed in the domains of audio [132], machine translation [50, 59], word representation learning , and more recently speech recognition [25]. This work also combine transformers

and parallel decoding for their suitable trade-off between computational cost and the ability to perform global computations.

Based on the encoder-decoder architecture of the transformers, Detection Transformers [22] were recently proposed as a simpler, fully differentiable end-to-end method, which requires no priors and no post-processing. In essence, for DETRs the object detection problem is dealt as a set prediction problem directly. Since the basis of our approach is based on DETR, we will describe next its main properties when applied to object detection.

DETR predicts all objects at once, and is trained in an end-to-end fashion, with a set loss function which performs bipartite matching between predicted and ground-truth objects. DETR does not require any hand-designed components that encode geometrical priors, such as non maximum suppression or spatial anchors. In addition, it does not require any custom layers (ROIAlign, ROIPool), thus making it reproducible in any deep learning framework, without the need to write custom GPU/CPU kernels for these custom layers to maximize performance. Finally, previous work on set-prediction was focused on autoregressive decoding with RNNs [138, 156, 168, 174, 181]. In contrast DETR matching function is permutation invariant, so is the transformer architecture in itself, thus allowing (non-autoregressive) parallel decoding [39, 50, 59, 131] of the predictions.

2.1.4 Towards Small Objects: Logo Detection

The problem of logo recognition itself has a rich history of research. It is a challenging problem since logos are usually small and tiny differences in text or shape can represent widely different logos. In the 1990s, the question was explored primarily in the field of information retrieval use-cases. An image descriptor was developed with affine Transformations and stored in the Retrieval database [40]. There were also several methods focused on the neural networks [23, 48] but neither the networks were as deep nor the results were as impressive as in recent work.

In the 2000's, improved image descriptors became feasible with the introduction of SIFT and similar methods [8, 119, 169]. These methodologies were used to better represent images for recognizing logos [17, 80, 146, 167, 235]. Apart from SIFT, Other approaches have also been explored by the community, metric learning [28], [128], using min-hashing [166] and bundling features for improved search [217]. Most of those approaches required complex pipelines for preprocessing images.

Recent work in logo recognition utilizes deep neural networks that offer superior performance with end-to-end automation of the pipeline. Broadly

speaking, the following approach is prevalent: an image is fed into a convolutional neural network and a classifier predicts [14, 15, 68, 74, 130, 183, 195]. All the aforementioned works require the use of prior information.

To the best of our knowledge, Detection Transformers had not been applied yet to logo detection due to their problems in detecting small objects. In order to solve this, we next describe the proposed method in order to extend the DETR architecture for logo detection without the use of geometrical priors, such as non maximum suppression or spatial anchors.

2.2 Methodology

In this work we improve the DETR architecture for small objects with the use of a Feature Pyramid Network, thus addressing one of the main drawbacks of the DETR architecture. In this section we will describe the basic DETR architecture, the criterion used for training, and the improvements we made in order to improve the performance for small objects.

2.2.1 Prediction of sets

One of the main challenges addressed in DETR is to treat object detection as a set-prediction problem in a non-autoregressive manner. In order to do this DETR predicts a pre-defined fixed-size of N predictions per image and pads the ground truth labels with as many \emptyset (no object) as necessary to reach N labels.

Let us denote \hat{y} as the set of prediction and y as the set of padded ground truth. Then to find a bipartite matching between these two sets we need to search for a permutation of N elements with the lowest cost:

$$\hat{\sigma} = \arg \min_{\sigma \in N} \sum_i^N \mathcal{L}_{match}(y_i, \hat{y}_{\sigma(i)}) \quad (2.1)$$

where $\mathcal{L}_{match}(y_i, \hat{y}_{\sigma(i)})$ is a pair-wise matching cost between ground truth y_i and a prediction with index $\sigma(i)$. This function minima is computed efficiently finding an optimal matching using the Hungarian algorithm [91]. The matching also takes into account the similarity of the predicted and ground truth boxes. We can view \hat{y} as a vector that contains a class label c and a bounding box b . For an index prediction $\sigma(i)$ the probability class is defined as $\hat{p}_{\sigma(i)}(c_i)$ and the predicted box as $\hat{b}_{\sigma(i)}$. Thus $\forall c_i \notin \emptyset$ $\mathcal{L}_{match}(y_i, \hat{y}_{\sigma(i)})$ can be defined as:

$$\mathcal{L}_{match}(y_i, \hat{y}_{\sigma(i)}) = \hat{p}_{\sigma(i)}(c_i) + \mathcal{L}_{box}(b_i, \hat{b}_{\sigma(i)}) \quad (2.2)$$

After the optimal matching is computed, the next step is to compute the actual loss for all pairs matched in Eq. (2.1). The loss is defined as a linear combination of negative log-likelihood for class prediction and a box loss:

$$\mathcal{L}_{Hungarian}(y, \hat{y}) = \sum_{i=1}^N -\log \hat{p}_{\hat{\sigma}(i)}(c_i) + \mathcal{L}_{box}(b_i, \hat{b}_{\sigma(i)}) \quad (2.3)$$

where $\hat{\sigma}$ is the optimal assignment computed in Eq. (2.1). When $c_i \in \emptyset$ The log probability term is down weighted by a factor of 10 to alleviate class imbalance.

The second part of the matching cost and the Hungarian loss is the box loss. These boxes are predicted directly in contrast with many other detectors that make the predictions with respect to some initial guesses. This, however, gives rise to the problem of the relative scaling of the loss, since the ℓ_1 loss will have different scales for small and large boxes, even if their relative error is similar. In DETR this problem is mitigated by linearly combining the aforementioned ℓ_1 loss and the generalized IoU loss [158] that is scale invariant. Thus the box loss $\mathcal{L}_{box}(b_i, \hat{b}_{\sigma(i)})$ is defined as:

$$\mathcal{L}_{box}(b_i, \hat{b}_{\sigma(i)}) = \lambda_{iou} \mathcal{L}_{iou}(b_i, \hat{b}_{\sigma(i)}) + \lambda_{L_1} \|b_i - \hat{b}_{\sigma(i)}\|_1 \quad (2.4)$$

where λ_{iou} and λ_{L_1} are hyperparameters that control the weight of each term in the loss, and $\mathcal{L}_{iou}(b_i, \hat{b}_{\sigma(i)})$ is the generalized IoU loss defined as:

$$\mathcal{L}_{iou}(b_i, \hat{b}_{\sigma(i)}) = 1 - \left(\frac{|b_{\sigma(i)} \cap \hat{b}_i|}{|b_{\sigma(i)} \cup \hat{b}_i|} - \frac{|B(b_{\sigma(i)}, b_i) \setminus b_{\sigma(i)} \cup \hat{b}_i|}{|B(b_{\sigma(i)}, \hat{b}_i)|} \right) \quad (2.5)$$

where $|\cdot|$ means "area", $B(b_{\sigma(i)}, b_i)$ means the largest box containing $b_{\sigma(i)}, b_i$. The areas are computer based on min / max of linear functions of the box coordinates and the union and intersection of box coordinates are used as shorthands for the boxes themselves.

2.2.2 The core DETR architecture

The DETR architecture is rather simple, therefore, unlike many modern detectors, it can be implemented in any deep learning framework in just a few hundred lines, without the need for huge configuration files such as those seen in Detectron2 [216] and without the need to implement custom layers.

The DETR architecture is mainly composed of 3 components. A conventional CNN backbone that given the initial image and yields an activation

map $f \in \mathbb{R}^{C \times H \times W}$. This feature map is then fed to a 1×1 convolution in order to reduce the channel dimension from C to d creating a new feature map $z_0 \in \mathbb{R}^{d \times H \times W}$. Since the encoder expects a sequence the z_0 activation map spatial dimensions are collapsed into one, thus resulting in a $d \times HW$ feature map. The transformer architecture, however, is permutation invariant, so this feature map input must be added to positional embeddings [11, 139], which are basically sine waves at different frequencies. This is the same trick used in the original transformer paper [199] but adapted to images. Each encoder layer has a standard architecture and consists of a multi-head attention module and a feed forward network (FFN).

The decoder follows the standard architecture of the transformer. It uses self-attention and encoder-decoder attention mechanisms to transform N embeddings of size d . These embeddings are called object queries, which are learned positional embeddings that are tasked with *looking* at different regions in the feature map to find objects. These embeddings are transformed by the decoder and then decoded independently into a set of box coordinates and class labels by small feed forward network, resulting in N final predictions. Notice how, the DETR transformer is extremely similar to the standard transformer architecture.

This unique ability that the transformer has, making use of these self- and encoder-decoder attention mechanisms, allows the model to reason globally about all the objects, while being able to use the whole image as context.

2.2.3 Feature Pyramid Networks + DETR

As we briefly mentioned in the introduction one of the flaws of DETR is that it while excelling on localizing large objects, it struggles to find small ones. This is due to the lack of low level features that can be fed into the transformer. This problem is addressed by adding a feature pyramid (FP) network [108] that allows us to capture the content of low level feature maps and feeds it into the model in a higher resolution. However, using low level features in this architecture is very computationally expensive, so a design decision must be made, as described next.

Let us denote the feature map of the last level of an FPN as $f \in \mathbb{R}^{C \times H \times W}$. This feature map is then projected into d dimensions using a 1×1 convolution and flattened into $d \times HW$ in order to be fed into the transformer. Notice that the complexity of the self-attention of the encoder is $\mathcal{O}(d^2HW + d(HW)^2)$, while $\mathcal{O}(d(HW)^2)$ is the complexity of computing attention weights for only one head. Feeding lower level features to the model will increase this complexity quadratically, since a feature map from the second to last level of the FPN has

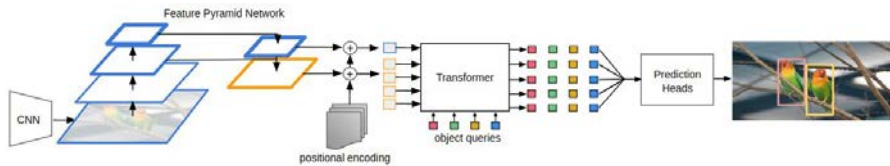


Figure 2.1: The DETR-FP approach to improve the localization of small objects. We introduce a feature pyramid and split the second to last level of the pyramid into four patches of the same size as the smallest feature map in the pyramid. This is all fed into the DETR pipeline and the predictions of each query for each feature map are concatenated.

a size of $C \times 2H \times 2W$.

In order to alleviate this cost, we take the approach shown in Figure 2.1. Where we crop the second to last level of the FPN into four equally sized $C \times H \times W$ patches and feed them all into the transformer. The object queries learn to *query* higher resolution feature maps for smaller objects. It is worth mentioning that this approach can be repeated for every level in the pyramid at the cost of a considerable memory increase due to the increase in size in the computational graph needed for backward propagation. In this work we only use the last two levels of the pyramid.

2.3 Experiments and Results

We compare the performance of DETR-FP against a strong Faster-RCNN baseline in the task of Logo detection, concretely in the QMUL-Openlogo benchmark [183]. The dataset is comprised of 27,083 images from 352 logo classes, built by aggregating and refining seven other logo datasets [14, 15, 81, 83, 107, 182, 196]. We use weights from ImageNet and MS-COCO [110] in different experiments for both architectures. The QMUL-OpenLogo dataset has three different settings in order to reproduce the aforementioned real world scenario situation, where new classes constantly have to be learned. There are three settings, but only the first one is fully supervised, where every logo classes contains 70% of training split and 30% of evaluation split. Note that this data split is enforced by the challenge organizers in order to compare each method in the benchmark fairly.

All the experiments are run on eight GeForce GTX 1080 TI GPUs, using Distributed Data Parallel from PyTorch [141].

The DETR-FP pipeline is extended from the original DETR work. All of the DETR-FP models are trained using AdamW [118] with improved weight decay handling, set to 10^{-4} , gradient clipping is applied with a maximal gradient norm of 0.1 in order to stabilize training. All models were trained with $N = 100$ object queries.

Backbone and Transformer

We use the same ResNet-50 backbone from Torchvision for all experiments, with weights from either ImageNet or MS-COCO, depending on the setting and a batch size of one image per GPU. Backbone batch normalization weights and statistics are frozen during training as it has been widely adopted in object detection. Two separate learning rates are used for the backbone and the transformer, 5^{-06} and 5^{-05} respectively. The transformer weights are initialized with Xavier initialization [55].

Faster-RCNN baseline

We take a Faster-RCNN model, provided by the Detectron2 framework [216] with weights from either ImageNet or Ms-COCO depending on the setting, and fine tune it on the OpenLogo dataset. For this Faster-RCNN baseline we use the same data augmentation techniques as in the DETR and use a training schedule of with $3x$ iterations (around 40 epochs). We use a batch size of two images per GPU and a learning rate of 0.02 with cosine annealing [117]. The rest of the settings are those set by default in the Detectron2 model zoo [216].

Positional encoding

The positional encoding is used to represent the association between encoder activations and their corresponding image features. The one used in DETR-FP adopts a generalization of the encoding in the original Transformer [199], but adapted to the 2D case [139]. Specifically, for a feature map $f \in \mathbb{R}^{d \times H \times W}$, $\frac{d}{2}$ sine and cosine functions with different frequencies are used for both spatial coordinates of each embedding independently. These embeddings are then concatenated to get the final $d \times H \times W$ channel positional encoding which is added to the input features.

Object Queries

The object queries are learned embeddings tasked with *looking* at different regions of the input feature map and *querying* it for boxes of different sizes

and shapes. One can think of them as learned anchors, that can generalize to objects of different shapes and sizes. As a result, each query specializes on certain areas and box sizes.

2.3.1 DETR vs. Faster-RCNN

We show qualitative and quantitative results of DETR in both MS-COCO [110] and OpenLogo [183] along with a comparison against a Faster-RCNN baseline. We first provide a deeper analysis into the results obtained for small objects using the original DETR architecture, and subsequently by decomposing the performance of each model into different types of errors using TIDE [18]. This will allow us to get a deeper insight into DETR performance based on the self-attention feature map for the encoder and the decoder around points of interest. In addition, these results will justify the use of Feature Pyramids, as a proper, coherent extension of DETR.

There are several differences between the training setting of DETR and the one used in Faster-RCNN. Transformers are usually trained with very long training schedules and with Adam or Adagrad optimizers. In the case of DETR the models are trained for 500 epochs with AdamW optimizer [118]. Faster-RCNN, however has a much shorter training schedule and is trained with SGD.

Despite these differences we use a Faster-RCNN baseline for comparison. Localizing small objects is a challenge, but it can be alleviated by replacing the 2×2 stride in the last group of the ResNet backbone with dilated convolution, resulting in a larger feature map. This is a standard practice in object detection. Table 2.1 shows how DETR matches the performance of the baseline in MS-COCO, with less operations (GFLOPS) and less parameters. Furthermore, using a transformer architecture allows us to visualize the attention maps and gain an insight into what each part of the model is trying to do. In the case of DETR the encoder seems to encode each instance of every object in an image in order to separate them, while the decoder focuses on localizing each of these instances in order to identify them. This behaviour is clearly illustrated in Figure 2.2.

Logo detection is a rather difficult problem for DETR. Logos are usually small, and this is precisely the main weakness of this approach. Dilation can help alleviate this problem as evidenced above, however it is very expensive to do in DETR, computationally and memory-wise, due to the increase in complexity (see section 2.2.3). This is not possible to do with our hardware, so we do not dilate the backbone in our experiments. Table 2.2 shows the results of both Faster-RCNN in DETR in the OpenLogo benchmark.

2.3 Experiments and Results

Table 2.1: Comparison with Faster-RCNN using a ResNet-50 on MS-COCO validation set. The + sign indicates an extended training schedule ($\sim 108epochs$) and DC5 indicate a dilated backbone. The dilated DETR model outperforms the rest, specially on large objects, however its performance on small objects is not as good as a Faster-RCNN model. (numerical results taken from [22]).

Model	GFLOPS	#params	AP	AP ₅₀	AP ₇₅	AP _S	AP _M	AP _L
Faster RCNN-DC5	320	166M	39.0	60.5	42.3	21.4	43.5	52.5
Faster RCNN-FPN	180	42M	40.2	61.0	43.8	24.2	43.5	52.0
Faster RCNN-DC5+	320	166M	41.1	61.4	44.3	22.9	45.9	55.0
Faster RCNN-FPN+	180	42M	42.0	62.1	45.5	26.6	45.4	53.4
DETR	86	41M	42.0	62.4	44.2	20.5	45.8	61.1
DETR-DC5	187	41M	43.3	63.1	45.9	22.5	47.3	61.1

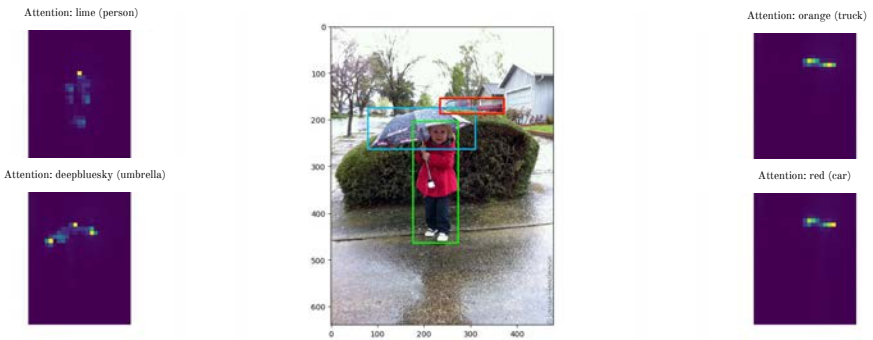
Table 2.2: The Faster-RCNN model without dilation or FPN falls clearly behind of DETR with the same conditions. The DC5 model has a comparable performance with DETR, while the FPN model outperforms the rest, except for large objects, where DETR excels.

Model	Weights	Schedule	AP	AP ₅₀	AP ₇₅	AP _S	AP _M	AP _L
Faster RCNN	ImageNet	40	31.8	53.1	33.2	15.9	32.27	44.9
Faster RCNN	MS-COCO	40	32	49.7	36.3	14.9	32.2	45.6
Faster RCNN-DC5	ImageNet	40	38.5	57.1	43.9	22.4	38.9	51.6
Faster RCNN-DC5	MS-COCO	40	40.0	62.0	45.2	24.4	41.8	51.6
Faster RCNN-FP	ImageNet	40	39.9	58.7	45.5	24.1	41.0	51.0
Faster RCNN-FP	MS-COCO	40	40.5	62.0	46.0	25.9	42.0	51.1
DETR	ImageNet	300	13.5	24.2	13.5	6.0	13.3	21.87
DETR	MS-COCO	300	38.6	57.6	43.3	24.7	39.2	52.0

It is clear that without a dilated backbone Faster-RCNN struggles considerably in comparison with DETR. However, with the addition of an FPN or by dilating the backbone, the performance is very similar. Also notice that DETR needs MS-COCO weights in order to perform well due to the absence of priors and the complex relations that the transformer has to learn, it becomes impossible to train the model in a reasonable amount of time with a small dataset such as OpenLogo. We visualize the attention maps for the encoder and decoder for some instances in the OpenLogo dataset (see Figure 2.3). Notice that in spite of the encoder struggling to separate instances of small objects, the decoder localizes them correctly.



(a) DETR encoder self-attention feature map, visualized around the center of four random ground truth annotations.



(b) DETR decoder attention map, for the four detections with highest score.

Figure 2.2: Visualization of DETR encoder and decoder attention in a random instance of the MS-COCO validation set. The decoder looks at corners in the objects in order to localize them properly while the encoder isolates each instance of an object individually, even when the bounding boxes of these objects are overlapped. When an object is very small in an image the DETR encoder fails to properly isolate it. This is the case for the car in the top left corner of the image.

2.3.2 Hyperparameter search

There are a fair number of parameters to set in DETR, the weight for each different criteria, the background coefficient, learning rates, to name a few. In an attempt to find the optimal values for these parameters for the OpenLogo benchmark. We conducted random search over all of them, during 50 epochs



(a) DETR encoder self-attention feature map, visualized around the center of four random ground truth annotations.



(b) DETR decoder attention map, for the four detections with highest score.

Figure 2.3: Visualization of DETR encoder and decoder attention in a random instance of the OpenLogo validation set. The decoder looks at corners in the objects in order to localize them properly while the encoder isolates each instance of an object individually. When the objects are small, the encoder fails to isolate them and the self-attention is blurred and unfocused. Even when this happens, most of the time, the decoder manages to localize the object and the model detects it correctly.

(for time purposes). The results can be best visualized using a parallel plot (see Figure 2.4). Clearly the values of the parameters vary the results significantly, but it is hard to tell which one is more important.

In order to establish the importance of each parameter one can use correlation between each hyperparameter and the metric we are trying to maximize. However correlation cannot capture second order interactions between inputs and it can behave poorly when comparing inputs with wildly different ranges. This importance can be obtained by training a random forest classifier with the hyperparameters as inputs and the metric one is trying to maximize as

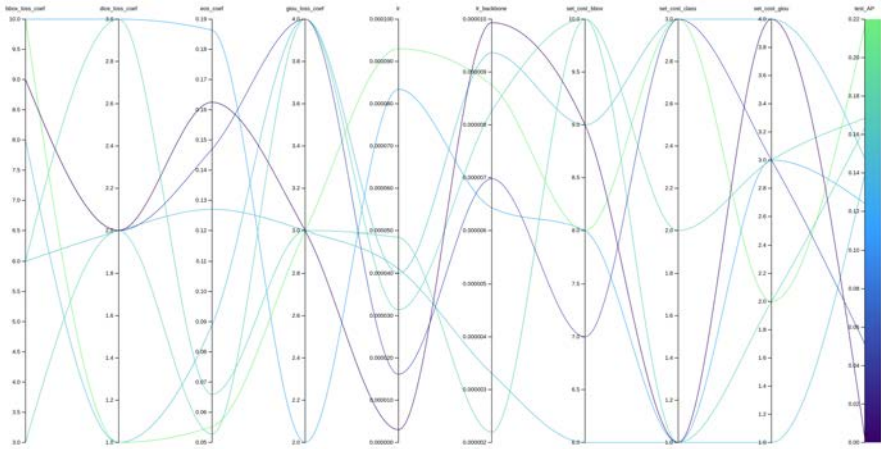


Figure 2.4: Parallel plot indicating the values selected by random search for each experiment and the resulting mAP. Each line represents one run. The greener the line, the higher the mAP, the opposite is true for blue lines. It seems that a high learning rate and a low weight for the no object class (*eof_loss_coef*) influence the mAP in a positive way. Keep in mind that in order to find optimal hyperparameters, more experiments over a greater range of values need to be run.



Figure 2.5: Importance of each parameter and its correlation with the mAP. Red and green indicate negative and positive correlation respectively. The importance and correlations values confirm that the learning rate and the relative classification weight of the no-object class (*eof_loss_coef*) are the hyperparameters that influence the performance of the model the most.

target and report the feature importance values for the random forest classifier [1]. The importance of each parameter and their correlation with the mAP, is shown in Figure 2.5.

2.3.3 Decomposing the performance

The standard metric used today for object detection is mean average precision (mAP). A complicated term that involves integrating over precision-recall curve and averaging over several criteria. There are many sources of errors that affect mAP, and yet, all we have to analyze the performance of our model is this number. Therefore it is hard to analyze which sources of error (e.g, classification, duplicate detections, localization, misses) is contributing the most to the errors our model is making. TIDE [18] breaks down the missing mAP into six types of errors, that fully explain where the model is losing performance. Let us denote IoU_{max} to denote the maximum IoU overlap of a false positive with a ground truth of the given category, t_f as the foreground IoU threshold and t_b as the background IoU threshold. Then, the six error types are defined as follows:

1. **Classification error:** $IoU_{max} \geq t_f$ for GT of the *incorrect* class (i.e., localized correctly, but classified incorrectly)
2. **Localization error:** $t_b \leq IoU_{max} \leq t_f$ for GT of the *correct* class (i.e., classified correctly, but localized incorrectly).
3. **Classification and Localization error:** $t_b \leq IoU_{max} \leq t_f$ for GT of the *incorrect* class. (i.e., classified and localized incorrectly).
4. **Duplicate detection error:** $IoU_{max} \geq t_f$ for GT of the *correct* class, after a higher-scoring detection already matched that GT.
5. **Background error:** $IoU_{max} \leq t_b$ for all GT (i.e., detected background as foreground)
6. **Missed GT error:** All undetected GT (false negatives) not already covered by classification and localization error.

Using these metrics we can decompose the error of our models, and get an insight into what can be improved in order to boost their performance. As seen in Figure 2.6, the Faster-RCNN model struggles with the classification of objects and detects a lot of background as foreground. The former is alleviated by the addition of dilation in the backbone or an FP, but the latter remains present across all of the Faster-RCNN models, along with a high number of miss detections. In contrast the DETR-FP improves over DETR with the addition of the feature pyramid (FP).

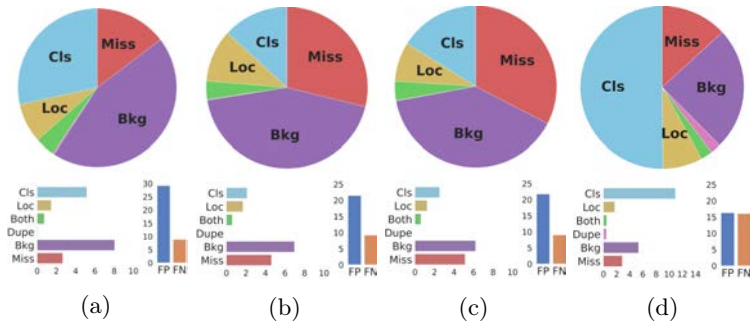


Figure 2.6: Performance of different models (a) Faster-RCNN, (b) Faster-RCNN-DC5, (c) Faster-RCNN-FPN, (d) DETR-FP with MS-COCO weights, fine tuned on the OpenLogo dataset. As one can see the Faster-RCNN model makes far more classification mistakes than the rest, due to the lack of dilation or an FPN. Furthermore all of the Faster-RCNN models suffer from a high background and miss detection errors, while DETR-FP suffer mostly from classification error.

Table 2.3: Quantitative results of each model in the OpenLogo validation set. It is clear that without dilation or low level feature information models make a lot of classification mistakes. *Cls*, *Loc*, *Dupe*, *Bkg*, *Miss* stand for the six types of errors described previously, while *FP*, *FN* stand for False Positive and False Negative.

Model	Cls	Loc	Both	Dupe	Bkg	Miss	FP	FN
Faster-RCNN	5.15	1.45	0.74	0.06	8.05	2.65	29.17	8.83
Faster-RCNN-DC5	2.11	1.69	0.59	0.02	6.97	4.62	21.46	9.18
Faster-RCNN-FP	2.53	1.24	0.61	0.01	6.21	5.14	21.71	9.00
DETR-FP (Our approach)	10.96	1.72	0.50	0.45	5.34	2.88	16.29	16.03

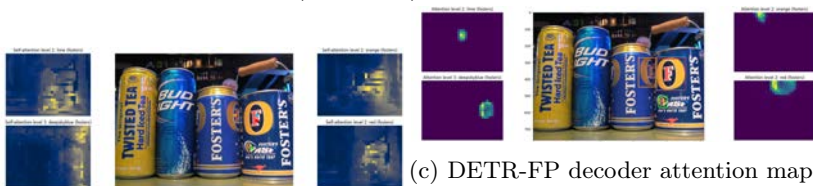
We also provide the exact numbers for the missing mAP decomposition, shown in Figure 2.6, these can be found in Table 2.3. Looking at the table it becomes clear that the performance of the model with dilation and FP are very similar, but the Faster-RCNN model falls clearly behind.

2.3.4 DETR-FP

In this section we provide some extra qualitative results on the OpenLogo validation set, by visualizing the detections for DETR-FP along with the attention of the transformer. Figure 2.7 shows DETR-FP predictions for one



(a) DETR-FP predictions and confidence (dashed lines) and GT (solid lines).



(b) DETR-FP encoder self-attention feature map, visualized around the center of four random ground truth annotations.

(c) DETR-FP decoder attention map, for the four detections with highest score. Keep in mind that the level 2 predictions are done on the 4 crops (top-left, top-right, bottom-left, bottom-right) of the second level of the pyramid.

Figure 2.7: DETR-FP Results for one image of the OpenLogo validation set. Notice how the queries have learned to *look* for small objects on the different crops of the second level of the pyramid and how despite the localization in the encoder does not seem useful or accurate, the decoder is capable of finding the object. However, we can observe some duplicate predictions due to the fact that the same object can be found in different levels of the feature pyramid.

image of the OpenLogo validation set. The figure shows how the queries in DETR-FP learn to *query* different crops of the lower levels of the feature pyramid for objects that are small or in the background. It is worth noticing that the model learns in which level to *look* for small, medium or large objects without any extra supervision.

Finally, we show in Figure 2.8 examples of small logo detections of DETR-FP, thanks to the Feature Pyramid.



Figure 2.8: Qualitative results of DETR-FP applied to logo detection. Note that logos are correctly detected on quite small regions.

2.4 Discussion

Within this work, we have elaborated on the enhancement of DETR, a novel fully differentiable end-to-end solution for object detection that eschews geometric priors and post-processing, focusing on the detection of small objects. We have amplified its performance by incorporating multi-level information via a Feature Pyramid (FP). Our comparison with the potent Faster-RCNN baseline in the MS-COCO and OpenLogo benchmarks yielded up to a 30

Nonetheless, the DETR-FP approach has significant areas for advancement. The substantial computational requirements impede us from leveraging the lower levels of the feature pyramid. Moreover, the introduction of varied

feature maps for the object queries to examine inflates the number of duplicate detections as the same object can be detected across different pyramid levels.

In light of these challenges, our prospective research aims to address these issues. Our goal is to enhance the performance of DETR-FP such that it exhibits a total improvement over robust baselines, which currently rely heavily on numerous geometric priors.

3 Embedding Propagation for Manifold Smoothing

Supervised training of neural networks requires a large amount of manually annotated data and the resulting networks tend to be sensitive to out-of-distribution (OOD) data. Self- and semi-supervised training schemes reduce the amount of annotated data required during the training process. However, OOD generalization remains a major challenge for most methods. Strategies that promote smoother decision boundaries play an important role in out-of-distribution generalization. For example, embedding propagation (EP) for manifold smoothing has recently shown to considerably improve the OOD performance for few-shot classification. EP achieves smoother class manifolds by building a graph from sample embeddings and propagating information through the nodes in an unsupervised manner. In this work, we extend the original EP paper providing additional evidence and experiments showing that it attains smoother class embedding manifolds and improves results in settings beyond few-shot classification. Concretely, we show that EP improves the robustness of neural networks against multiple adversarial attacks as well as semi- and self-supervised learning performance.

3.1 Related Work

In this paper, we study the effect of EP as a manifold regularization method and extend its use beyond few-shot learning to adversarial attacks, self- and semi-supervised learning. So we next review the literature for each one of these fields.

Regularization is a major area of research in machine learning [180, 207]. Whether it is directly enforcing constraints on networks weights [75, 163, 173], or regularizing the embedding manifold [10, 192, 229], regularization has been shown to aid models generalize. For example, TPN [114] introduces a meta-learning approach to label propagation by learning a graph construction module that exploits the manifold structure in the data. EPNet [164] attempts to smooth the class embedding manifold by applying an embedding propagation

operation on extracted features. Similarly, manifold mixup [201] leverages interpolations of the hidden layers of the network as an additional training signal and it has been shown to improve adversarial robustness and work well in a self-supervised setting. Techniques similar to embedding propagation have also been applied as a message passing algorithms in graph neural networks [89, 218], here we recast it as a regularization technique.

Adversarial attacks were introduced by [189]. The authors showed that convolutional neural networks are extremely sensitive to small perturbations in the input image. So visual perturbations imperceptible to the human eye are sufficient to cause the model to incorrectly misclassify an example with very high confidence.

Adversarial attacks can be classified into three categories depending on the knowledge of the attacker about the targeted model: white-box, gray-box and black-box. In a white box setting the adversary has full knowledge of the target model, including its parameters and architecture, so the attacker can easily craft adversarial examples by any means. In a gray-box threat model, only the structure of the target model is known to the attacker. Lastly, in a black-box threat model only the task of the target model is known [136], thus the attacker has to resort to query-level access to the black-box model in order to generate adversarial examples [24]. Since the introduction of FGSM [189], many adversarial attacks have been proposed, such as projected gradient descent (PGD) [121], Jacobian-based saliency map attack (JSMA) [137] and Fast Adaptive Boundary attack (FAB) [37].

Semi-supervised learning aims to leverage a set of unlabeled data in order to improve the performance on a downstream task [27]. [12] categorize the different semi-supervised learning methods into three categories: consistency regularization, entropy minimization, and traditional regularization, as detailed next.

Consistency regularization consists of performing extensive data augmentation [34, 177] to expand the decision boundaries of classifiers, so that they remain consistent on unlabeled data [96, 125, 172, 190].

Entropy minimization methods ensure that decision boundaries only pass through low-density regions, which is a common assumption in semi-supervised learning [27]. This property is here enforced by minimizing the entropy of the model outputs on unlabeled data [12, 58, 125]. Likewise, pseudo-label methods can reduce the entropy by directly discretizing the predictions of the model on unlabeled data [103].

Regularization techniques for semi-supervised learning constrain models to increase their bias in order to improve their generalization on unlabeled data [230]. The MixUp technique [229] is a popular regularization method that has been leveraged in multiple works to improve semi-supervised learning performance. In essence, the prediction at an interpolation of unlabeled points is forced to be consistent with the interpolation of the predictions at those points, thus moving the decision boundary to low-density regions of the data distribution. This strategy has been applied for interpolation consistency training (ICT) [202] and manifold mixup [201]. Similarly, the Mixmatch technique [12] uses MixUp to mix labeled and unlabeled data to produce pseudo-labels.

Embedding propagation is also considered as a MixUp strategy, since it leverages embedding interpolations based on a similarity graph, and intersects with the family of transductive semi-supervised learning methods [198]. These methods consider the relationship between instances in the test set to predict them as a whole, improving the performance of classifiers in the low-data regime. Likewise, embedding propagation also considers the relationships between instances by forming a graph from the query samples in an episode.

Self-supervised learning methods train models on unlabeled data by minimizing a contrastive learning loss or by learning to solve a pretext task. Current state-of-the-art self-supervised learning methods such as MoCo [32, 64] and SimCLR [29, 30], are based on contrastive learning: these methods use contrastive losses to measure the similarities of sample pairs in representation space. Approaches based on pretext tasks propose to solve an artificially designed proxy task. The underlying assumption is that by solving this proxy task, the model will acquire general knowledge required to solve the downstream tasks. A wide range of pretext tasks have been proposed, e.g., colorization [225, 230], recovering corrupted input (denoising) [204], forming pseudo-labels by transformations of a single image [42], patch ordering [41, 129] or tracking [212].

Recent works [21, 33, 67, 76, 215, 220] also investigate approaches around the selection of negative examples in self-supervised learning. [84] proposed using the hardest existing negatives to synthesize additional hard negatives on the fly, i.e., directly in the feature space, by mixing two of the hardest negatives or by mixing the query itself with one of the hardest negatives. Similarly, applying embedding propagation in a self-supervised setting, also creates hard negatives and positives on the fly as a byproduct of the EP algorithm.

Few-shot learning denotes those methods that learn to solve a task from a very reduced set of labelled data. For example, one-shot classification

methods learn a classifier with just one example per class. Most few-shot learning methods are included in two broad categories: meta-learning and transfer-learning. Meta-learning aims to learn a representation that can be robustly adapted to a new problem with few samples. For instance, authors in [135, 179, 186, 206] embed input data into a Hilbert space and perform distance-based classification. Another examples are those optimization-based approaches [46, 150, 226] which learn a good initialization that can be adapted to solve a specific problem in few optimization steps.

On the other hand, transfer-learning [31, 124] aims to learn generalisable representations from training data so that any new task can easily be solved with a simple classifier. Many of these approaches build on top of a pre-trained feature extractors [171, 213]. For example, authors in [123] introduced self-supervision to learn more transferable representations. Also, graph-based approaches [73, 88, 114] have been proposed to leverage the relationships between the samples in each episode by forming a graph and propagating information between nodes. In particular, Embedding Propagation [164] is a graph-based approach that uses a non-parametric operation [232] to propagate information between the nodes, achieving smoother decision boundaries and better few-shot generalization. In this work we show that EP offers improvement beyond few-shot learning and we extend it to other settings such as adversarial robustness and self- and semi- supervised learning. Different from [164], the goal of this work is to delve deeper into the benefits of EP rather than achieving state-of-the-art performance.

3.2 Methodology

In this paper we extend the embedding propagation method introduced by [164], whose basis is described in this section. Given a set of features $Z \in \mathbb{R}$ extracted from some input $X \in \mathbb{R}$ by a feature extractor $f : X \rightarrow Z$, EP maps those features to a set of interpolated features. Finally, the output of EP is fed into a classifier to label the images. [164] found that EP smooths the classification surface by pushing the boundaries away from the data, improving generalization [6, 105]. The EP approach differs from label propagation [232] and TPN [114] in that EP is completely unsupervised see Figure 3.1. Furthermore TPN is a meta-learning approach, to label propagation, hence it requires learning a graph construction module beforehand. Next we describe the EP algorithm in more detail.

In the image classification domain, embedding propagation takes a set of feature vectors $\mathbf{z}_i \in \mathbb{R}^m, i \in 1..|Z|$, obtained from applying a feature extractor

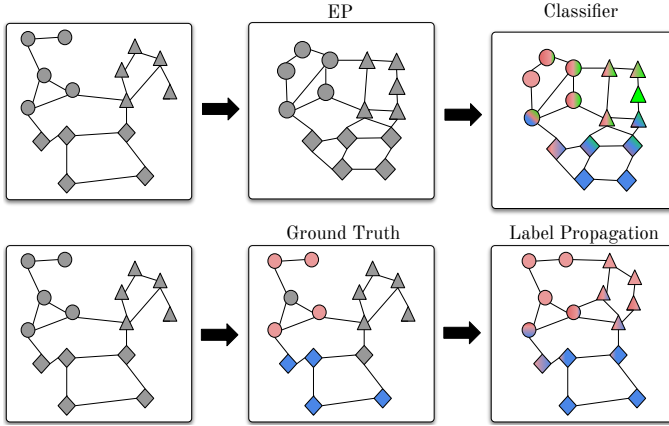


Figure 3.1: Illustration of the embedding propagation method, in comparison with label propagation (LP). The grey nodes represent unlabeled samples. Since EP is completely unsupervised, it does not require labels to reorganize the manifold and create a smooth classification surface. In contrast, LP requires initial labels. Note how EP increases the similarity between all pairs of points and forces samples that are close together to have similar classification score. *Best viewed in color.*

(CNN) to the input images. Then, it outputs a set of embeddings $\tilde{\mathbf{z}}_i \in \mathbb{R}^m$ through the following two steps. Firstly, for each pair of features (i, j) , the model computes the distance as $d_{ij}^2 = \|\mathbf{z}_i - \mathbf{z}_j\|_2^2$ and the adjacency matrix as $A_{ij} = \exp(-d_{ij}^2/\sigma^2)$, where σ^2 is a scaling factor and $A_{ii} = 0, \forall i$, as done in TPN [114]. The authors of the original paper chose $\sigma^2 = \text{Var}(d_{ij}^2)$ which was found to stabilize training.

Secondly, the Laplacian of the adjacency matrix is computed,

$$L = D^{-\frac{1}{2}} A D^{-\frac{1}{2}}, \quad D_{ii} = \sum_j A_{ij}. \quad (3.1)$$

Finally, the propagator matrix P is obtained using the label propagation formula described in [232] as,

$$P = (I - \alpha L)^{-1}, \quad (3.2)$$

where $\alpha \in \mathbb{R}$ is a scaling factor, and I is the identity matrix. As a result, the

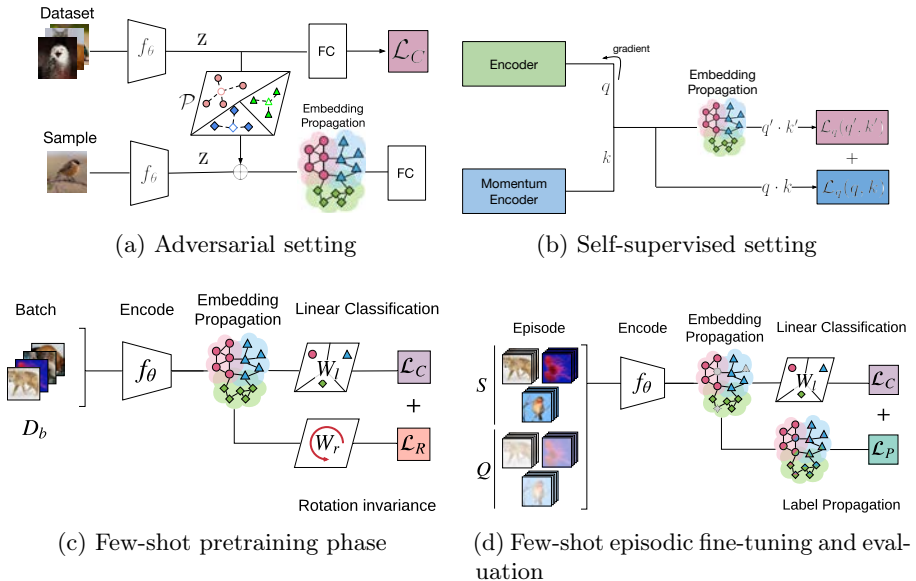


Figure 3.2: Overview of the EPNet training procedure across different tasks. **(a)** Non-transductive (inductive) version of the embedding propagation algorithm. First, the prototypes \mathcal{P} are built during the last training epoch. Then, at test time, EP is applied on the prototypes along with a single test sample. **(b)** Integration of the embedding propagation algorithm in MoCo. During the pre-training phase EP is applied on the keys and queries and the output is used in a secondary contrastive loss $\mathcal{L}_q(q', k')$. For few-shot learning **(c, d)**, the model is trained to learn general feature representations using a standard classification loss \mathcal{L}_C and an auxiliary rotation loss \mathcal{L}_R (left). Then, the model is fine-tuned using episodic learning to learn to generalize to novel classes by minimizing the standard classification loss \mathcal{L}_C and a label propagation loss \mathcal{L}_P (right).

embeddings are obtained as follows,

$$\tilde{\mathbf{z}}_i = \sum_j P_{ij} \mathbf{z}_j. \quad (3.3)$$

Notice that $\tilde{\mathbf{z}}_i$ are now a weighted sum of their neighbors. Thus, we hypothesize that undesired noise in the feature vectors is reduced after being averaged out

by the embedding propagation operation. This operation is simple to implement and compatible with a wide range of feature extractors and classifiers. Further, note that the computational complexity of our approach is $\mathcal{O}(n^2)$, which is similar to the complexity of the label propagation algorithm [232] as discussed in [208]. Further, note that the computational complexity of Eq. 3.2 is negligible for few-shot episodes [114] since the size of the episode is small[77].

Smoothness measure We use the Laplace operator Δ or Laplacian to measure the smoothness of the decision surface around a set of embeddings before and after applying the embedding propagation. The Laplacian is given by the sum of second partial derivatives of a function with respect to each independent variable:

$$\Delta f(x, y) = \sum_{i=1}^n \frac{\delta^2 f}{\delta x_i^2} + \frac{\delta^2 f}{\delta y_i^2}, \quad (3.4)$$

where x and y represent the two dimensions in the Cartesian coordinate frame and f is a classification function. Since we are only interested on the Laplacian around the decision boundary, we generate the minimal mesh that contains all the datapoints and use the discrete laplace operator in the form of a convolution:

$$\Delta f(x, y) = \mathbf{D}_{xy}^2 * f, \quad (3.5)$$

where $*$ is the convolution operator and \mathbf{D}_{xy}^2 is the Laplacian convolution kernel [78]. For each point in the grid, the absolute value of the magnitude of the Laplacian indicates a sharp change in the decision boundary. Thus, we approximate the total surface smoothness as the the definite integral of the Laplacian on the 2d grid. Since the grid is discrete, we compute smoothness \mathcal{S} as the inverse of the summation over all the grid values:

$$\mathcal{S} = \sum_y \sum_x \frac{1}{1 + |\Delta f(x, y)|}. \quad (3.6)$$

3.3 Experiments and Results

Next we present additional evidence of how EP smooths the classification surface and adapt it to different settings: adversarial attacks, self- and semi-supervised learning and few-shot learning. Although EP is applied at different

stages of the machine learning pipeline for each of the following experiments (see Figure 3.2), the EP algorithm will remain unchanged across all experiments.

3.3.1 Datasets

miniImagenet [150] consists of a subset of the Imagenet dataset [170] comprised of 100 classes with 600 images per class. Classes are divided in three disjoint sets of 64 base classes, 16 for validation and 20 novel classes.

tieredImagenet [155] is a more challenging subset of the Imagenet dataset [170] where class subsets are chosen from supersets of the wordnet hierarchy. The top hierarchy has 34 super-classes, which are divided into 20 base (351 classes), 6 validation (97 classes) and 8 novel (160 classes) categories.

CIFAR10 [90] is comprised of 60,000 32x32 colour images divided into 10 classes, with 6,000 images per class. There are 50,000 training images and 10,000 test images.

CIFAR100 [90] is just like the CIFAR10 dataset, except it has 100 classes containing 600 images each. There are 500 training images and 100 testing images per class.

MNIST [102] is a dataset of 70,000 small 28x28 pixels gray-scale images of handwritten single digits between 0 and 9 (10 classes). There are 60,000 examples in the training dataset and 10,000 in the test dataset.

Fashion-MNIST [219] is a dataset of Zalando's article images consisting of a training set of 60,000 examples and a test set of 10,000 examples. Each example is a 28x28 grayscale image, associated with a label from 10 classes.

STL-10 [35] is a dataset of 96x96 color images, categorized into 10 classes, with 500 training images and 800 test images per class. The dataset also has 100,000 unlabeled images for unsupervised learning. Images were drawn from Imagenet labeled examples.

3.3.2 Manifold smoothness

The embedding propagation algorithm is based on the closed-form solution of the label propagation algorithm proposed by [232]. One of the main advantages

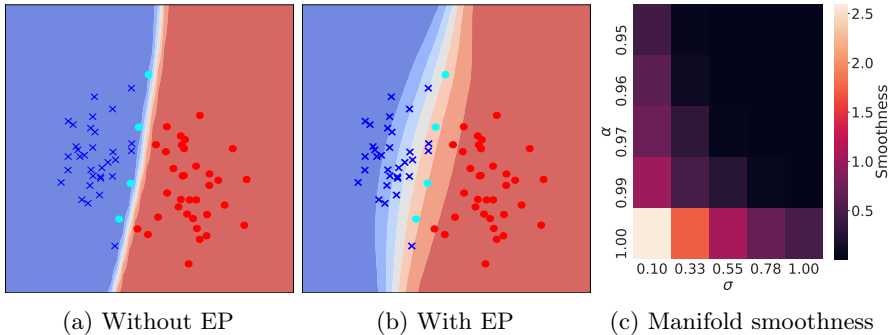


Figure 3.3: **(a, b)** Comparison of the class embedding manifold without and with embedding propagation on a toy classification dataset. Notice how without EP **(a)** the adversarial examples (cyan) cross the decision boundary and are misclassified, the smoothness achieved by applying EP **(b)** at test time on the same classifier prevents this misclassification. **(c)** Effect of α and σ on the smoothness of the class embedding manifold. The higher the α and the smaller the value of σ the smoother the manifold becomes, notice the lower diagonal of the matrix. Smoothness is given by Equation 3.6

of label propagation is that the decision boundaries are smooth with respect to the structure of the data [232] and this is a desirable property for semi-supervised learning algorithms [27] since it encourages points that are close together in embedding space to share the same label. This is important to propagate label information from labeled to unlabeled datapoints. We have included this explanation in Section 4.2. Here, we investigate if the decision boundaries remain smooth when propagation is performed directly in embedding space (see Equation 3.3) instead of the output space.

Experimental setup According to [100] a smooth function is that in which $f(x) = f(x + \epsilon)$ for small values of ϵ . In order to assess smoothness before and after applying embedding propagation, we generate a 2D toy dataset of randomly sampled embeddings with their corresponding labels. The dataset has two classes and 50 data points per class. Samples from both classes are drawn from two different, opposing gaussians. We resorted to a low-dimensional dataset since other real datasets such as *mini*Imagenet would require dimensionality reduction techniques for visualization, resulting in loss of information.

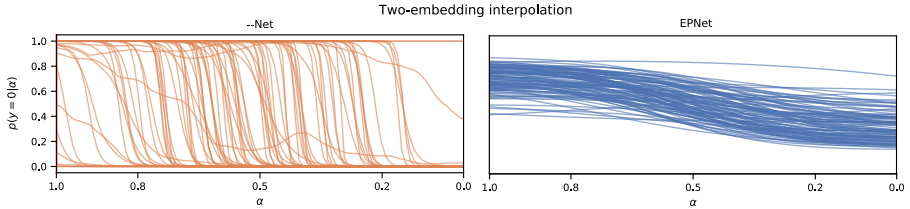


Figure 3.4: Interpolation of embedding pairs for two random data points of the *mini*Imagenet dataset with different classes vs probability of belonging to the first of these two classes. The right figure shows the class probability for Resnet-12 embeddings extracted from EPNet, and the left figure (-Net) from the same network trained without embedding propagation. The scalar α controls the weight of the first embedding in the linear interpolation.

Results Defining a term to empirically measure the smoothness of the classification surface allows us to measure how the hyperparameters that control embedding propagation (Sec 3.2) affect the smoothness of the class embedding manifold. In EP (Eq. 3.2), the hyperparameter α controls the amount of propagation performed in the graph and σ is the radius of the RBF function used to calculate the similarity matrix. Therefore, the value of α should be directly correlated with \mathcal{S} (Eq. 3.6) and the value of σ inversely correlated with \mathcal{S} . Figure 3.3c shows that the former hypothesis holds, thus showing that α and σ control the smoothness of the class embedding manifold. Furthermore, we show the smoothing effect of EP on a toy classification dataset in Figure 3.3b (more details can be found in the Appendix). The manifold hypothesis from semi-supervised learning theory holds that smoother decision boundaries aid generalization, as shown in [201]. Hence by applying EP, encouraging smoother decision boundaries, we improve the classification of adversarial examples.

Lastly, to further reinforce the smoothness hypothesis, we visualize embedding interpolations with and without embedding propagation. We use EPNet to obtain image embeddings and select a set of random pairs $\mathbf{z}_i, \mathbf{z}_j$ that belong to different classes y_i, y_j . We then interpolate between each pair as $\tilde{\mathbf{z}} = \alpha \cdot \mathbf{z}_i + (1 - \alpha)\mathbf{z}_j$ where $\alpha \in [0..1]$, and plot this value against $p(y_i|\tilde{\mathbf{z}})$ in Figure 3.4. We also plot $p(y_i|\tilde{\mathbf{z}})$ where embeddings were obtained using EPNet without embedding propagation (-Net). We observe that EPNet has significantly smoother probability transitions than -Net as the embedding $\tilde{\mathbf{z}}$ changes from \mathbf{z}_i to \mathbf{z}_j . In contrast, -Net yields sudden probability transitions. This suggests that embedding propagation encourages smoother decision boundaries.

Table 3.1: Adversarial attacks results across four different datasets. Notice that manifold mixup fails against iterative perturbations (PGD [121], FAB [37]), while EP, despite only being applied at test time, increases adversarial robustness considerably. Furthermore, a combination of both regularization methods improves results substantially across datasets for [37]. We report the average of five different runs. Vanilla refers to a setting where neither EP nor mixup is applied.

	CIFAR10	CIFAR100	MNIST	FashionMNIST
No perturbation				
Vanilla	93.65 \pm 1.23	76.37 \pm 1.41	99.37 \pm 0.04	94.59 \pm 0.18
Mixup	93.49 \pm 0.65	72.72 \pm 1.89	99.21 \pm 0.12	94.78 \pm 0.22
EP	94.30 \pm 0.39	76.13 \pm 1.27	99.32 \pm 0.08	94.66 \pm 0.15
EP + Mixup	92.71 \pm 0.46	71.76 \pm 1.07	99.39 \pm 0.06	94.53 \pm 0.20
FGSM [57]				
Vanilla	17.24 \pm 1.09	6.59 \pm 0.23	61.77 \pm 20.37	45.07 \pm 1.73
Mixup	18.78 \pm 2.55	5.46 \pm 0.57	84.234 \pm 10.98	38.88 \pm 10.64
EP	31.24 \pm 0.67	9.59 \pm 0.47	84.44 \pm 7.40	61.74 \pm 4.65
EP + Mixup	20.89 \pm 2.69	6.38 \pm 0.64	81.34 \pm 10.68	57.92 \pm 7.74
PGD [121]				
Vanilla	0.006 \pm 0.004	0.01 \pm 0.01	22.82 \pm 12.15	0.81 \pm 0.46
Mixup	0.03 \pm 0.01	0.02 \pm 0.01	29.09 \pm 14.20	2.45 \pm 0.45
EP	11.70 \pm 0.90	3.01 \pm 0.73	43.6 \pm 23.90	22.99 \pm 5.97
EP + Mixup	8.79 \pm 0.75	0.85 \pm 0.09	52.86 \pm 24.04	19.01 \pm 7.22
FAB [37]				
Vanilla	0.58 \pm 0.07	0.09 \pm 0.02	0.03 \pm 0.01	0.09 \pm 0.02
Mixup	0.69 \pm 0.11	0.09 \pm 0.01	1.91 \pm 2.04	3.16 \pm 2.20
EP	5.64 \pm 0.33	5.85 \pm 0.21	14.20 \pm 6.28	10.07 \pm 0.94
EP + Mixup	9.95 \pm 0.98	8.24 \pm 1.35	61.99 \pm 20.41	35.12 \pm 3.16

3.3.3 Adversarial robustness

Few-shot learning algorithms are tested outside of the original distribution given that few-shot learning datasets use a disjoint set of test classes. Similarly, adversarial attacks try to modify a sample to move it outside of the original training distribution in order to cause unexpected behavior of the model. [201] showed that smoother decision boundaries improve adversarial robustness. Like-

wise, in the previous experiment, we have shown that embedding propagation has a smoothing effect on the class embedding manifold, similar to the effect caused by manifold mixup [201]. Therefore, in this section we explore whether similar benefits are observed from applying embedding propagation.

Experimental setup Adversarial attacks exploit the linear nature of neural networks and their difficulty generalizing to OOD data. They imperceptibly modify an input sample as to cause misclassification with high confidence. In this work we focus on white box attacks, where the attacker has full access to the model gradients. Concretely, we evaluate our method on: FGSM [57], PDG [121] and FAB [37] attacks.

In this setting, we only consider one test sample at a time in order to make embedding propagation non-transductive and decouple its performance from the ordering of the batch. However, EP requires multiple embeddings in order to build a graph (Eq. 3.1). To address this issue, we compute class prototypes from the training dataset and use those prototypes to form a graph for each test sample. Let $Z \in \mathbb{R}^k$ be the output of a feature extractor, \mathcal{C} the set of classes in our dataset and N the total of samples in our training set. Then the prototypes matrix is defined as $\mathcal{P} \in \mathbb{R}^{k \times \mathcal{C}}$ and it is computed as:

$$\mathcal{P}_c = \frac{1}{N_c} \sum_{i \in c}^N z_i, \forall c \in \mathcal{C} \quad (3.7)$$

where N_c is the number of examples belonging to class c . Notice that obtaining the prototype matrix \mathcal{P} does not require any additional training, a forward pass on the training dataset is all that is required. At test time, we apply EP on the concatenation of \mathcal{P} with the embedding of single data point z_i , resulting in \tilde{z}_i (Eq. 3.2). Then the classifier is applied to \tilde{z}_i . This process is illustrated in Figure 3.2a. Note that we only apply the embedding propagation operation at test time, when the adversarial attacks are performed, since we observed similar results when applying it both at train and at test time.

Results As seen in Table 3.1 EP increases adversarial robustness against strong iterative perturbations, with an average improvement with respect to manifold mixup of 12.17% against [121] and 7.85% against [37]. Furthermore, we show that EP and manifold mixup are not mutually exclusive, and they can be combined to improve performance against FAB attacks [37] with an improvement of up to 47%. It is worth noticing that manifold mixup, improves adversarial robustness against single step attacks, but fails against iterative perturbations, despite being applied during training. Conversely, EP improves

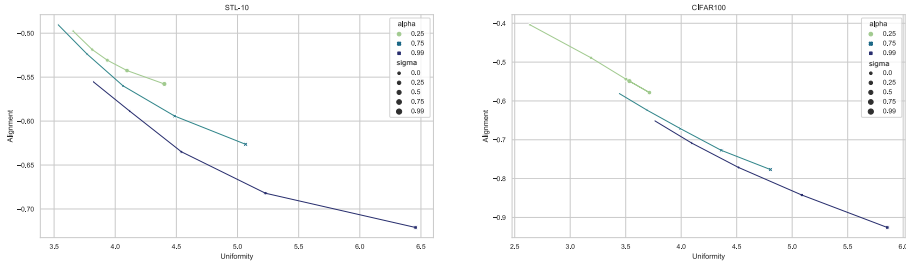


Figure 3.5: *Alignment* and *Uniformity* values obtained for different values of α and σ on the STL-10 (left) and CIFAR100 (right) datasets. The x and y axis correspond to $-\mathcal{L}_{Uniform}$ and $-\mathcal{L}_{align}$ of [211], respectively. Harder positives and negatives make the embedding space more uniform and less aligned.

Table 3.2: Self-Supervised results for STL-10 and CIFAR100 datasets where both Manifold mixup and embedding propagation are applied in the same way during MoCo pre-training. We report the average of five different runs.

	STL-10	CIFAR100
MoCo	85.28 \pm 0.75	74.68 \pm 0.18
MoCo + Manifold Mixup	85.75 \pm 0.48	74.85 \pm 0.31
MoCo-EP	86.02 \pm 0.65	75.02 \pm 0.56

robustness against both single and multiple step attacks, while being applied at inference time only with no additional training required.

3.3.4 Self-supervised Learning

We experiment on the self-supervised and semi-supervised learning scenarios, where the model has to learn from an unlabeled set of examples. Mixup [229] has been shown to improve results in these scenarios. Works such as [84, 203] leverage embedding interpolations to create hard positives and hard negatives, resulting in improved self-supervised learning performance.

Manifold mixup [201] and EP [164], also have a smoothing effect on the classification surface. In this section, we explore how embedding propagation compares to manifold mixup in a self-supervised scenario. Notice that this effect is a natural byproduct of Equation 3.3 and thus the embedding propagation algorithm itself requires no modifications from its original implementation to be applied in self-supervised learning.

Experimental setup We adapt the original MoCo [64] implementation to integrate embedding propagation. The pre-training process is shown in Figure 3.2b. In this setting we use EP to generate new embeddings and use them in an additional contrastive loss. Consider an encoder that outputs an encoded query q and a momentum encoder that outputs coded samples k that are keys of a dictionary. Letting only one key k_+ to match the query q , the contrastive loss function used in MoCo can be defined as

$$\mathcal{L}_q(q, k) = -\log \frac{\exp(q \cdot k_+ / \tau)}{\sum_{i=0}^K \exp(q \cdot k_i / \tau)}, \quad (3.8)$$

where τ is a temperature hyper-parameter. We introduce an additional loss where the embedding propagation is applied; $EP(q \oplus k)$ to obtain new queries q' and keys k' . Thus, the criteria to optimize becomes:

$$\mathcal{L}_q = \alpha \mathcal{L}_q(q, k) + (1 - \alpha) \mathcal{L}_q(q', k') \quad (3.9)$$

where α is a weighting hyper-parameter set to 0.6 (found through random search) for all experiments. In contrast to label propagation [232], embedding propagation is completely unsupervised, which makes it possible to apply it during MoCo’s pre-training phase. For comparison, we also provide results with manifold mixup [201] applied to q and k in the same way as EP. The main difference between the two methods is that manifold mixup considers random pairs of samples while EP takes into account the topology of the data. The hyperparameters manifold mixup’s Dirichlet distribution are the best found through random search.

Results As seen in Table 3.2 embedding propagation increases the validation accuracy with respect to a MoCo baseline by 0.74% in STL-10 and by 0.34% in CIFAR100. EP also outperforms manifold mixup in both datasets by 0.27% and 0.17%, respectively. We hypothesize that the improvement is due to the creation of artificial hard negatives and positives by the embedding propagation operation during the training process. In fact, [203] showed that mixup can be used in a self-supervised setting to synthesize hard positives. Similarly, EP naturally synthesizes hard positives and negatives taking into account the topology of the data.

Recently [211] proposed two losses or metrics for assessing the quality of contrastive learning representations. The first one (\mathcal{L}_{align}) measures the absolute distance between representations with the same label, while the second one ($\mathcal{L}_{Uniform}$) measures how uniformly distributed are the representations in the hyper-sphere. In Figure 3.5, we show how the alignment decreases and the

3.3 Experiments and Results

Table 3.3: Comparison of test accuracy against state-of-the art methods for Few-shot classification using *mini*Imagenet and *tiered*Imagenet with the 1-shot and 5-shot settings. The second column shows number of parameters per model in thousands (K). -Net is identical to EPNet but without EP. We report the average of 600 episodes.

	Params	<i>mini</i> Imagenet		<i>tiered</i> Imagenet	
		1-shot	5-shot	1-shot	5-shot
CONV-4					
Matching [206]	112K	43.56 \pm 0.84	55.31 \pm 0.73	-	-
MAML [114]	112K	48.70 \pm 1.84	63.11 \pm 0.92	51.67 \pm 1.81	70.30 \pm 0.08
ProtoNet [179]	112K	49.42 \pm 0.78	68.20 \pm 0.66	53.31 \pm 0.89	72.69 \pm 0.74
ReNet [186]	223K	50.44 \pm 0.82	65.32 \pm 0.70	54.48 \pm 0.92	71.32 \pm 0.78
GNN [49]	1619K	50.33 \pm 0.36	66.41 \pm 0.63	-	-
TPN [114]	171K	53.75 \pm 0.86	69.43 \pm 0.67	57.53 \pm 0.96	72.85 \pm 0.74
CC+rot [51]	112K	54.83 \pm 0.43	71.86 \pm 0.33	-	-
SIB [72]	112K	58.00 \pm 0.60	70.70 \pm 0.40	-	-
EGNN [88]	5068K	-	76.37 \pm N/A	-	80.15 \pm N/A
-Net (ours)	112K	57.18 \pm 0.83	72.57 \pm 0.66	57.60 \pm 0.93	73.30 \pm 0.74
EPNet (ours)	112K	59.32 \pm 0.88	72.95 \pm 0.64	59.97 \pm 0.95	73.91 \pm 0.75
RESNET-12					
ProtoNets++ [221]	7989K	56.52 \pm 0.45	74.28 \pm 0.20	58.47 \pm 0.64	78.41 \pm 0.41
TADAM [135]	7989K	58.50 \pm 0.30	76.70 \pm 0.30	-	-
MetaOpt-SVM [104]	12415K	62.64 \pm 0.61	78.60 \pm 0.46	65.99 \pm 0.72	81.56 \pm 0.53
TPN [114]	8284K	59.46 \pm N/A	75.65 \pm N/A	-	-
Robust-20++ [43]	11174K	58.11 \pm 0.64	75.24 \pm 0.49	70.44 \pm 0.32	85.43 \pm 0.21
MTL [185]	8286K	61.20 \pm 1.80	75.50 \pm 0.80	-	-
CAN [70]	8026K	67.19 \pm 0.55	80.64 \pm 0.35	73.21 \pm 0.58	84.93 \pm 0.38
BD-CSPN [111]	7989K	65.94 \pm N/A	79.23 \pm N/A	-	-
-Net (ours)	7989K	65.66 \pm 0.85	81.28 \pm 0.62	72.60 \pm 0.91	85.69 \pm 0.65
EPNet (ours)	7989K	66.50 \pm 0.89	81.06 \pm 0.60	76.53 \pm 0.87	87.32 \pm 0.64
WRN-28-10					
LEO [171]	37582K	61.76 \pm 0.08	77.59 \pm 0.12	66.33 \pm 0.05	81.44 \pm 0.09
Robust-20++ [43]	37582K	62.80 \pm 0.62	80.85 \pm 0.43	-	-
wDAE-GNN [52]	48855K	62.96 \pm 0.15	78.85 \pm 0.10	68.18 \pm 0.16	83.09 \pm 0.12
CC+rot [51]	37582K	62.93 \pm 0.45	79.87 \pm 0.33	70.53 \pm 0.51	84.98 \pm 0.36
Manifold mixup [123]	37582K	64.93 \pm 0.48	83.18 \pm 0.72	-	-
FEAT [224]	37582K	65.10 \pm 0.20	81.11 \pm 0.14	70.41 \pm 0.23	84.38 \pm 0.16
SimpleShot [213]	37582K	65.87 \pm 0.20	82.09 \pm 0.14	70.90 \pm 0.22	85.76 \pm 0.15
SIB [72]	37582K	70.00 \pm 0.60	79.20 \pm 0.40	-	-
BD-CSPN [111]	37582K	70.31 \pm 0.93	81.89 \pm 0.60	78.74 \pm 0.95	86.92 \pm 0.63
LaplacianShot [238]	37582K	74.86 \pm 0.19	84.13 \pm 0.14	80.18 \pm 0.21	87.56 \pm 0.15
TIM-GD[19]	37582K	77.80 \pm N/A	87.40 \pm N/A	82.10 \pm N/A	89.80 \pm N/A
-Net (ours)	37582K	65.98 \pm 0.85	82.22 \pm 0.66	74.04 \pm 0.93	86.03 \pm 0.63
EPNet (ours)	37582K	70.74 \pm 0.85	84.34 \pm 0.53	78.50 \pm 0.91	88.36 \pm 0.57

uniformity increases as the α and σ in the EP operation increase. Indicating that EP helps the proxy task to obtain a better representation of the embedding space as shown in [84] by creating hard-negatives and positives.

3.3.5 Few-Shot and Semi-supervised Learning

In this section we review the use of EP in the few-shot learning scenario. First we describe the experimental setup, then we provide implementation details, and finally we report the results. Note that we consider transductive few-shot as a form of semi-supervised learning.

Experimental setup We follow the common few-shot learning setup [155, 206] where three datasets are given: a *base* dataset (\mathcal{D}_b), a *novel* dataset (\mathcal{D}_n), and a *validation* dataset (\mathcal{D}_v). The base dataset is composed of a large amount of labeled images, the novel dataset is composed of labeled images from previously unseen classes and it is used to evaluate the transfer learning capabilities of a model. Lastly, the validation dataset \mathcal{D}_v contains classes not present in either \mathcal{D}_b or \mathcal{D}_n and is used to conduct hyperparameter search.

Furthermore, we have access to episodes. Each episode consists of n classes sampled uniformly without replacement from the set of all classes, a support set S (k examples per class) and a query set Q (q examples per class). This is referred to as n -way k -shot learning.

Given an episode, inference is performed by sequentially performing embedding and label propagation on features extracted from the input image. More formally, this is performed as follows. Let $\tilde{Z} \in \mathbb{R}^{(k+q) \times m}$ be the matrix of propagated embeddings obtained by jointly applying Eq. 3.1-3.3 to the support and query sets. Let $P_{\tilde{Z}}$ be the corresponding propagator matrix. Further, let $Y_S \in \mathbb{R}^{k \times n}$ be a one-hot encoding of the labels. We compute the logits for the query set (\hat{Y}_Q) by performing label propagation as described in [232].

For few-shot learning, we train the model in two phases. During the first phase we train two linear classifiers parametrized by W_l and W_r , respectively. The first classifier is trained to predict the class labels of examples in \mathcal{D}_b . It is optimized by minimizing the cross-entropy loss,

$$\mathcal{L}_c(\mathbf{x}_i, y_i; W_l, \theta) = -\ln p(y_i | \tilde{\mathbf{z}}_i, W_l), \quad (3.10)$$

where $y_i \in \mathcal{Y}_b$ and the probabilities are obtained by applying softmax to the logits provided by the neural network. For fair comparison with recent literature [51, 123] we also add a self-supervision loss. Hence, the second classifier is trained to predict image rotations, minimizing the following loss,

$$\mathcal{L}_r(\mathbf{x}_i, r_j; W_r, \theta) = -\ln p(r_j | \tilde{\mathbf{z}}_i, W_r), \quad (3.11)$$

where $r_j \in \{0^\circ, 90^\circ, 180^\circ, 270^\circ\}$, and $p(r_j | \tilde{\mathbf{z}}_i, W_r)$ is the probability of the input being rotated by r_j as predicted by a softmax classifier with weights W_r .

Thus the criteria to optimize in this first phase becomes:

$$\mathcal{L}_c(\mathbf{x}, y; W_l, \theta) + \mathcal{L}_r(\mathbf{x}, r; W_r, \theta). \quad (3.12)$$

In the second phase we use episodic training in order to generalize to novel classes. This process is illustrated in Figure 3.2d. In this phase, the model uses two classifiers. The first one is based on label propagation, and it computes class probabilities by applying a softmax to the query set logits \hat{Y}_Q .

$$\mathcal{L}_p(\mathbf{x}_i, y_i; \theta) = -\ln p(y_i | \tilde{\mathbf{z}}_i, \tilde{\mathbf{Z}}, Y_S). \quad (3.13)$$

The second classifier is used to predict the base classes as during the pre-training phase, and thus, it is identical to the W_l -based classifier used in pre-training. It is included to preserve a discriminative feature representation. Hence, the criteria to optimize becomes:

$$\operatorname{argmin}_{\theta, W_l} \left[\frac{1}{|Q|} \sum_{(\mathbf{x}_i, y_i) \in Q} \mathcal{L}_p(\mathbf{x}_i, y_i; \theta) + \frac{1}{|S \cup Q|} \sum_{(\mathbf{x}_i, y_i) \in S \cup Q} \frac{1}{2} \mathcal{L}_c(\mathbf{x}_i, y_i; W_l, \theta) \right]. \quad (3.14)$$

Implementation details For fair comparison with previous work, we used three common feature extractors: (i) a 4-layer convnet [179, 206] with 64 channels per layer, (ii) a 12-layer resnet [135], and (iii) a wide residual network (WRN-28-10) [171, 228]. For *mini* and *tiered* Imagenet, images are resized to 84×84 . Results for Imagenet-FS and few-shot semi-supervised learning can be found in [164]. We denote as EPNet the model resulting of combining these feature extractors with the EP procedure.

We evaluate 2 variations of our method: (i) EPNet as described in Eq. 3.1-3.3; (ii) -Net, which is identical to EPNet but without applying EP.

We also consider the few-shot semi-supervised learning scenario, where we have access to an unlabeled set of images U . We use the unlabeled set as follows. First, we use the same inference procedure as previously described to predict the labels \hat{c}_U for the unlabeled set as pseudo-labels. Then, we augment the support set with U using their pseudo-labels as the true labels. Finally, we apply the aforementioned inference procedure on the new support set to predict the labels for the query set.

Results are shown in Table 3.3. We compare the performance of the same neural network with and without EP (-Net and EPNet) against different few-shot classification methods across different backbones. EGNN uses a

Table 3.4: Semi-Supervised Learning (SSL) results with 100 unlabeled samples. -Net is identical to EPNet but without embedding propagation. *Re-implementation of [227]. We report the average of five different runs

	Backbone	<i>miniImagenet</i>		<i>tieredImagenet</i>	
		1-shot	5-shot	1-shot	5-shot
TPN _{SSL} [114]	CONV-4	52.78	66.42	55.74	71.01
k-Means _{masked,soft} [155]	CONV-4	50.41 \pm 0.31	64.39 \pm 0.24	-	-
-Net (ours)	CONV-4	57.18 \pm 0.83	72.57 \pm 0.66	57.60 \pm 0.93	73.30 \pm 0.74
EPNet (ours)	CONV-4	59.32 \pm 0.88	72.95 \pm 0.64	59.97 \pm 0.95	73.91 \pm 0.75
-Net _{SSL} (ours)	CONV-4	63.74 \pm 0.97	75.30 \pm 0.67	65.01 \pm 1.04	74.24 \pm 0.80
EPNet _{SSL} (ours)	CONV-4	65.13 \pm 0.97	75.42 \pm 0.64	66.63 \pm 1.04	75.70 \pm 0.74
LST [106]	RESNET-12	70.10 \pm 1.90	78.70 \pm 0.80	77.70 \pm 1.60	85.20 \pm 0.80
-Net (ours)	RESNET-12	65.66 \pm 0.85	81.28 \pm 0.62	72.60 \pm 0.91	85.69 \pm 0.65
EPNet (ours)	RESNET-12	66.50 \pm 0.89	81.06 \pm 0.60	76.53 \pm 0.87	87.32 \pm 0.64
-Net _{SSL} (ours)	RESNET-12	73.42 \pm 0.94	83.17 \pm 0.58	80.26 \pm 0.96	88.06 \pm 0.59
EPNet _{SSL} (ours)	RESNET-12	75.36 \pm 1.01	84.07 \pm 0.60	81.79 \pm 0.97	88.45 \pm 0.61
*k-Means _{masked,soft} [155]	WRN-28-10	52.78 \pm 0.27	66.42 \pm 0.21	-	-
TransMatch [227]	WRN-28-10	63.02 \pm 1.07	81.19 \pm 0.59	-	-
-Net (ours)	WRN-28-10	65.98 \pm 0.85	82.22 \pm 0.66	74.04 \pm 0.93	86.03 \pm 0.63
EPNet (ours)	WRN-28-10	70.74 \pm 0.85	84.34 \pm 0.53	78.50 \pm 0.91	88.36 \pm 0.57
-Net _{SSL} (ours)	WRN-28-10	77.70 \pm 0.96	86.30 \pm 0.50	82.03 \pm 1.03	88.20 \pm 0.61
EPNet _{SSL} (ours)	WRN-28-10	79.22 \pm 0.92	88.05 \pm 0.51	83.69 \pm 0.99	89.34 \pm 0.59

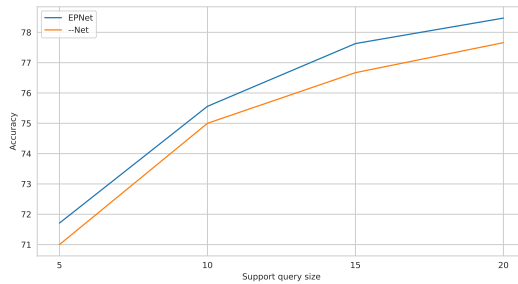


Figure 3.6: Performance of a small convolutional network (CONV-4) on the *miniImagenet* dataset with and without Embedding Propagation. Notice how the improvement obtained by EP is consistent for all query sizes, showing that EP remains effective in high-shot classification settings.

graph neural net on top of conv-4, hence the large amount of parameters. We observe that EP consistently improves the performance with respect to the same backbone without EP (-Net). We observe that the improvement is most

significant in the one-shot scenarios, since EP leverages unlabeled queries to improve the classification performance. Specifically, with the largest backbone (WRN-28-10), EP improves up to 5% and 2% in 1-shot and 5-shot respectively in *mini*Imagenet. Moreover, note that EP becomes more effective on higher capacity backbones, with an average improvement of 4% across datasets with a WRN-28-10 backbone. We hypothesize that these backbones provide more accurate embeddings that result in accurate graphs that attain more consistent information propagation between nodes.

Table 3.4 shows results in the SSL setting where 100 additional unlabeled samples are available [114, 155] (EPNet_{SSL}). Notice that including unlabeled samples increases the accuracy of EPNet for all settings, surpassing the state of the art by a wide margin of up to 16% accuracy points for the 1-shot WRN-28-10. Similar to previous experiments, removing EP from EPNet (-Net) is detrimental for model performance, supporting our hypotheses. Furthermore, in Figure 3.6, we show that the improvements of EP remain consistent even in high shot settings. Additional results for few-shot SSL and Imagenet-FS [61] and ablations from [164] can be found in the Appendix.

3.4 Discussion

Our exploration of the embedding propagation (EP) procedure has confirmed its efficacy in improving few-shot learning performance. The primary proposition suggests that EP contributes to the smoothening of the class embedding manifold, thereby functioning as a regularizer. This aligns with existing literature which highlights the necessity of smooth class embedding manifolds for semi-supervised learning [27], and in bolstering adversarial robustness [201]. Through our work, we have offered further quantitative and qualitative evidence to substantiate the idea that EP leads to a smoother classification surface, as gauged by the Laplacian measure. Furthermore, our research builds upon the study by [164], elucidating that EP’s benefits are not confined to few-shot classification, but extend to enhancing adversarial robustness and the performance of self/semi-supervised learning.

4 A principled Benchmark for Visual Counterfactual Explainers

Explainability methods have been widely used to provide insight into the decisions made by statistical models, thus facilitating their adoption in various domains within the industry. Counterfactual explanation methods aim to improve our understanding of a model by perturbing samples in a way that would alter its response in an unexpected manner. This information is helpful for users and for machine learning practitioners to understand and improve their models. Given the value provided by counterfactual explanations, there is a growing interest in the research community to investigate and propose new methods. However, we identify two issues that could hinder the progress in this field. (1) Existing metrics do not accurately reflect the value of an explainability method for the users. (2) Comparisons between methods are usually performed with datasets like CelebA, where images are annotated with attributes that do not fully describe them and with subjective attributes such as “Attractive”. In this work, we address these problems by proposing an evaluation method with a principled metric to evaluate and compare different counterfactual explanation methods. The evaluation is based on a synthetic dataset where images are fully described by their annotated attributes. As a result, we are able to perform a fair comparison of multiple explainability methods in the recent literature, obtaining insights about their performance.

4.1 Related Work

Explainability methods. Since most successful machine learning models are uninterpretable [66, 79, 101], modern explainability methods have emerged to provide explanations for these types of models, which are known as post-hoc methods. An important approach to post-hoc explanations is to establish feature importance for a given prediction. These methods [3, 60, 160, 176] involve locally approximating the machine learning model being explained with a simpler interpretable model. However, the usage of proxy models hinders the truthfulness of the explanations. Another explainability technique is visualizing the factors that influenced a model’s decision through heatmaps [44, 47, 234].

Heatmaps are useful to understand which objects present in the image have contributed to a classification. However, heatmaps do not show *how* areas of the image should be changed and they cannot explain factors that are not spatially localized (e.g., size, color, brightness, etc).

Explanation through examples or counterfactual explanations addresses these limitations by synthesizing alternative inputs (counterfactuals) where a small set of attributes is changed resulting in a different classification. These counterfactuals are usually created using generative models. A set of methods condition the generative model on attributes annotated in the dataset by using a conditional Generative Adversarial Network (GAN) [82, 112, 175, 197, 222]. However, this approach restricts the explanations to the provided attributes which do not reflect the entirety of the image properties, making the applicability of these methods challenging where annotations are scarce. In order to generate counterfactuals without recurring to annotated attributes, another set of methods uses VAEs or unconditional GANs [56] that do not depend on attributes during generation [38, 126, 143, 145, 161]. See Table 4.1 for a comparison of the methods considered in our work.

Explainability Benchmarks. DiVE [161] and DiCE [126] propose metrics that allow researchers to evaluate the quality of an explanation. These metrics evaluate the proximity of explanations to their original sample, and how diverse these are. Unfortunately, they are easy to game. For example, an explainer could maximize diversity by always modifying the same counterfactual attribute but randomly perturbing other non-counterfactual attributes to produce new redundant explanations. We propose a more general, harder to game metric that allows us to evaluate a set of explainers in order to identify their strengths and weaknesses through fair comparisons. Further, the set of attributes of a dataset can influence the evaluation of the explainability methods. CelebA [115] is a common dataset used for generating counterfactual explanations [38, 161], and it is labeled with a series of attributes, such as "Attractive", that fail to fully describe the true underlying factors that generated the images (e.g, illumination, occlusions, contrast, etc). Likewise, there is no guarantee that unsupervised disentanglement methods such as VAEs identify the true factors of variations without making strong assumptions [2]. We sidestep these problems by evaluating all explainers in a common latent space with known attributes that fully describe the samples. Recently [142] published a benchmark (CARLA) with an extensive comparison of several counterfactual explanation methods across 3 different tabular datasets. Our work differs from CARLA in three important ways: (1) we propose a principled metric to compare counterfactual explanation methods, (2) we introduce a new synthetic benchmark that allows comparing multiple explainers in a fair manner in the same latent space. (3) We

Table 4.1: Comparison of explainers considered in this work. First column indicates whether counterfactuals are found with gradient descent. Second column indicates whether the explainer takes into account changes in pixel space during optimization (e.g., visual similarity loss) Last column indicates if the explainer performs feature selection to generate counterfactuals.

Method	Gradient based	Optimizes x-space	Feature selection
DiCE [126]	✓	✗	✗
DiVE [161]	✓	✓	✗
GS [99]	✗	✗	✓
StylEx [97]	✗	✗	✓
Latent-CF [5]	✓	✗	✗
xGEM [82]	✓	✓	✗

focus on counterfactual visual explanations, which require access to a common latent space for fair comparison since pixel-level counterfactuals are difficult to interpret (e.g., adversarial attacks).

4.2 Methodology

In the following lines we describe a principled framework to quantify the quality of counterfactual explainers and show how it can be applied to compare multiple methods in the literature. In Section 4.2.1 we define the data generation process, in Section 4.2.2 we define the counterfactual generation process, in Section 4.2.3 we define the concept of optimal classifier used to compare the predictions of a model, and in Section 4.2.4 we define the metric used to evaluate counterfactual explanation methods.

4.2.1 Data generation

Many explainability methods in the literature are designed for the image domain [26, 82, 97, 161, 178]. In this area, most datasets can be described with a data generating process where a set of latent variables (\mathbf{z}) result in an image (x) and a corresponding label (y), see Figure 4.1a. However, not all the latents that generate the image have an impact on the label (\mathbf{z}_{ind}). For example, the image brightness does not affect the presence of a dog. In addition, some latents can be correlated with the label (\mathbf{z}_{corr}). For instance, whenever there is a dog there is usually a dog collar. Formally, we consider a data generating process where a set of latent variables $\mathbf{z} \in \mathbb{R}^d$ are sampled from a prior $p(\mathbf{z})$, and a

generator that produces images $p(x|\mathbf{z})$. Labels are generated using $p(y|\mathbf{z}_{\text{causal}})$, where $\mathbf{z}_{\text{causal}}$ is a subset of \mathbf{z} containing direct causal parents of y (Figure 4.1a). We also define \mathbf{z}_{corr} as the set of attributes that are correlated to y but not part of $\mathbf{z}_{\text{causal}}$.¹ Sometimes, these correlated attributes may have stronger predictive power, but relying on them would lead to unreliable predictions. For instance using the sky background for classifying airplanes. To generate datasets, we rely on a structural causal model (SCM) [144], corresponding to a sequence of stochastic equations producing random variables based on the causal parents in the causal graph as described in Figure 4.1a.

In order to obtain a known mapping between \mathbf{z} and x , we propose to leverage symbols [94], a synthetic dataset generator with many controllable attributes (font, character, color, rotation, size, *etc*). In addition, using a synthetic dataset allows us to control the effect of \mathbf{z} on x and specify the amount of change in x relative to the amount of change in \mathbf{z} (and vice-versa). Using symbols, we train an image generator $x = g(\mathbf{z})^2$, which is used to generate subsequent datasets. In summary, given an image x we task an encoder q with predicting the attributes that describe the image (\mathbf{z}). We also train a generator g to reconstruct x from \mathbf{z} . Finally, we leverage g to generate new datasets. The generator g is provided to the explainers to offer a differential mapping from \mathbf{z} to x . We believe this is a strength of our benchmark compared to using datasets of natural images, since it allows for unambiguous generation of symbols due to the unique attribute space for sampling.

4.2.2 Counterfactual generation

Given an image x and a classifier $\hat{f}(x)$, a counterfactual explanation method (explainer) produces x' , a perturbed version of x that shows some insight about the sensitivity of \hat{f} to the semantic attributes that describe x . The perturbation is commonly performed on a learned latent space \mathbf{z} . In general, explainers are tasked to learn an encoder and find a useful latent space, but this task is hard and still under active research. In order to bring a better comparison between explainers, we provide them access to the generating function g and \mathbf{z} so that explanations are generated in the same latent space. This gives us the opportunity to let explainers work directly in latent space by defining

¹ $z \in \mathbf{z}_{\text{corr}}$ could be correlated to y for two different reasons: i) $y \rightarrow z$ ii) a confounder α such that $y \leftarrow \alpha \rightarrow z$. Note that α may be element of $\mathbf{z}_{\text{causal}}$ or outside of the scene, such as the photograph.

²In this work, we consider deterministic generators. A more general formulation would be $g(x|\mathbf{z})$

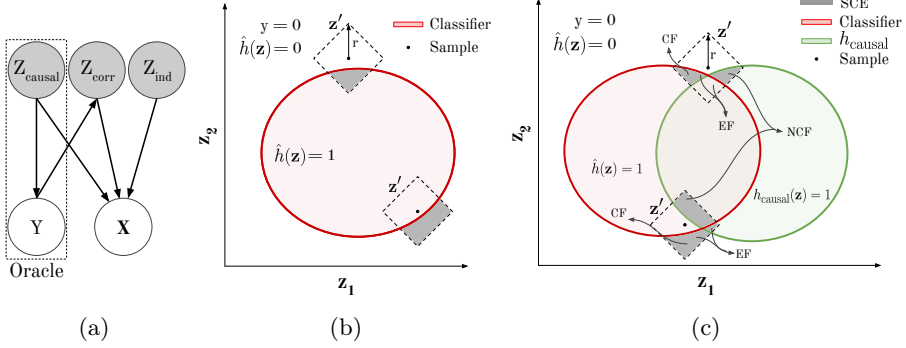


Figure 4.1: (a) Example of a causal graph satisfying the problem setup of section 4.2.1. (b) Successful counterfactual explanation as defined in [82, 126]. That is, a successful counterfactual changes (gray) the classifier prediction (red) for the sample (point). The dashed square represents the maximum L1 norm of the perturbation performed by an explainer (c) Our definition of successful counterfactual explanation (gray) considers any change where an oracle (green) behaves differently from the classifier (red). EF: estimator flips, NCF: non-causal flips, CF: causal flips.

$\hat{h}(\mathbf{z}) := \hat{f}(g(\mathbf{z}))$. In other words, we define an explainer as:

$$\{\mathbf{z}'_i\}_{i=1}^n = e(\mathbf{z}, \hat{h}, g), \quad (4.1)$$

where \mathbf{z}'_i is the i th counterfactual explanation from \mathbf{z} found by explainer e on the latent classifier \hat{h} . Working in latent spaces greatly simplifies the task of an explainer, but we will see that there are still a variety of challenges to be addressed. Namely, the notion of *optimal* classifier or *stable* classifier may be ill-defined or may not always exist.

4.2.3 Optimal Classifier

Counterfactual explanation methods tend to produce trivial explanations by perturbing the attribute being classified from the input [161]. It is likely that an explainer that changes the model’s predictions by perturbing non-causal attributes (such as the background of an image) is more informative when it comes to exposing unwanted biases of the model. To distinguish between these two kinds of explanations, an “oracle” is required, whose predictions are

contrasted with those of the model. If an explanation changes both the oracle and the model’s predictions, the explanation is deemed trivial and discarded. However, if the explanation only changes one of the two, the explanation is non-trivial. In the absence of a human oracle who knows the causal attribute being classified, the authors resort to an optimal predictor or *ground truth* classifier. However, the concept of optimal predictor tends to be ill-defined and varies with its application, hence while assuming the existence of a ground truth classifier, we must proceed cautiously. To show that, we next define the concepts of Bayes classifier, causal classifier, and finally the causal classifier with non-reversible generator used in this work.

Causal classifier with reversible generator. The causal classifier makes predictions solely based on the causal parents of y in the causal graph G . In latent space: $h_{\text{causal}}(\mathbf{z}) = \arg \max_y p(y|\mathbf{z}_{\text{causal}})$. When the generator $x = g(\mathbf{z}, \epsilon_x)$ is reversible, we obtain $f_{\text{causal}}(x) = h_{\text{causal}}(g^{-1}(x))$. Interestingly, this classifier is robust to changes of $p(\mathbf{z})$ as long as $p(y|\mathbf{z}_{\text{causal}})$ and $p(x|\mathbf{z})$ remain unchanged.

Causal classifier with non-reversible generator. It is worth noting that when the generator is not reversible, a given x can lead to many \mathbf{z} , which prevents from directly recovering \mathbf{z} from x . A natural choice is to rely on the posterior distribution $f(x) = \sum_{\mathbf{z}} p(\mathbf{z}|x)h_{\text{causal}}(\mathbf{z})$, where $p(\mathbf{z}|x) \propto p(\mathbf{z})p(x|\mathbf{z})$. However, this posterior now depends on $p(\mathbf{z})$, making the new classifier no longer independent to distribution shift when $p(\mathbf{z})$ is changed to e.g. $p'(\mathbf{z})$. This leads to the following negative result:

Proposition 1 *1 If there exists a pair \mathbf{z}, \mathbf{z}' s.t. $g(\mathbf{z}) = g(\mathbf{z}')$ and $h_{\text{causal}}(\mathbf{z}) \neq h_{\text{causal}}(\mathbf{z}')$, then for any deterministic classifier $\hat{f}(x)$, there is a prior $p'(\mathbf{z})$ s.t. the accuracy of \hat{f} is 0 with respect to h_{causal} .*

This shows that since the concept of optimal predictor is commonly ill-defined and application-dependent, we must proceed with care when assuming the existence of a ground truth classifier.

4.2.4 Evaluating Counterfactual Explanations

The goal for counterfactual generation methods is to find all the attributes that make a classifier behave differently from a causal classifier (see Figure 4.1c). Note that [126] only considered counterfactuals that change the predictions of a classifier (Figure 4.1b), and [161] only considered the top region in Figure 4.1c. These definitions do not cover cases such as when the oracle changes its prediction while the classifier’s stay the same. Following [82, 126, 161], we also measure the similarity between the original example and the counterfactuals

used to explain it. The reason is that counterfactuals should be relatable to original samples so that a human can interpret what is the sensitive semantic attribute. Next, we define the components of the proposed metric (Eq. 4.7).

Proximal change [82, 126]. An explanation must be relatable to the original sample, thus it needs to be proximal. That is, the change \mathbf{z}' needs to stay within a certain radius r from \mathbf{z} . Using L1 norm, the set of proximal \mathbf{z}' is defined as follows:

$$P_r(\mathbf{z}) = \{\mathbf{z}' \mid \|\mathbf{z} - \mathbf{z}'\|_1 \leq r\} \quad (4.2)$$

Estimator Flip (EF) [82, 126]. This is defined as a proximal change on \mathbf{z} leading to a change in prediction of the estimator \hat{h} (see Figure 4.1b).

$$\text{EF}(\mathbf{z}) = \left\{ \mathbf{z}' \mid \hat{h}(\mathbf{z}') \neq \hat{h}(\mathbf{z}) \right\} \cap P_r. \quad (4.3)$$

Non-Causal Flip (NCF). Counterfactuals obtained by estimator flips (EF) are common in the literature as they do not require the knowledge of h_{causal} . However, if we have access to h_{causal} , we can detect a new set of explanations: a proximal change in \mathbf{z}' that changes the prediction of \hat{h} but not of h_{causal} :

$$\text{NCF}(\mathbf{z}) = \left\{ \mathbf{z}' \mid \text{EF}(\mathbf{z}) \wedge h_{\text{causal}}(\mathbf{z}') = h_{\text{causal}}(\mathbf{z}) \right\} \cap P_r. \quad (4.4)$$

Causal Flip (CF). Additionally, access to h_{causal} allows us to detect another new set of explanations: a proximal change in \mathbf{z}' that changes the prediction of h_{causal} but not \hat{h} :

$$\text{CF}(\mathbf{z}) = \left\{ \mathbf{z}' \mid \hat{h}(\mathbf{z}') = \hat{h}(\mathbf{z}) \wedge h_{\text{causal}}(\mathbf{z}') \neq h_{\text{causal}}(\mathbf{z}) \right\} \cap P_r. \quad (4.5)$$

Thus, we define the set of successful counterfactual explanation (SCE) as follows:

$$\text{SCE}(\mathbf{z}) = (\text{NCF} \cup \text{CF}). \quad (4.6)$$

In summary, having knowledge of the causal factors (access to h_{causal}) allows us to evaluate counterfactual explanations in a new way as illustrated in the following example. Given a dog classifier and an image of a dog, a counterfactual example that changes the background of the image in a way that alters the classifier’s prediction (NCF) will almost certainly provide valuable insight about the model’s behaviour. The same can be said about a counterfactual example that removes the dog from the image without altering the classifier’s prediction

(CF) (see Figure 4.1c). Note that these counterfactuals cannot be detected without causal knowledge, which is only available if we have access to the entire data generating process *i.e.*, a synthetic dataset.

Orthogonal and complement subset. Note that both EF and SCE are possibly infinite sets and cannot be easily interpreted by humans. We could return the explanation minimizing some notion of distance on \mathbf{z} or x , however a good explainer should return a useful and diverse set of explanations.

To this end, we propose a metric that only takes into account the subset of orthogonal and complementary explanations. Otherwise, it is trivial to report many *explanations* that are a modification of an existing explanation without being useful. For instance, modifying the hair color to trigger a change in gender classification is a good finding, but changing the hair color again, and removing some clouds in the sky would not constitute a useful explanation. Hence, only admitting orthogonal explanations enforces a useful diversity. However, we also admit complementary explanations. That is, if darker hair triggers a change in gender classification and lighter hair also triggers a change, these are two useful explanations. In short, given two explainers that find counterfactuals by perturbing the most sensitive attribute, the orthogonality and complementary requirements ensure that the one that provides a more diverse set of counterfactuals by also perturbing less sensitive attributes scores higher. This is important because it rewards explainers that give a more complete description of the model to the user. There may be use cases where only the most sensitive attribute matters. However, everything else being equal, we argue that, in general, it is favorable to have access to a diversity of explanations.

The explainer is responsible for returning explanations produced with orthogonal or complementary perturbation vectors. To verify whether explanations are orthogonal or complementary we use a greedy algorithm³. Concretely, we sort the explanations by how proximal they are to the original sample and add the first one to the set. Then we iterate through the rest and sequentially add every subsequent explanation that is orthogonal or complementary to all the explanations currently in the set (see Algorithm 1 for implementation). The resulting orthogonal and complement set is referred to as $SCE_{\perp}(\mathbf{z})$. We use the cardinality of this set to evaluate the performance of explainers:

$$\mathcal{S}_{\#} = |SCE_{\perp}(\mathbf{z})|. \tag{4.7}$$

We consider the proposed setup to be fairer than previous works, since: (1) all explainers are compared in the same latent space, resulting in a fair

³The complexity of the resulting algorithm for a given number of explanations n is $\mathcal{O}(n^3)$

Algorithm 1 Orthogonal Set

Input: original sample $z \in \mathbb{R}^d$, successful counterfactuals $e_{sc} \in \mathbb{R}^{n \times d}$, threshold τ

Output: an orthogonal set of counterfactuals

```

 $\Delta_{sc} \leftarrow e_{sc} - z$ ; // calculate perturbation vector
indices  $\leftarrow \text{argsort}(\|\Delta_{sc}\|_1)$ ; // sort perturbations by increasing norm
 $\Delta_{orth} \leftarrow \Delta_{sc}[\text{indices}[0]]$ ; // initialize set of orthogonal perturbations
for  $i = 1$  to  $n$  do
   $p \leftarrow \Delta_{sc}[\text{indices}[i]]$ ; // select the next perturbation
   $sim \leftarrow \cos(p, \Delta_{orth})$ ; // calculate similarity of  $p$  with all elements in the set
  if  $(\forall j \in \text{abs}(sim_j) < \tau)$  or  $(\exists j \in sim_j + 1 < \tau)$  then
    |  $\Delta_{orth} \leftarrow [\Delta_{orth}; p]$ ; // add perturbation to the set
  end
end
return  $z + \Delta_{orth}$ ; // return set of orthogonal counterfactuals

```

evaluation, (2) uninformative explanations are discarded leveraging knowledge of the causal factors, (3) it is designed to be more difficult to game by repeating counterfactual explanations, and (4) it rewards explainers that return a more complete set of explanations.

4.3 Experiments and Results

In this section we give an overview of the different methods (see Table 4.1) and datasets that are comprised within our benchmark. Since we provide access to a common interpretable latent space, we evaluate explainers that do not depend on a concrete latent decomposition. The code is written in PyTorch [140] and is made public along with the datasets and pretrained weights for the models used in this work.

Latent-CF [5]: A simple method that performs adversarial perturbations in the latent space until a counterfactual with confidence higher than threshold tol is found.

DiCE [126]: A method that aims to produce a diverse set of counterfactual examples directly from a series of attributes or latent space by proposing a series of perturbations that change the predictions of a classifier. This is achieved by gradient-based optimization of multiple loss functions with respect to the

attributes or latents and the classifier:

$$\mathcal{L} = \underbrace{\text{hinge_loss}(\hat{h}(\mathbf{z}'), y, \text{margin})}_{(A)} + \underbrace{\lambda_1 \text{dist}(\mathbf{z}, \mathbf{z}')}_{(B)} + \underbrace{\lambda_2 \text{dpp_diversity}(\mathbf{z}')}_{(C)}, \quad (4.8)$$

where optimizing (A) pushes the prediction of the classifier \hat{f} towards y up to some margin, (B) ensures that counterfactuals (\mathbf{z}') are close to the original samples (\mathbf{z}), and (C) maximizes the distance between each pair of counterfactuals.

xGEM [82]: A method equivalent to DiCE without the diversity term (C).

DiVE [161]: A method similar to DiCE that leverages the Fisher Information (FI) to find non-trivial counterfactuals, *i.e.* samples that change the classifier prediction without changing the causal attribute $\mathbf{z}_{\text{causal}}$, thus focusing on spurious correlations. This is done by masking out latent dimensions with the highest FI while optimizing a cost equivalent to Eq. 4.8.

Growing Spheres (GS) [99]: A method that given a data point \mathbf{z} identifies its closest neighbour classified differently \mathbf{e} referred to as *enemy*. This is done by finding the smallest l_2 -ball around \mathbf{z} that contains an *enemy*. Once \mathbf{e} is found the dimensions with small changes in \mathbf{e} with respect to \mathbf{z} are discarded through a feature selection process, maximizing the sparsity of $\mathbf{e} - \mathbf{z}$.

StyleEx [97]⁴: They find a latent perturbation in a direction that maximizes the difference in the output of the classifier for the original sample and its perturbed counterpart.

Informed Search (IS): An explainer that knows about the data generation process in Figure 4.1a. Thus, IS generates explanations by perturbing the spuriously correlated attributes \mathbf{z}_{corr} .

4.3.1 Datasets

We design a synthetic benchmark based on the symbols dataset [94]. In this benchmarks images are fully defined by 3 categorical attributes (48 fonts, 48 characters, 2 background colors) and 4 continuous attributes (x-translation, y-translation, rotation, scale), see Figure 4.2.

An advantage of symbols is the large amount of values in its categorical attributes such as character and font. This allows us to design different scenarios by introducing spurious correlations based on subsets of these attributes. From now on, we assume $\mathbf{z}_{\text{causal}} = \text{char} \in [1..48]$ and set $h_{\text{causal}} = \mathbf{z}_{\text{causal}} \bmod 2$, creating a binary classification problem. Then we leverage the font attribute to

⁴Since we already provide an interpretable set of latent attributes we evaluate only the Attribute finding (AttFind) algorithm from the paper

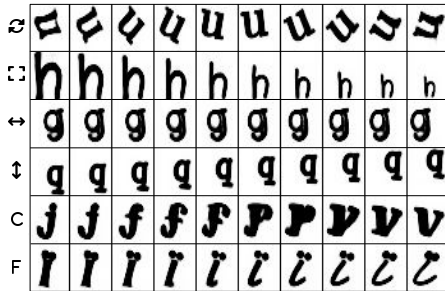


Figure 4.2: Interpolations produced by our learned generator ($g(\mathbf{z})$). Images are changed as the attribute’s value changes smoothly from one value to another (left-right). From top to bottom: rotation, scale, h-translation, v-translation, char, font.

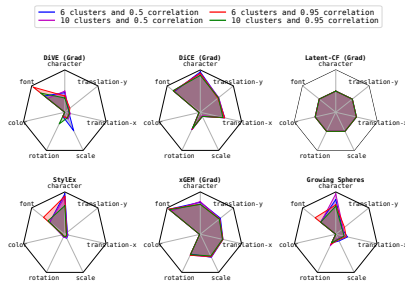


Figure 4.3: **Average attribute perturbation** for each method/scenario. Gradient based methods perturb almost all attributes while gradient-agnostic methods perturb only one or two. DiVE focuses almost solely on font.

introduce spurious correlations (\mathbf{z}_{CORR} in Figure 4.1a). Note that increasing the number of fonts in \mathbf{z}_{CORR} (the rest will be in \mathbf{z}_{IND}) increases the random chance of finding a counterfactual by accidentally switching the font. Likewise, increasing the amount of correlation between \mathbf{z}_{CORR} and y makes spurious correlations easier to find since the classifier latches stronger on them. We hypothesize that stronger correlations will benefit gradient-based explainers, which will find higher gradient curvature for highly correlated fonts. To explore how explainers behave under different scenarios, we consider 6 and 10 spurious fonts with 50% and 95% correlation with y , resulting in a total of 4 scenarios. Further, we introduce a 5% of noise to the $\mathbf{z}_{\text{CAUSAL}}$ attribute (character) to encourage classifiers to also rely on the font.

4.3.2 Metric Evaluation

We evaluate the performance of six different methods with the metric defined in Eq. 4.7. Each method is evaluated in several different datasets with varying levels of difficulty as described in 4.2.1 and 4.3.1.

It is hard to diversify. A good explainer should be able to predict the behavior of the model with respect to changes in the different attributes that generate the data. In the case of a classifier, finding the attributes that induce it to change its prediction in order to reveal if it is relying on attributes that are

Table 4.2: Score (Eq. 4.7) and percentage of trivial counterfactuals () obtained by each explainer for each of the different datasets described in 4.3.1. Values represent an average score across batches for the entire dataset for 3 different runs.

		Correlation	0.50	0.95	0.50	0.95
#Spurious	Explainer	$\mathcal{S}_\#$	$\mathcal{S}_\#$	Trivial (%)	Trivial (%)	
6	IS (Oracle) 4.3	2.40 ± 0.30	2.67 ± 0.20	0.00 ± 0.00	0.00 ± 0.00	
	DiCE [126]	1.18 ± 0.01	1.17 ± 0.01	9.37 ± 0.11	6.78 ± 0.26	
	DiVE [161]	1.02 ± 0.00	1.00 ± 0.01	2.51 ± 0.09	1.68 ± 0.02	
	GS [99]	1.01 ± 0.00	1.01 ± 0.00	4.49 ± 0.40	2.34 ± 0.15	
	StylEx [97]	1.04 ± 0.00	1.17 ± 0.00	2.41 ± 0.00	1.58 ± 0.00	
	Latent-CF [5]	0.82 ± 0.00	0.93 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	
	xGEM [82]	1.18 ± 0.02	1.15 ± 0.01	12.46 ± 0.03	6.45 ± 0.07	
10	IS (Oracle) 4.3	2.80 ± 0.40	3.63 ± 0.20	0.00 ± 0.00	0.00 ± 0.00	
	DiCE [126]	1.13 ± 0.01	1.19 ± 0.01	8.62 ± 0.46	6.70 ± 0.08	
	DiVE [161]	1.00 ± 0.00	1.04 ± 0.00	2.28 ± 0.03	1.44 ± 0.04	
	GS [99]	1.01 ± 0.00	1.00 ± 0.01	4.91 ± 0.25	1.95 ± 0.08	
	StylEx [97]	1.15 ± 0.00	1.12 ± 0.00	3.37 ± 0.00	1.62 ± 0.00	
	Latent-CF [5]	0.81 ± 0.00	0.81 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	
	xGEM [82]	1.15 ± 0.00	1.16 ± 0.00	10.17 ± 0.28	6.38 ± 0.11	

independent from the class being predicted is a desirable goal. In the pursuit of this goal, an explainer should ideally populate $\text{SCE}(\mathbf{z})$ (see Eq. 4.6) with explanations altering each of the attributes that are correlated with the data. However, as shown in Table 4.2 explainers fail to consistently find more than one altering attribute.

Performance saturates with 6 fonts. We observe that methods do not significantly increase the number of successful counterfactuals when adding 4 more spurious fonts (Table 4.2). This is, partially, because methods tend to focus on changing the $\mathbf{z}_{\text{causal}}$ attribute character as seen in Figure 4.3, which leads to trivial counterfactuals. When adding more fonts, the font identification task becomes more difficult for the classifier, which makes it more sensitive to characters and exacerbates this problem. For a more extensive ablation illustrating this phenomenon see Figure 4.4.

Gradients tend to perturb most of the attributes. Figure 4.3 offers insight into how each method perturbs \mathbf{z} and we can see that gradient-based methods tend to perturb almost all attributes equally, exploring the perturbation space in many directions. In the extreme, we found that Latent-CF slightly modifies all the latent attributes, producing counterfactuals that resemble adversarial attacks. While modifying all the attributes increases the chances of finding 1 good explanation on average, it also prevents the explainer

Table 4.3: From left to right: We report the percentage of estimator flips EF (Eq. 4.3) [82, 126], percentage of successful counterfactuals SCE (Eq. 4.6) and what percentage of those are Non-Causal Flips (Eq. 4.4) and Causal Flips (Eq. 4.5). Values represent an average score across batches for the entire dataset for 3 different runs.

Correlation		0.50		0.95		0.50		0.95		0.50		0.95	
#Spurious	Explainer	EF (%)		SCE (%)		Causal Flip Rate (%)		Non-Causal Flip Rate (%)					
6	IS (Oracle)	40.55 \pm 0.16	76.97 \pm 0.24	67.5 \pm 4.40	62.25 \pm 4.45	0.00 \pm 0.00	0.00 \pm 0.00	100 \pm 0.00	100 \pm 0.00				
	DiCE [126]	32.90 \pm 0.05	31.96 \pm 0.11	55.42 \pm 2.60	56.67 \pm 2.88	26.46 \pm 1.56	29.22 \pm 1.20	73.54 \pm 1.56	70.78 \pm 1.20				
	DiVE [161]	25.02 \pm 0.38	36.58 \pm 0.12	67.5 \pm 1.25	60.83 \pm 2.60	6.55 \pm 0.28	3.44 \pm 0.15	93.45 \pm 0.28	96.56 \pm 0.15				
	GS [99]	36.14 \pm 1.22	34.94 \pm 0.49	31.67 \pm 7.10	31.67 \pm 15.63	0.00 \pm 0.00	0.00 \pm 0.00	100 \pm 0.00	100 \pm 0.00				
	Stylex [97]	23.66 \pm 0.00	24.84 \pm 0.00	23.07 \pm 0.00	25.80 \pm 0.00	17.02 \pm 0.00	21.40 \pm 0.00	82.98 \pm 0.00	78.60 \pm 0.00				
	Latent-CF [5]	20.98 \pm 0.00	24.18 \pm 0.00	20.96 \pm 0.00	24.18 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	100 \pm 0.00	100 \pm 0.00				
	xGEM [82]	70.61 \pm 0.28	75.98 \pm 0.25	76.67 \pm 9.21	78.33 \pm 1.90	4.18 \pm 0.69	2.96 \pm 0.22	95.82 \pm 0.00	97.04 \pm 0.00				
	10	IS (Oracle)	35.33 \pm 0.16	71.19 \pm 0.08	54.50 \pm 2.43	51.25 \pm 1.25	0.00 \pm 0.00	0.00 \pm 0.00	100 \pm 0.00	100 \pm 0.00			
DiCE [126]		31.77 \pm 0.45	33.25 \pm 0.23	45.00 \pm 4.33	39.17 \pm 0.72	26.16 \pm 3.43	27.65 \pm 0.79	73.84 \pm 3.43	72.35 \pm 0.79				
DiVE [161]		22.39 \pm 0.31	31.8 \pm 0.25	60.00 \pm 1.25	54.58 \pm 0.72	6.83 \pm 0.57	2.25 \pm 0.01	93.17 \pm 0.57	97.75 \pm 0.01				
GS [99]		37.38 \pm 0.38	36.38 \pm 0.54	44.58 \pm 12.52	40.42 \pm 5.90	0.00 \pm 0.00	0.00 \pm 0.00	100 \pm 0.00	100 \pm 0.00				
Stylex [97]		24.20 \pm 0.00	23.04 \pm 0.00	23.65 \pm 0.00	23.77 \pm 0.00	21.60 \pm 0.00	20.35 \pm 0.00	78.40 \pm 0.00	79.65 \pm 0.00				
Latent-CF [5]		22.00 \pm 0.00	23.33 \pm 0.00	21.98 \pm 0.00	23.24 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	100 \pm 0.00	100 \pm 0.00				
xGEM [82]		66.45 \pm 0.63	74.95 \pm 0.30	61.67 \pm 5.20	62.92 \pm 5.90	4.51 \pm 0.35	2.15 \pm 0.22	95.49 \pm 0.00	97.85 \pm 0.27				

from finding multiple non-trivial diverse explanations. On the other hand, methods that are gradient-agnostic focus on perturbing one or two attributes, resulting in a more narrow search space. This increases the risk of methods focusing on $\mathbf{z}_{\text{causal}}$ (Figure 4.1a). This is evidenced in Figure 4.3, where Stylex and GS considerably perturb the character attribute. Interestingly, the perturbation pattern of DiVE shares some similarities with Stylex and GS due to gradient masking.

DiVE focuses on changing the font. As shown in Figure 4.3, DiVE perturbs almost *exclusively* the \mathbf{z}_{corr} attribute (font), specially for high correlation values, this indicates that the method successfully distinguishes between $\mathbf{z}_{\text{causal}}$ and \mathbf{z}_{corr} attributes. However, it is not able to consistently perturb the font in the right way to produce a diverse set of counterfactuals as evidenced by its score (Table 4.2).

Non-triviality is not enough. Table 4.2 (right) shows the average percentage of trivial counterfactuals found by each method. We observe that methods that tend to produce a higher number of successful explanations (left) tend to also produce a larger number of trivial counterfactuals (right), which are discarded in our metric.

Quality over quantity As seen in Table 4.3 some explainers obtain a high percentage of successful counterfactuals SCE, sometimes even higher than the oracle (xGEM, DiVE). However, this is not reflected in their score $\mathcal{S}_{\#}$ (Table 4.2), which is considerably lower than the oracle’s. This is because

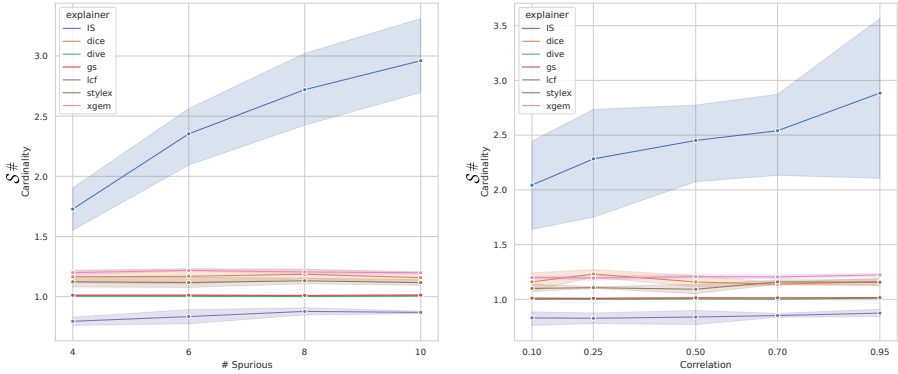


Figure 4.4: Sensitivity of every explainer to varying amount of correlation levels (left) and number of spuriously correlated attributes (right) (see Section 4.3.1) measured with our score (Eq. 4.7). Note how the performance of the explainers, excluding the oracle (IS), does not scale and is not sensitive to the level of correlation and the amount of correlated variables. This supports our findings that explainers are unable to provide a diverse set of explanations and focus on changing the causal attribute $\mathbf{z}_{\text{causal}}$.

even though the explainers can find a high number of counterfactuals they are discarded by our metric since they are not orthogonal or complementary and thus redundant. Further, note that the score measured using estimator flips EF (Eq. 4.3) [82, 126] is not correlated with our score $S_{\#}$ (Table 4.2). For example, xGEM obtains a higher score than the oracle (IS) despite the latter returning a more complete set of explanations. This shows how previously proposed metrics [82, 126] can be gamed by explainers by generating many redundant explanations that fail to fully describe the model’s behaviour. Figure 4.4, also supports this finding, showing how the performance of the oracle (IS) is the only one affected by the amount of spuriously correlated attributes and their level of correlation (see Section 4.3.1).

Explainers exploit bad classifiers. As seen in Table 4.2 and in Figure 4.4 explainers are not significantly affected by the amount of spurious correlation \mathbf{z}_{corr} introduced. This indicates that, in contrast with the oracle (IS), methods produce explanations by changing the font attribute $\mathbf{z}_{\text{causal}}$ (as seen in Figure 4.3) without changing the classifier’s prediction, thus creating a successful counterfactual (Eq. 4.5). These counterfactuals expose failure cases of the classifier and are therefore useful, since they show that the classifier

is unable to classify some characters. Table 4.3 shows that DiCE [126] and StyleEx [97] produce a high amount of these counterfactuals, while GS [99] and Latent-CF [5] always change the classifier’s prediction and thus produce none. The oracle (IS) is not designed to perturb $\mathbf{z}_{\text{causal}}$ in any way so it cannot produce any causal counterfactuals.

In Figure 4.5 we show two ways in which explainers obtain causal counterfactuals. Note that besides confusing the classifier by modifying the character’s diacritic, explainers can create new characters entirely by merging two letters together or even adding an accent mark to a consonant. This behaviour is unavoidable in the absence of an optimal classifier.

Additional insights As seen in Table 4.2 explainers are unable to generate a diverse set of counterfactual explanations. However, Table 4.3 highlights some differences between methods when it comes to other metrics. If the objective is to maximize the number of estimator flips EF (Eq. 4.3) or the number of successful counterfactuals SCE (Eq. 4.6) we recommend using xGEM. If the objective is to maximize the number of causal flips (Eq. 4.5) we recommend using StyleEx or DiCE. That said, we have shown that explainers generate a high amount of redundant counterfactuals, and thus we recommend caution when choosing them based on how they maximize these individual metrics.

4.3.3 Limitations

As discussed in Section 4.2.3, the core limitation of explaining image classifiers via latent perturbations is the lack of accurate reversible generators. If the generator is not reversible, a given x can lead to many \mathbf{z} , which prevents the direct recovery of \mathbf{z} from x . It might be a good idea to bypass the pixel space completely and work directly on \mathbf{z} , this however, would produce explanations outside the image domain and therefore, uninterpretable by humans, which is ultimately not very useful. It could be argued that the generator used in this work could be modified to yield better image reconstructions given any \mathbf{z} , however this will always be hindered by the aforementioned limitation. More generally, most methods rely on some sort of latent decomposition in order to search for counterfactuals in a latent space. However, it is still not clear how the true latent variables of the data generating process are not identifiable [116]. In this work we circumvent this problem by using a synthetic dataset. On the other hand, [87] showed that, with further assumptions, it is possible to identify the latent variables [87]. Moreover, in a temporal setup, it is possible to identify which of these latent variables are the causal ones [92]. Finally, in a multi-task setup where distribution shift occurs, it is possible to identify which variables are robust to distributions shift and hence, likely to be the causal

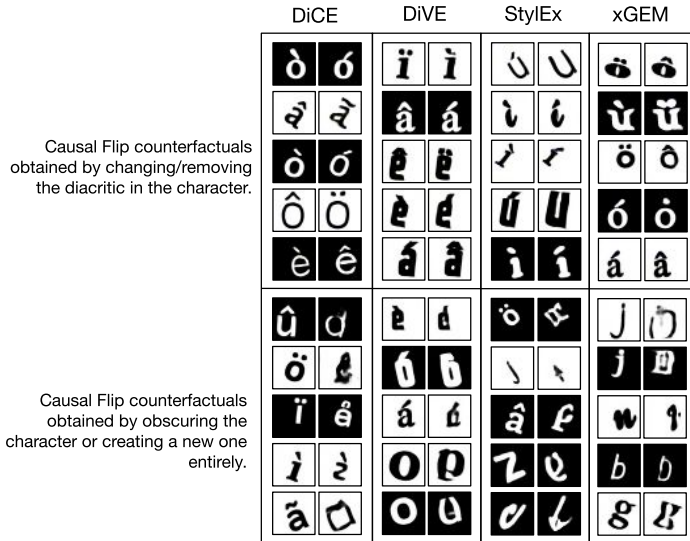


Figure 4.5: Some of the Causal Flip counterfactuals (Eq. 4.6) obtained by each method separated into two different subcategories.

ones. In summary, although it would be possible to approximate the true latent factors in some cases, it would require making additional assumptions about the data.

We make an effort to establish a fair, principled metric that is useful. However, this metric does not depict all the properties of an explainer such as fragility or speed. It is possible, albeit unlikely, that our definition of useful/informative explanation might not always align with that of the user. For example, it is possible in some cases for trivial explanations to be informative, however trivial explanations are easily obtained by explainers, while non-trivial ones are elusive. Thus, we focus on the latter. It is important to note that our definition of SCE (Eq. 4.6) fits the context of image classifiers best. To evaluate explainers in other domains (*i.e.*, algorithmic recourse [85]) a more flexible definition should be adopted. Lastly, the generator we use in this work can generate images with certain implicit biases.

4.4 Discussion

Benchmark In this work, we have introduced a more comprehensive definition of good counterfactual (Section 4.2) that we instantiate as a metric (Section 4.2.4) as well as a fair evaluation setup (Section 4.2.1) in the form of a benchmark. Previous evaluation setups use datasets like CelebA where the causal data generation process is unknown and use metrics that are easy to game. In contrast, our evaluation setup uses a more comprehensive and fair metric while providing control over the entire data generating process and therefore knowledge of the causal factors, providing a tool to evaluate properties of explainers that are impossible to evaluate in a non-synthetic setup. Even though knowledge of causal factors is rare when working in real world scenarios, it is possible to adapt our metric to take only into account an orthogonal and complement set of estimator flips EF (Eq. 4.3) which do not require causal knowledge. However, any evaluation schema that does not include causal information would be incomplete. Further, if an explainer fails to provide a set of useful and diverse explanations for our simple synthetic dataset it is very unlikely that it is able to do so for real datasets. Nevertheless, we recommend users to also evaluate explainers using real world data.

Oracle We argue that successful counterfactuals should be considered in the perspective of a human. In the absence of a human, we must resort to an optimal classifier, whose task is to contrast the predictions of the model with the optimal prediction and spot unexpected behaviors; acting as an oracle. Without an oracle, it is not clear how we could assess whether a model is working as intended. We show that the optimal classifier is commonly ill-defined in the image domain, because it is not always possible to access an invertible image generator (Section 4.2.3). Therefore, it cannot be expected that a classifier trained on pixel space achieves optimal performance.

Results Our experimental results could indicate that the different counterfactual explainers in the literature perform similarly and there has been little improvement in the recent years (Table 4.2). Although most of them find a single explanation in average, we found that they do it in different ways (Figure 4.3). We hope our findings encourage further research on fair evaluation benchmarks.

5 EarthView: A Large Scale Remote Sensing Dataset

This chapter presents EarthView, a comprehensive dataset specifically designed for self-supervision on remote sensing data, intended to enhance deep learning applications on Earth monitoring tasks. The dataset spans 22 tera pixels of global remote-sensing data, combining imagery from a diverse range of sources, including NEON, Sentinel, and a novel release of 1m spatial resolution data from Satellogic. Our dataset provides a wide spectrum of image data with varying resolutions, harnessed from different sensors and organized coherently into accessible hdf5 files. This data spans five years, from 2017 to 2022. Accompanying the dataset, we introduce EarthMAE, a tailored Masked Autoencoder, developed to efficiently tackle the distinct challenges of remote sensing data. Trained in a self-supervised fashion, EarthMAE effectively processes different data modalities such as hyperspectral, multi-spectral, topographical data, segmentation maps, and temporal structure. We regard this innovative combination of an expansive, diverse dataset and a versatile model adapted for self-supervised learning as a significant stride forward in deep learning for Earth monitoring.

5.1 Related Work

5.1.1 Dataset for training

The success of large-scale deep learning models has triggered research on larger datasets that can fit the capacity of current systems. [184] introduced BigEarthNet, a large-scale benchmark archive for remote sensing image understanding. This dataset consists of 590,326 Sentinel-2 image patches, annotated with multiple land-cover classes. The annotations were provided by the CORINE Land Cover database, and the dataset was significantly larger than existing archives in remote sensing. The authors demonstrated that training models on BigEarthNet improved accuracy compared to pre-training on ImageNet, indicating its potential for advancing operational remote sensing applications. [147] addressed the need for multi-label annotated datasets in remote sensing for

semantic scene understanding. They developed MLRSNet, a multi-label high spatial resolution remote sensing dataset containing 109,161 samples within 46 scene categories. Each image in MLRSNet has at least one of 60 predefined labels, enabling training deep learning models for multi-label tasks such as scene classification and image retrieval. The authors highlighted the importance of MLRSNet as a benchmark dataset and its complementary nature to existing datasets like ImageNet. [193] presented the Five-Billion-Pixels dataset, aiming to enable country-scale land cover mapping with meter-resolution satellite imagery. The dataset comprises more than 5 billion labelled pixels from 150 high-resolution Gaofen-2 satellite images. They proposed a deep-learning-based unsupervised domain adaptation approach to transfer classification models trained on labelled data to unlabeled data for large-scale land cover mapping. The experiments demonstrated promising results across different sensors and geographical regions, showcasing the potential of the dataset and proposed approach. In a concurrent work, [7] introduced Satlas, a large-scale dataset for remote sensing image understanding. Satlas is comprehensive in terms of both breadth and scale, containing 302 million labels across 137 categories over a cumulative of 17 trillion pixels. The authors evaluated multiple baselines and a proposed method on Satlas. Pre-training on Satlas significantly improved performance on downstream tasks compared to ImageNet and other baselines.

While the previous benchmarks constitute a significant step in data availability for remote sensing, they are typically limited by the cost of obtaining labels. This has motivated the construction of unlabeled datasets that can leverage uncurated data from many different sources. For example, [122] proposed to leverage unlabeled data with Seasonal Contrast (SeCo). They collected a dataset of Sentinel-2 patches without human supervision, consisting of 1 million multi-spectral image patches from approximately 200,000 locations worldwide. By capturing seasonal changes with images from different dates, they aimed to enhance the training of models for remote sensing tasks. While SeCo focused on uniformly covering most of the inhabited regions of Earth with Sentinel-2 data, [13] focused on densely covering Europe with multiple data sources (Copernicus, Sentinel-2, and Planet) over space and time (500,000 locations in Europe with daily readings for a year). In this work, we combine multiple data sources at different points in time while considering most of the inhabited Earth, resulting in a dataset that we named EarthView. Concretely, EarthView offers a larger and more diverse collection of unlabeled data by combining a high-quality curated selection from multiple data sources (Sentinel, NEON, and Satelloic), achieving a larger scale and variety than previous works (over 22 trillion pixels, with temporal revisits, and from 60 to 0.1m resolution). We share EarthView in a highly accessible format and available

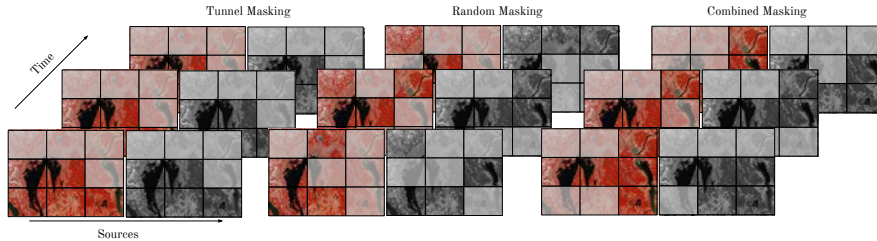


Figure 5.1: Different masking schemas explored in our work. Random masking, masks random patches across sources/time while tube masking masks the same patches. Combined masking combines both of them by first masking some patches consistently across sources/time and then masking a subset of the remaining ones randomly.

through Hugging Face, which enables easy integration into research projects. These qualities make our dataset a valuable resource for exploring uncharted patterns and structures in an unsupervised learning setting.

5.1.2 Learning from unlabelled data

Multi-view self-supervised learning methods have played a crucial role in building large models with remote sensing data [9, 122, 214]. In addition to multi-view SSL, reconstruction-based SSL with MAEs [62] has also been explored in the context of remote sensing. Scale-MAE, proposed by [154], explicitly learns relationships between different scales, resulting in robust multiscale representations. [210] introduced MIM, using masked image modelling for remote sensing scene classification. SatMAE by [36] introduced a pre-training framework leveraging temporal and multi-spectral satellite imagery, encoding groups of bands independently with a spectral positional encoding. SpectralMAE, presented by [236], focused on the reconstruction of arbitrary combinations of bands and data sources. Given the versatility of MAE-based approaches to handling multiple data sources, we choose this model class to experiment with the EarthView dataset introduced in this work. Concretely, we generalize SatMAE and SpectralMAE by combining multiple masking strategies, i.e. we combine masking all bands given a random position in an image with randomly masking individual bands in random positions (see Figure 5.1). In experiments, we find that this strategy is effective for learning from heterogeneous data sources like the proposed EarthView data.

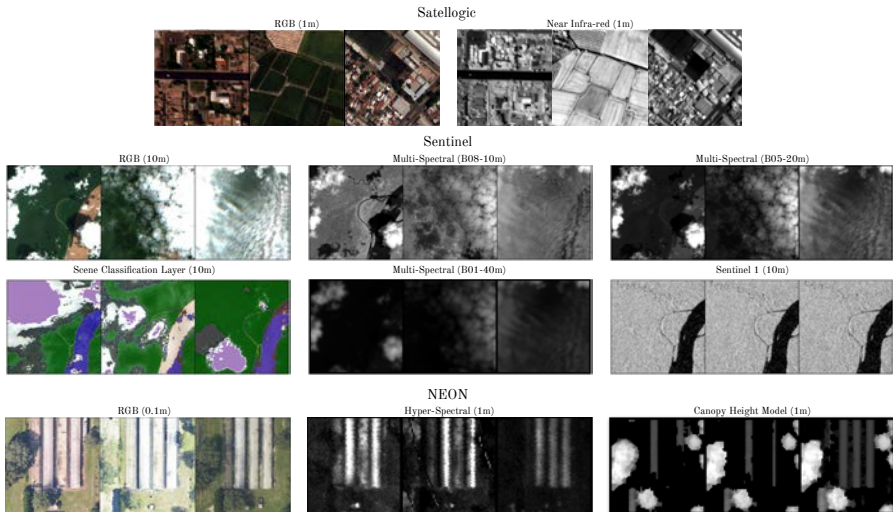


Figure 5.2: Samples from the dataset

5.2 Proposed Dataset

This work introduces an extensive dataset tailored for self-supervision on Earth monitoring data. It is based on the assumption that structure in the data brings an essential signal to self-supervised algorithms for finding high-level semantic representation that can make sense of the data. To this end, we combine spectral, temporal and spatial structures in a large-scale dataset composed of multiple sources and multiple spatial resolutions. The data gathered for this project is drawn from a triad of distinct sources, namely Sentinel, NEON, and Satellogic. Each of these contributes unique facets and dimensionalities to the integrated dataset.

5.2.1 Sentinel

Accessibility While Sentinel data is distributed under a creative commons license, very large datasets are less accessible. Since the Google Earth engine throttles the download speed, it is prohibitively long to download terabytes of data. We thus had to resort to AWS, but since *requester pays* the bandwidth,¹

¹AWS hosts Sentinel data for free, but the user has to pay for the bandwidth usage when downloading.

we required a large budget just for collecting this data.²

Sensors Sentinel’s constellation offers a wide range of sensors. For this project we focus on synthetic aperture radar (SAR) from **Sentinel-1**, and multi-spectral from **Sentinel-2**. The other sensors offer a spatial resolution that is too low for our purpose. For SAR, we use the level-1 Ground Range Detected (GRD) product available in AWS. We stack the different polarizations (VH, VV) resulting in two bands and resample it to 10m resolution. Sentinel-1 images are then saved in uint16 format to reduce their size in bytes. Finally, Sentinel-2 is composed of 13 spectral bands. The main bands, (blue, green, red, near-infrared), have 10m GSD, but due to atmospheric absorption of other wavelengths other bands have 20m and 60m resolution.

Spatial Distribution Our aim is to gather a wide range of regions covering the planet, however, we also want to avoid highly redundant patterns such as ocean, desert and forests. To this end, we gather inspiration from [122] and collect Sentinel-2 tiles that overlap regions within a 50km radius around the top largest cities in the world. Each footprint area (100km x 100km) is large enough to cover coastal, agricultural and rural regions. We also sample Sentinel-2 tile regions that cover Satellogic data. Since Sentinel-1 does not follow the same grid system as Sentinel-2, we use the collected Sentinel-2 tile footprints to query Sentinel-1 captures, and crop them accordingly.

For each Sentinel-2 tile footprint, we extract 500 non-overlapping regions of 3,840 m x 3,840 m. Out of the all possible candidates (over 670) per tile, we select the best ones based on the amount of clouds and entropy (the ones with more class diversity using Sentinel-2 SCL mask). We end up collecting over 2,000 Sentinel-2 tiles, resulting in over 1M unique regions.

Temporal distribution Temporality also offers an important signal for a model to learn how scenes evolve over time. However, a long sequence could significantly increase the redundancy and size of the dataset. Hence we limit to 10 revisits per location, where 5 are densely sampled over time and other 5 are sampled with 3 months interval to ensure coverage of the seasons.

²AWS stores data in large tiles even though we only needed a fraction of the tile, we had to download a very large amount of tiles to obtain a broad coverage

5.2.2 Satellogic

Accessibility Satellogic is a provider of high-resolution remote sensing imagery. While data is not freely accessible, with the publication of this paper, Satellogic offers to release a significant portion of its 2021-2022 data under the license CC BY-NC 2.0.

Sensors Imagery is acquired at 1m GSD from space over 4 bands (blue, green, red, near-infrared).

Spatial Distribution The acquisition of imagery is on demand by the customer hence, there is not a broad coverage nor a systematic revisit. Hence, to maximize the number of revisits, we selected sites with high temporal overlap, and we rejected samples with high cloud coverage.

Temporality The resulting set of patches contain a varying number of revisits, ranging from 1-5.

5.2.3 NEON

NEON data combines high-resolution RGB, hyperspectral and lidar data for the study of ecological sites in the United States.

Accessibility NEON data is redistributed under CC0 1.0 and accessible on the NEON data portal.

Sensors NEON offers high-resolution RGB at 0.1m GSD and hyperspectral data comprised of 426 spectral bands at 1m GSD. It is also accompanied by lidar, which is post-processed to estimate the tree canopy height at 1m GSD.

Spatial Distribution This incredibly high-resolution data comes with very limited spatial distribution. We have collected data from 12 of the available sites with multiple sub-locations on each of these sites (See Figure 5.3). Each location spans $64\text{m} \times 64\text{m}$, covering $640 \text{ pixels} \times 640 \text{ pixels}$.

Temporality This data also offers yearly revisits with some limitations. Sites contain 3 revisits and the exact date was not collected. Nevertheless, we matched all available revisits for each location that we collected.

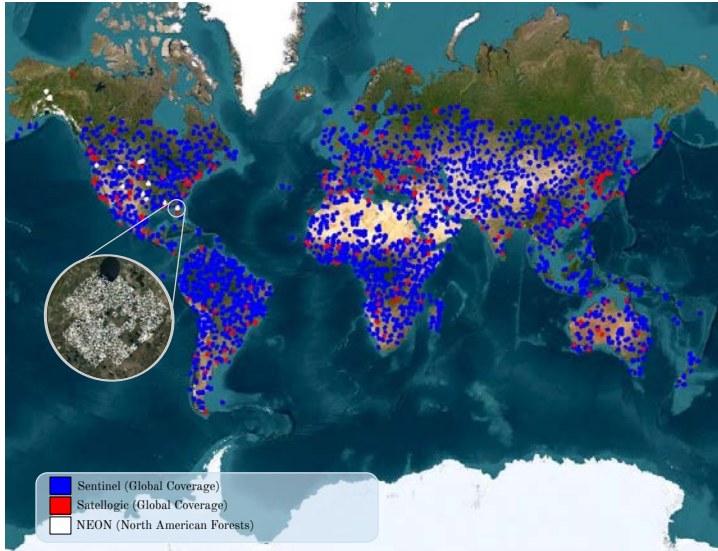


Figure 5.3: Spatial coverage for each source. Note that a coloured area may contain more than 1 patch.

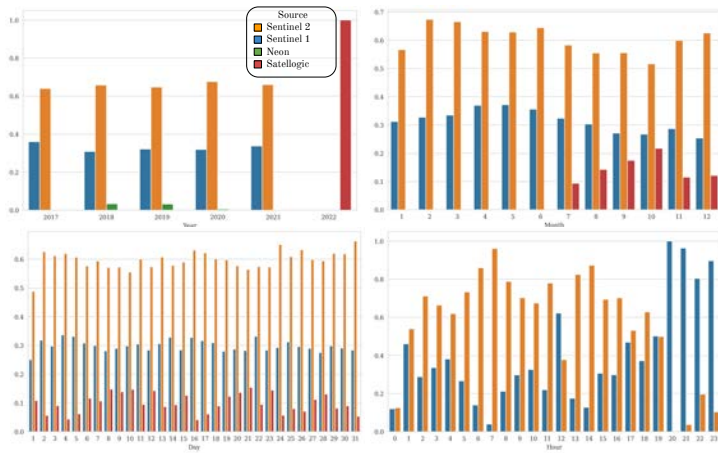


Figure 5.4: Temporal distribution of the dataset. For Neon we only have the year available and for satellogic we don't have the hour of the day when the picture was taken.

Table 5.1: Dataset overview

	Sensor	# bands	GSD (m)	Pixel per patch	area (m)	# revisits	# patches	# Giga gray pixels
Sentinel-1	SAR	2	10	384 x 384	3840 x 3840	3 - 9	1065514	942.70
Sentinel-2	Multi-Spectral	13	10, 20, 60	384 x 384	3840 x 3840	10	1065514	8,772.33
NEON-RGB	RGB	3	0.1	640 x 640	64 x 64	3	35501	130.87
NEON-Hyperspectral	Hyperspectral	369	1	64 x 64	64 x 64	3	35501	160.97
NEON-Elevation	Lidar	1	1	64 x 64	64 x 64	3	35501	0.44
Satelloic	RGBN	4	1	224 x 224	224 x 224	1 - 5	12476835	12,520.75

5.2.4 Hosting and Storage

Hosting Our aim is to make this data accessible for free. However, the data is too large for free hosting services like Zenodo and a *requester pays* approach on AWS leads to high costs for each download by the user. To this end, a partnership with Huggingface ensures persistent accessibility of the data at no cost with high bandwidth. For the reviewing process, a temporary subset of the dataset is stored on Zenodo.

Storage For accessibility and to minimize bandwidth, we store the dataset in functional subsets. That is, each of the different sensors can be downloaded separately and the set of locations for each is partitioned into 100 subsets. Also, each subset ensures uniform coverage of the data and matches the locations of other sensors for the same subset. This allows the user to download only a subset of the data if needed.

Format The dataset is stored in Hierarchical Data Format version 5 (HDF5) files, each dedicated to a discrete geographic location per data source. This organizational schema promotes efficient and methodical access to the data. Within each file, data is logically categorized by resolution and arrayed in a four-dimensional matrix structure (time, spectral bands, height, width).

Meta-Data We also store the geo-referenced bounding box, time stamp and other information when available in a standardized JSON format. We also provide azimuth angles and elevation data for each crop.

5.3 Proposed Model

In this work, we leverage a Masked Autoencoder (MAE) [63], distinguished by its asymmetrical encoder-decoder architecture. It incorporates an encoder that functions exclusively on a visible subset of patches, and a streamlined decoder, that rebuilds the original image from the latent representation and mask tokens.

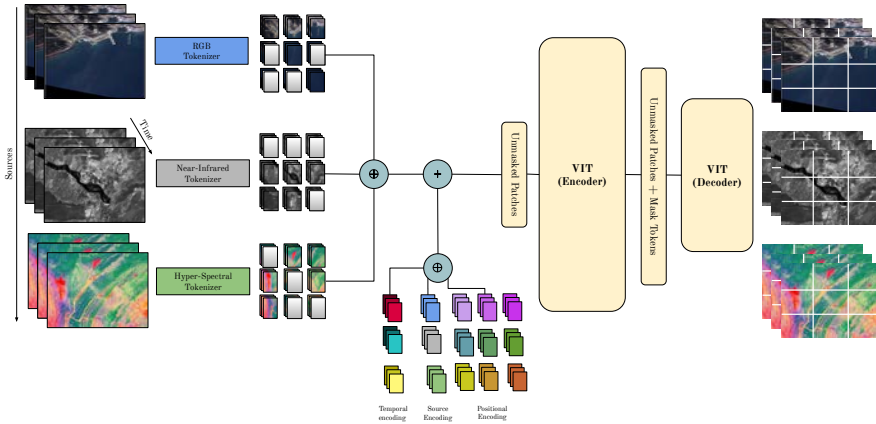


Figure 5.5: EarthMAE: The model we use in this work leverages time information and can digest data from an arbitrary number of sources. Each input source is tokenized into a fixed number of patches and then all patches are concatenated. The time, source and positional encodings are concatenated and added to the patches.

This model, recognized for its proficiency in self-supervised learning tasks, has been appropriately restructured to manage remote sensing data as described next.

5.3.1 EarthMAE

Our EarthMAE model (Figure 5.5) remains faithful to the original architecture, albeit we adjusted the tokenizers and positional encodings to leverage time and different modalities.

Tokenizers We incorporated a distinct tokenizer for each source, owing to the fact that different sources contain a disparate number of channels. This method offers a more nuanced comprehension of the data, accounting for the varied characteristics associated with different bands and sources.

Encoding We introduced source and temporal encodings, analogous to positional encodings. This provides the model with the capability to handle data from multiple sources at the same time (e.g., multi-spectral, RGB, hyper-spectral) along with multiple timesteps per source.

5.3.2 Training Paradigm

Our training approach makes the most of the unique mix of data in our dataset. This includes varied sensors and data types, such as multispectral data from Sentinel, hyperspectral data from NEON, RGB data from Satellogic, and specific bands like Sentinel-2 RGB and SCL used for segmentation tasks. To manage this broad spectrum of data, we've set up a task-based training system. Here, we distribute tasks across multiple GPUs, with each GPU handling a specific task. This way, we can process different types of data and sensors in parallel, making the process more efficient.

The training objective is the euclidian distance between the model's reconstruction and the normalized pixel values on masked patches same as in the original MAE [62] paper.

One of the notable aspects of our model is its use of flexible masking strategies. Instead of sticking to the traditional tube masking [194], where all timesteps are masked the same way, we've also incorporated random masking, where timesteps are masked in a more arbitrary manner. We've even included a combined approach that merges the two strategies. These alternative masking methods have proven to be beneficial, leading to better results across our varied datasets. Particularly, random and combined masking seems to enable the model to better understand the temporal patterns in the data, which in turn boosts the model's performance.

We've also made sure to include timestep information in our training approach, thanks to the timestamps provided in our dataset's metadata. This additional temporal layer gives the data more depth, reflecting the unique temporal characteristics of each dataset. This method allows our model to adjust to these temporal aspects, handling variable timesteps, and making it more suited for practical remote sensing tasks.

In short, our training approach uses a task-based system, and diverse masking strategies, and includes temporal information. These components work together to improve the flexibility and effectiveness of our EarthMAE model. This comprehensive approach matches well with the challenges presented by the various sensor data, timesteps, and masking schemes that are typical in the self-supervised learning of remote sensing data.

5.4 Experiments and Results

Our experiments aimed to understand the impacts of different data sources (NEON, Sentinel, Satellogic), the inclusion of temporality, and various masking

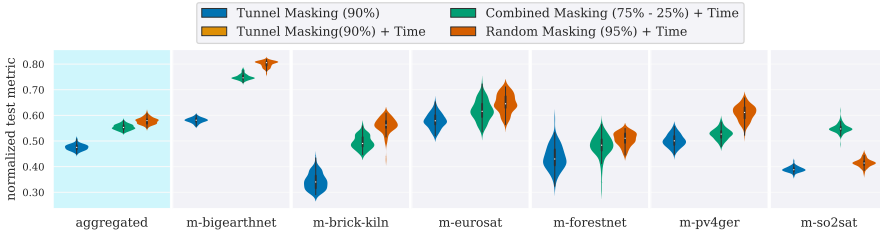


Figure 5.6: Performance of different masking schemas across time. It seems that random masking with a 95% ratio consistently outperform the rest. Tube masking 75% of the patches and randomly 25% of the remaining ones (Combined Masking) offers improved performance on one of the benchmarks. Results are reported across 10 different seeds.

strategies on the performance of our Masked Autoencoder (MAE) model.

To evaluate each pre-trained model, we leverage the classification benchmark of GeoBench [93]. This benchmark is specifically designed to evaluate pre-trained models on remote sensing data. They curated 6 classification datasets, including a modified version BigEarthNet, to cover a range of downstream tasks. On each downstream task, the pre-trained model is fine-tuned, and the best hyperparameter is selected on the validation set and re-trained with 10 different seeds to be evaluated on the test set. A bootstrap procedure is used to report the uncertainty of the interquartile mean³. A single aggregated result is obtained by averaging the normalized score⁴.

Following the standard MAE training process, all models were pre-trained for 400 epochs using a 90% masking ratio with tube masking [194], where all timesteps and sources are masked in the same way. We also experimented with different masking strategies by introducing random masking (where timesteps and sources are masked randomly with a 95% ratio) and combined masking, which mixes tube and random masking strategies.

Our experiments were conducted on several datasets, including m-BigEarthNet, m-Brick-Kiln, m-EuroSAT, m-ForestNet, m-PV4GER, and m-SO2SAT. The results showed significant variations in model performance, depending on the data sources, timesteps inclusion, and the applied masking schemes.

³The average of a sample where the top 25% and bottom 25% are discarded to be more robust to outliers.

⁴The benchmark provides per dataset normalization constants such that their weak baseline, e.g. ResNet18, has a score of zero and their strong baseline, e.g. SwinV2, has a score of 1.

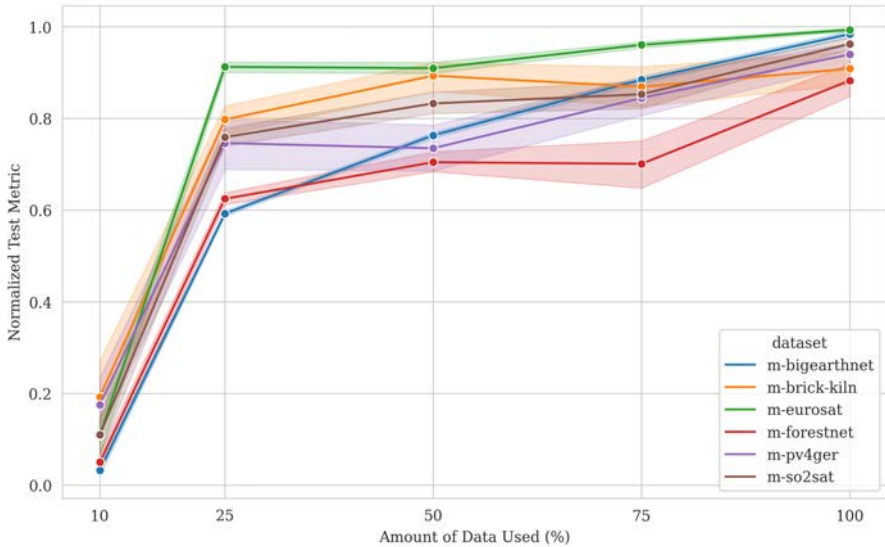


Figure 5.7: Performance on downstream tasks for different dataset sizes. Note how for some of the tasks (m-bigearhtnet, m-forestnet) the performance seems to increase almost linearly, indicating that there is still considerable room for improvement with an even larger dataset. Results are reported across 10 different seeds.

As seen in Figure 5.6, the inclusion of temporal features improves performance across all datasets, as long as the masking schema is not the same across timesteps. This suggests that the tube masking technique [194], which applies the same mask across all timesteps, might limit the effectiveness of integrating temporal features. This approach hinder the model’s ability to fully extract the temporal semantic information in the images. As a result, we saw improved performance when we used random or combined masking techniques, indicating that these strategies could better leverage the temporal context embedded in the data.

Figure 5.7 shows how performance on downstream tasks varies for different dataset sizes. For harder tasks the performance increases almost linearly indicating that despite the potential redundancy satellite imagery, our dataset is diverse enough so that using it at full scale offers improved results. In fact, it seems that for tasks like m-bigearhtnet and m-forestnet would benefit from an even larger scale.

5.5 Discussion

Our research introduces and describes an extensive, unique remote sensing dataset with over 22 trillion pixels, representing the most comprehensive dataset in its field. Its multiple sensors and data types offer a broad range of information to researchers, potentially revolutionizing remote sensing and related fields.

We have also developed EarthMAE, a model customized to handle this dataset’s intricacies. However, the crux of our work lies in the dataset’s enormous potential. The variety and size of the dataset allow for a thorough investigation of diverse sensor types and data structures. Its potential for distributed computation across multiple GPUs enables efficient exploration of various self-supervised learning situations.

Our work delves into multiple masking strategies, including random, tunnel, and combined, affecting the model’s performance. The dataset’s temporal information was integrated into the model using timestamp metadata, a move expected to enhance accuracy across a range of remote sensing tasks.

The implications of this dataset go well beyond the boundaries of our research. It opens up new horizons for future studies in self-supervised learning, remote sensing applications, and beyond. We are excited to see how this resource will fuel innovation and address intricate problems in the coming years. It is our hope that the work captured in this paper will act as a catalyst for future investigations, paving the way for remarkable progress.

Limitations While the EarthView dataset provides a range of sources, sensors and scales to train from, it does not provide other potential modalities such as text, or weather data. The EarthMAE model provided with this work does not reach the full potential of the EarthView dataset. We provide this as a teaser and we encourage other researchers to explore larger models trained on this dataset in combination with other available datasets such as [7].

6 Conclusions and Future work

6.1 Conclusions

This thesis marks an exploratory journey through the landscape of robustness in the field of computer vision. Over four detailed chapters, we've dipped our toes into a variety of areas. In chapter 2 we've worked to improve the detection of small objects in the DETR model. DETR-FP showed potential for better object detection when focusing on background or small objects, however this improvement comes at the cost of performance both in speed and in the detection of medium and large objects, probably due to repeated detections in the queries. In chapter 3 we explored different properties of the embedding propagation (EP) algorithm which has been shown to improve few-shot learning performance. We provide quantitative and qualitative insights showing that EP leads to a smoother manifold. We extend EP's results beyond few-shot showing that it improves adversarial robustness considerably and self-supervised learning performance.

In the third chapter 4, we introduced an evaluation framework that can bring some much-needed clarity to the field of counterfactual explanations. This framework addressed existing gaps in the evaluation process and gave us a better understanding of the explainability landscape. We make an effort to establish a fair, principled metric that is useful that provides unified metrics for evaluating different counterfactual explanation methods. The benchmark consists of synthetic images fully described by their annotated attributes which are accessible to the explainers through a differentiable generator. We show that modern explainers tend to produce redundant explanations and struggle to find correlated non-causal attributes in the data. We hope our findings encourage further research on fair evaluation benchmarks.

In the final chapter 5, we unveiled EarthView, a large-scale remote sensing dataset that we believe will open up new research possibilities in the creation of a foundational model for remote sensing applications. This chapter introduces the largest remote sensing dataset to date with over 22 trillion pixels. This curated dataset presents an opportunity for future research. We eagerly anticipate how

this extensive resource will be employed to drive innovation and tackle complex challenges in the years to come.

Despite these advancements, we're well aware that we're only at the start of our journey towards robustness in computer vision. Each study, while offering its unique insights and contributions, also pointed out areas that need further exploration. As the field of computer vision continues to evolve and find its way into more and more real-world applications, the pursuit of robustness will continue to be a major guiding factor.

This thesis is a contribution to the ongoing conversation on robustness in computer vision. It emphasizes the importance of coming up with innovative solutions and creating robust evaluation frameworks. We've learned that a truly robust system is not one that never fails, but one that is capable of learning from its failures and continuously improving. As we look towards the future, this kind of resilience will be key to the progress of computer vision systems.

6.2 Future Perspective

Object detection has evolved from two-stage detectors like R-CNN, Fast-RCNN, and Faster-RCNN [53] to one-stage detectors that streamline predictions [153]. However, these methodologies often depend on geometric priors. The DETR model [22] disrupts this norm by treating object detection as a set prediction problem, eliminating the need for these priors.

Future research could focus on combining multi-scale feature fusion techniques with DETR to improve small object detection. Further exploration of transformers [199] and their potential in parallel sequence generation might offer enhanced performance.

Refining bipartite matching loss functions to efficiently handle varying object sets across images while maintaining permutation-invariance is another worth while pursuit [91, 205]. Lastly, exploring different attention mechanism, tailored for object detection [237] is another promising direction.

Our work with Embedding Propagation (EP) reveals its potential for model robustness against adversarial attacks, and its application in few-shot, self-, and semi-supervised learning tasks. We suggest further exploration in scaling EP to more complex models and incorporating it into various neural network types [164].

There's scope for expanding the understanding of EP as a manifold regularization method. Improved comprehension of its theoretical underpinnings will enable more advanced application to adversarial attacks [57] and semi-

supervised learning tasks.

Incorporating EP as a MixUp strategy, exploiting embedding interpolations based on a similarity graph, may improve performance in low-data regimes. Exploring how this can be utilized in self-supervised learning could yield novel techniques and applications. The field of few-shot learning, which includes methods like meta-learning and transfer-learning, stands to benefit greatly from an in-depth exploration of EP, adding robustness to models and smoother decision boundaries in high-dimensional data spaces.

Explainability in deep learning is witnessing significant advancements, particularly in the area of counterfactual explanations. As underscored by the limitations of existing metrics like DiVE [161] and DiCE [126], the field is in need of robust, hard-to-game metrics for evaluating explainability methods. Future research should be geared towards developing metrics that ensure fair comparisons and can capture different aspects of explainability more accurately. An important direction for future research could be the incorporation of causal inference principles into the generation of counterfactual explanations, identifying which causal variables are the causal ones [92] to enhance their validity and interpretability. A challenging problem in this area is that the concept of optimal classifier is ill-defined solving this problem would help identifying causal factors in the absence of a human observer.

Our work in Chapter 5 led to the introduction of EarthView, a massive remote sensing dataset designed for self-supervised learning. However, there's a vast expanse of potential research that can stem from this. A key future direction is to devise novel self-supervised learning methods that can effectively exploit the unique properties of this dataset. For instance, there's potential to investigate temporal modeling approaches that take advantage of the dataset's extensive temporal coverage. Additionally, given the dataset's multimodality, exploring methods to effectively fuse and learn from different types of data (like RGB, hyperspectral, and radar) could be another exciting line of research. Lastly, given the potentially high redundancy intrinsic to satellite imagery a promising research direction would be to devise a data selection method that detects images that will not contribute much to training in a self-supervised setting, thus reducing redundancy in the while mantaining performance.

6.3 Scientific Articles

This dissertation has led to the following publications:

6.3.1 Journals

- **Velazquez, D**, Gonfaus, J. M., Rodriguez, P., Roca, F. X., Ozawa, S., and Gonzalez, J. (2021). Logo Detection With No Priors. *IEEE Access*, 9, 106998-107011.
- **Velazquez, D**, Rodriguez, P., Lacoste, A., Laradji, I. H., Roca, X., and González, J. (2023). Explaining Visual Counterfactual Explainers. *Transactions on Machine Learning Research*.
- **Velazquez, D**, Rodríguez, P., Gonfaus, J. M., Roca, F. X., and González, J. (2022). A closer look at embedding propagation for manifold smoothing. *The Journal of Machine Learning Research*, 23(1), 11447-11473.
- Rodriguez, P., **Velazquez, D.**, Cucurull, G., Gonfaus, J. M., Roca, F. X., and Gonzalez, J. (2019). Pay attention to the activations: A modular attention mechanism for fine-grained image recognition. *IEEE Transactions on Multimedia*, 22(2), 502-514.
- Rodríguez, P., **Velazquez, D.**, Cucurull, G., Gonfaus, J. M., Roca, F. X., Ozawa, S., and González, J. (2020). Personality trait analysis in social networks based on weakly supervised learning of shared images. *Applied Sciences*, 10(22), 8170.
- Ramírez-Cifuentes, D., Freire, A., Baeza-Yates, R., Puntí, J., Medina-Bravo, P., **Velazquez, D. A.**, Gonfaus, J.M. and González, J. (2020). Detection of suicidal ideation on social media: multimodal, relational, and behavioral analysis. *Journal of medical internet research*, 22(7), e17758.
- Ramírez-Cifuentes, D., Freire, A., Baeza-Yates, R., Sanz Lamora, N., Álvarez, A., González-Rodríguez, A., **Velazquez, D** and González, J. (2021). Characterization of anorexia nervosa on social media: textual, visual, relational, behavioral, and demographical analysis. *Journal of medical Internet research*, 23(7), e25925.

6.4 Contributed Code and Datasets

- **Explainability Benchmark (BeX)**: Code and dataset that comprises the benchmark presented in [200] <https://github.com/dvd42/BeX>
- **EarthView** Large scale earth monitoring dataset for foundation models. Coming soon.

6.4.1 In the Media

- “Racisme digital i COVID-19. Discursos racistes i antiracistes a Twitter durant la pandèmia” <https://tinyurl.com/2p9mzewz>, 2022

Bibliography

- [1] Feature importance, tree interpreter. <https://course18.fast.ai/lessonsm1/lesson4.html>. Accessed: 2020-09-13.
- [2] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.
- [3] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7):e0130140, 2015.
- [4] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.
- [5] Rachana Balasubramanian, Samuel Sharpe, Brian Barr, Jason Wittenbach, and C Bayan Bruss. Latent-cf: a simple baseline for reverse counterfactual explanations. *arXiv preprint arXiv:2012.09301*, 2020.
- [6] Peter Bartlett and John Shawe-Taylor. Generalization performance of support vector machines and other pattern classifiers. *Advances in Kernel methods—support vector learning*, 1999.
- [7] Favyen Bastani, Piper Wolters, Ritwik Gupta, Joe Ferdinando, and Aniruddha Kembhavi. Satlas: A large-scale, multi-task dataset for remote sensing image understanding. *arXiv preprint arXiv:2211.15660*, 2022.
- [8] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *European conference on computer vision*, pages 404–417. Springer, 2006.
- [9] Jasmine Bayrooti, Noah Goodman, and Alex Tamkin. Multispectral self-supervised learning with viewmaker networks. *arXiv preprint arXiv:2302.05757*, 2023.

- [10] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *JMLR*, 7(Nov):2399–2434, 2006.
- [11] Irwan Bello, Barret Zoph, Ashish Vaswani, Jonathon Shlens, and Quoc V Le. Attention augmented convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3286–3295, 2019.
- [12] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. Mixmatch: A holistic approach to semi-supervised learning. *arXiv preprint arXiv:1905.02249*, 2019.
- [13] Priyash Bhugra, Benjamin Bischke, Christoph Werner, Robert Syrnicki, Carolin Packbier, Patrick Helber, Caglar Senaras, Akhil Singh Rana, Tim Davis, Wanda De Keersmaecker, et al. Rapidai4eo: Mono-and multi-temporal deep learning models for updating the corine land cover product. In *IGARSS 2022-2022 IEEE International Geoscience and Remote Sensing Symposium*, pages 2247–2250. IEEE, 2022.
- [14] Simone Bianco, Marco Buzzelli, Davide Mazzini, and Raimondo Schettini. Logo recognition using cnn features. In *International Conference on Image Analysis and Processing*, pages 438–448. Springer, 2015.
- [15] Simone Bianco, Marco Buzzelli, Davide Mazzini, and Raimondo Schettini. Deep learning for logo recognition. *Neurocomputing*, 245:23–30, 2017.
- [16] Navaneeth Bodla, Bharat Singh, Rama Chellappa, and Larry S Davis. Soft-nms—improving object detection with one line of code. In *Proceedings of the IEEE international conference on computer vision*, pages 5561–5569, 2017.
- [17] Raluca Boia, Alessandra Bandrabur, and Corneliu Florea. Local description using multi-scale complete rank transform for improved logo recognition. In *2014 10th International Conference on Communications (COMM)*, pages 1–4. IEEE, 2014.
- [18] Daniel Bolya, Sean Foley, James Hays, and Judy Hoffman. Tide: A general toolbox for identifying object detection errors. In *ECCV*, 2020.
- [19] Malik Boudiaf, Imtiaz Ziko, Jérôme Rony, Jose Dolz, Pablo Piantanida, and Ismail Ben Ayed. Information maximization for few-shot learning. *Advances in Neural Information Processing Systems*, 33, 2020.

-
- [20] Changqing Cao, Bo Wang, Wenrui Zhang, Xiaodong Zeng, Xu Yan, Zhejun Feng, Yutao Liu, and Zengyan Wu. An improved faster r-cnn for small object detection. *IEEE Access*, 7:106838–106846, 2019.
- [21] Yue Cao, Zhenda Xie, Bin Liu, Yutong Lin, Zheng Zhang, and Han Hu. Parametric instance classification for unsupervised visual feature learning. *arXiv preprint arXiv:2006.14618*, 2020.
- [22] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. *arXiv preprint arXiv:2005.12872*, 2020.
- [23] Francesca Cesarini, Enrico Francesconi, Marco Gori, Simone Marinai, JQ Sheng, and Giovanni Soda. A neural-based architecture for spot-noisy logo recognition. In *Proceedings of the Fourth International Conference on Document Analysis and Recognition*, volume 1, pages 175–179. IEEE, 1997.
- [24] Anirban Chakraborty, Manaar Alam, Vishal Dey, Anupam Chattopadhyay, and Debdeep Mukhopadhyay. Adversarial attacks and defences: A survey. *arXiv preprint arXiv:1810.00069*, 2018.
- [25] William Chan, Chitwan Saharia, Geoffrey Hinton, Mohammad Norouzi, and Navdeep Jaitly. Imputer: Sequence modelling via imputation and dynamic programming. *arXiv preprint arXiv:2002.08926*, 2020.
- [26] Chun-Hao Chang, Elliot Creager, Anna Goldenberg, and David Duvenaud. Explaining image classifiers by counterfactual generation. *arXiv preprint arXiv:1807.08024*, 2018.
- [27] Olivier Chapelle and Alexander Zien. Semi-supervised classification by low density separation. In *AISTATS*, volume 2005, pages 57–64, 2005.
- [28] Jingying Chen, Maylor K Leung, and Yongsheng Gao. Noisy logo recognition using line segment hausdorff distance. *Pattern recognition*, 36(4):943–955, 2003.
- [29] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

- [30] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey Hinton. Big self-supervised models are strong semi-supervised learners. *arXiv preprint arXiv:2006.10029*, 2020.
- [31] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Frank Wang, and Jia-Bin Huang. A closer look at few-shot classification. In *ICLR*, 2019.
- [32] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020.
- [33] Ching-Yao Chuang, Joshua Robinson, Lin Yen-Chen, Antonio Torralba, and Stefanie Jegelka. Debaised contrastive learning. *arXiv preprint arXiv:2007.00224*, 2020.
- [34] Dan Claudiu Cireșan, Ueli Meier, Luca Maria Gambardella, and Jürgen Schmidhuber. Deep, big, simple neural nets for handwritten digit recognition. *Neural computation*, 22(12):3207–3220, 2010.
- [35] Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 215–223. JMLR Workshop and Conference Proceedings, 2011.
- [36] Yezhen Cong, Samar Khanna, Chenlin Meng, Patrick Liu, Erik Rozi, Yutong He, Marshall Burke, David Lobell, and Stefano Ermon. Satmae: Pre-training transformers for temporal and multi-spectral satellite imagery. *Advances in Neural Information Processing Systems*, 35:197–211, 2022.
- [37] Francesco Croce and Matthias Hein. Minimally distorted adversarial examples with a fast adaptive boundary attack. In *International Conference on Machine Learning*, pages 2196–2205. PMLR, 2020.
- [38] Emily Denton, Ben Hutchinson, Margaret Mitchell, and Timnit Gebru. Detecting bias with generative counterfactual face attribute augmentation. 2019.
- [39] 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.

- [40] David S Doermann, Ehud Rivlin, and Isaac Weiss. Logo recognition using geometric invariants. In *Proceedings of 2nd International Conference on Document Analysis and Recognition (ICDAR'93)*, pages 894–897. IEEE, 1993.
- [41] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *Proceedings of the IEEE international conference on computer vision*, pages 1422–1430, 2015.
- [42] Alexey Dosovitskiy, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with convolutional neural networks. Citeseer, 2014.
- [43] Nikita Dvornik, Cordelia Schmid, and Julien Mairal. Diversity with cooperation: Ensemble methods for few-shot classification. In *ICCV*, pages 3723–3731, 2019.
- [44] Andrew Elliott, Stephen Law, and Chris Russell. Explaining classifiers using adversarial perturbations on the perceptual ball. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10693–10702, 2021.
- [45] Dumitru Erhan, Christian Szegedy, Alexander Toshev, and Dragomir Anguelov. Scalable object detection using deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2147–2154, 2014.
- [46] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, pages 1126–1135. JMLR. org, 2017.
- [47] Ruth Fong, Mandela Patrick, and Andrea Vedaldi. Understanding deep networks via extremal perturbations and smooth masks. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2950–2958, 2019.
- [48] Enrico Francesconi, Paolo Frasconi, Marco Gori, Simone Marinai, JQ Sheng, Giovanni Soda, and Alessandro Sperduti. Logo recognition by recursive neural networks. In *International Workshop on Graphics Recognition*, pages 104–117. Springer, 1997.
- [49] Victor Garcia and Joan Bruna. Few-shot learning with graph neural networks. *ICLR*, 2017.

- [50] Marjan Ghazvininejad, Omer Levy, Yinhan Liu, and Luke Zettlemoyer. Mask-predict: Parallel decoding of conditional masked language models. *arXiv preprint arXiv:1904.09324*, 2019.
- [51] Spyros Gidaris, Andrei Bursuc, Nikos Komodakis, Patrick Pérez, and Matthieu Cord. Boosting few-shot visual learning with self-supervision. In *CVPR*, pages 8059–8068, 2019.
- [52] Spyros Gidaris and Nikos Komodakis. Generating classification weights with gnn denoising autoencoders for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 21–30, 2019.
- [53] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.
- [54] Ross B. Girshick. Fast R-CNN. *CoRR*, abs/1504.08083, 2015.
- [55] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256, 2010.
- [56] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.
- [57] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.
- [58] Yves Grandvalet, Yoshua Bengio, et al. Semi-supervised learning by entropy minimization. In *CAP*, pages 281–296, 2005.
- [59] Jiatao Gu, James Bradbury, Caiming Xiong, Victor OK Li, and Richard Socher. Non-autoregressive neural machine translation. *arXiv preprint arXiv:1711.02281*, 2017.
- [60] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Dino Pedreschi, Franco Turini, and Fosca Giannotti. Local rule-based explanations of black box decision systems. *arXiv preprint arXiv:1805.10820*, 2018.

-
- [61] Bharath Hariharan and Ross Girshick. Low-shot visual recognition by shrinking and hallucinating features. In *ICCV*, pages 3018–3027, 2017.
- [62] 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*, pages 16000–16009, 2022.
- [63] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross B Girshick. Masked autoencoders are scalable vision learners. corr abs/2111.06377 (2021). *arXiv preprint arXiv:2111.06377*, 2021.
- [64] 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*, pages 9729–9738, 2020.
- [65] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. *CoRR*, abs/1703.06870, 2017.
- [66] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [67] Chih-Hui Ho and Nuno Vasconcelos. Contrastive learning with adversarial examples. *arXiv preprint arXiv:2010.12050*, 2020.
- [68] Steven CH Hoi, Xiongwei Wu, Hantang Liu, Yue Wu, Huiqiong Wang, Hui Xue, and Qiang Wu. Logo-net: Large-scale deep logo detection and brand recognition with deep region-based convolutional networks. *arXiv preprint arXiv:1511.02462*, 2015.
- [69] Jan Hosang, Rodrigo Benenson, and Bernt Schiele. Learning non-maximum suppression. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4507–4515, 2017.
- [70] Ruibing Hou, Hong Chang, MA Bingpeng, Shiguang Shan, and Xilin Chen. Cross attention network for few-shot classification. In *Advances in Neural Information Processing Systems*, pages 4005–4016, 2019.
- [71] Han Hu, Jiayuan Gu, Zheng Zhang, Jifeng Dai, and Yichen Wei. Relation networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3588–3597, 2018.

- [72] Shell Xu Hu, Pablo Moreno, Yang Xiao, Xi Shen, Guillaume Obozinski, Neil Lawrence, and Andreas Damianou. Empirical bayes transductive meta-learning with synthetic gradients. In *International Conference on Learning Representations (ICLR)*, 2020.
- [73] Yuqing Hu, Vincent Gripon, and Stéphane Pateux. Exploiting unsupervised inputs for accurate few-shot classification, 2020.
- [74] Forrest N Iandola, Anting Shen, Peter Gao, and Kurt Keutzer. Deeplogo: Hitting logo recognition with the deep neural network hammer. *arXiv preprint arXiv:1510.02131*, 2015.
- [75] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, pages 448–456, 2015.
- [76] Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, and Ondřej Chum. Mining on manifolds: Metric learning without labels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7642–7651, 2018.
- [77] Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, and Ondrej Chum. Label propagation for deep semi-supervised learning. In *CVPR*, pages 5070–5079, 2019.
- [78] Ramesh Jain, Rangachar Kasturi, and Brian G Schunck. *Machine vision*, volume 5. McGraw-hill New York, 1995.
- [79] Simon Jégou, Michal Drozdal, David Vazquez, Adriana Romero, and Yoshua Bengio. The one hundred layers tiramisù: Fully convolutional densenets for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 11–19, 2017.
- [80] Alexis Joly and Olivier Buisson. Logo retrieval with a contrario visual query expansion. In *Proceedings of the 17th ACM international conference on Multimedia*, pages 581–584, 2009.
- [81] Alexis Joly and Olivier Buisson. Logo retrieval with a contrario visual query expansion. In *MM '09: Proceedings of the seventeen ACM international conference on Multimedia*, pages 581–584, 2009.

- [82] Shalmali Joshi, Oluwasanmi Koyejo, Been Kim, and Joydeep Ghosh. xgems: Generating exemplars to explain black-box models. *arXiv preprint arXiv:1806.08867*, 2018.
- [83] Y. Kalantidis, LG. Pueyo, M. Trevisiol, R. van Zwol, and Y. Avrithis. Scalable triangulation-based logo recognition. In *in Proceedings of ACM International Conference on Multimedia Retrieval (ICMR 2011)*, Trento, Italy, April 2011.
- [84] Yannis Kalantidis, Mert Bulent Sariyildiz, Noe Pion, Philippe Weinzaepfel, and Diane Larlus. Hard negative mixing for contrastive learning. *arXiv preprint arXiv:2010.01028*, 2020.
- [85] Amir-Hossein Karimi, Bernhard Schölkopf, and Isabel Valera. Algorithmic recourse: from counterfactual explanations to interventions. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 353–362, 2021.
- [86] Sami Khanal, Kushal Kc, John P Fulton, Scott Shearer, and Erdal Ozkan. Remote sensing in agriculture—accomplishments, limitations, and opportunities. *Remote Sensing*, 12(22):3783, 2020.
- [87] Ilyes Khemakhem, Diederik Kingma, Ricardo Monti, and Aapo Hyvarinen. Variational autoencoders and nonlinear ica: A unifying framework. In *International Conference on Artificial Intelligence and Statistics*, pages 2207–2217. PMLR, 2020.
- [88] Jongmin Kim, Taesup Kim, Sungwoong Kim, and Chang D Yoo. Edge-labeling graph neural network for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11–20, 2019.
- [89] Johannes Klicpera, Aleksandar Bojchevski, and Stephan Günnemann. Predict then propagate: Graph neural networks meet personalized pagerank. *arXiv preprint arXiv:1810.05997*, 2018.
- [90] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.
- [91] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

- [92] Sébastien Lachapelle, Pau Rodriguez, Yash Sharma, Katie E Everett, Rémi Le Priol, Alexandre Lacoste, and Simon Lacoste-Julien. Disentanglement via mechanism sparsity regularization: A new principle for nonlinear ica. In *Conference on Causal Learning and Reasoning*, pages 428–484. PMLR, 2022.
- [93] Alexandre Lacoste, Nils Lehmann, Pau Rodriguez, Evan David Sherwin, Hannah Kerner, Björn Lütjens, Jeremy Andrew Irvin, David Dao, Hamed Alemohammad, Alexandre Drouin, Mehmet Gunturkun, Gabriel Huang, David Vazquez, Dava Newman, Yoshua Bengio, Stefano Ermon, and Xiao Xiang Zhu. Geo-bench: Toward foundation models for earth monitoring, 2023.
- [94] Alexandre Lacoste, Pau Rodriguez, Frédéric Branchaud-Charron, Parmida Atighehchian, Massimo Caccia, Issam Laradji, Alexandre Drouin, Matt Craddock, Laurent Charlin, and David Vázquez. Symbols: Probing learning algorithms with synthetic datasets. *arXiv preprint arXiv:2009.06415*, 2020.
- [95] Alexandre Lacoste, Evan David Sherwin, Hannah Kerner, Hamed Alemohammad, Björn Lütjens, Jeremy Irvin, David Dao, Alex Chang, Mehmet Gunturkun, Alexandre Drouin, Pau Rodriguez, and David Vazquez. Toward foundation models for earth monitoring: Proposal for a climate change benchmark, 2021.
- [96] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. *arXiv preprint arXiv:1610.02242*, 2016.
- [97] Oran Lang, Yossi Gandelsman, Michal Yarom, Yoav Wald, Gal Elidan, Avinatan Hassidim, William T Freeman, Phillip Isola, Amir Globerson, Michal Irani, et al. Explaining in style: Training a gan to explain a classifier in stylespace. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 693–702, 2021.
- [98] Issam Laradji, Pau Rodriguez, Freddie Kalaitzis, David Vazquez, Ross Young, Ed Davey, and Alexandre Lacoste. Counting cows: Tracking illegal cattle ranching from high-resolution satellite imagery. *arXiv preprint arXiv:2011.07369*, 2020.
- [99] Thibault Laugel, Marie-Jeanne Lesot, Christophe Marsala, Xavier Renard, and Marcin Detyniecki. Inverse classification for comparison-based interpretability in machine learning. *arXiv preprint arXiv:1712.08443*, 2017.

-
- [100] Deep Learning. Ian goodfellow, yoshua bengio, aaron courville. *The reference book for deep learning models*, 2016.
- [101] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.
- [102] Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010.
- [103] Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 2, 2013.
- [104] Kwonjoon Lee, Subhransu Maji, Avinash Ravichandran, and Stefano Soatto. Meta-learning with differentiable convex optimization. In *CVPR*, pages 10657–10665, 2019.
- [105] Wee Sun Lee, Peter L Bartlett, and Robert C Williamson. Lower bounds on the vc dimension of smoothly parameterized function classes. *Neural Computation*, 7(5):1040–1053, 1995.
- [106] Xinzhe Li, Qianru Sun, Yaoyao Liu, Qin Zhou, Shibao Zheng, Tat-Seng Chua, and Bernt Schiele. Learning to self-train for semi-supervised few-shot classification. In *NeurIPS*, pages 10276–10286, 2019.
- [107] Y. Liao, X. Lu, C. Zhang, Y. Wang, and Z. Tang. Mutual enhancement for detection of multiple logos in sports videos. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 4856–4865, 2017.
- [108] 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.
- [109] 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.
- [110] 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 *European conference on computer vision*, pages 740–755. Springer, 2014.

- [111] Jinlu Liu, Liang Song, and Yongqiang Qin. Prototype rectification for few-shot learning. *ECCV*, 2019.
- [112] Shusen Liu, Bhavya Kailkhura, Donald Loveland, and Yong Han. Generative counterfactual introspection for explainable deep learning. In *2019 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pages 1–5. IEEE, 2019.
- [113] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multi-box detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.
- [114] Yanbin Liu, Juho Lee, Minseop Park, Saehoon Kim, Eunho Yang, Sungju Hwang, and Yi Yang. Learning to propagate labels: transductive propagation network for few-shot learning. In *ICLR*, 2019.
- [115] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.
- [116] Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *international conference on machine learning*, pages 4114–4124. PMLR, 2019.
- [117] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.
- [118] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.
- [119] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.
- [120] Christoph Lüscher, Eugen Beck, Kazuki Irie, Markus Kitza, Wilfried Michel, Albert Zeyer, Ralf Schlüter, and Hermann Ney. Rwth asr systems for librispeech: Hybrid vs attention–w/o data augmentation. *arXiv preprint arXiv:1905.03072*, 2019.
- [121] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

- [122] Oscar Manas, Alexandre Lacoste, Xavier Giró-i Nieto, David Vazquez, and Pau Rodriguez. Seasonal contrast: Unsupervised pre-training from uncurated remote sensing data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9414–9423, 2021.
- [123] Puneet Mangla, Mayank Singh, Abhishek Sinha, Nupur Kumari, Vineeth N Balasubramanian, and Balaji Krishnamurthy. Charting the right manifold: Manifold mixup for few-shot learning. *arXiv preprint arXiv:1907.12087*, 2019.
- [124] Thomas Mensink, Jakob Verbeek, Florent Perronnin, and Gabriela Csurka. Metric learning for large scale image classification: Generalizing to new classes at near-zero cost. In *European Conference on Computer Vision*, pages 488–501. Springer, 2012.
- [125] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1979–1993, 2018.
- [126] Ramaravind K Mothilal, Amit Sharma, and Chenhao Tan. Explaining machine learning classifiers through diverse counterfactual explanations. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 607–617, 2020.
- [127] Maik Netzband, William L Stefanov, and Charles Redman. *Applied remote sensing for urban planning, governance and sustainability*. Springer Science & Business Media, 2007.
- [128] David Nister and Henrik Stewenius. Scalable recognition with a vocabulary tree. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 2161–2168. Ieee, 2006.
- [129] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European conference on computer vision*, pages 69–84. Springer, 2016.
- [130] Gonçalo Oliveira, Xavier Frazão, André Pimentel, and Bernardete Ribeiro. Automatic graphic logo detection via fast region-based convolutional networks. In *2016 International Joint Conference on Neural Networks (IJCNN)*, pages 985–991. IEEE, 2016.

- [131] Aaron Oord, Yazhe Li, Igor Babuschkin, Karen Simonyan, Oriol Vinyals, Koray Kavukcuoglu, George Driessche, Edward Lockhart, Luis Cobo, Florian Stimberg, et al. Parallel wavenet: Fast high-fidelity speech synthesis. In *International conference on machine learning*, pages 3918–3926. PMLR, 2018.
- [132] Aaron van den Oord, Yazhe Li, Igor Babuschkin, Karen Simonyan, Oriol Vinyals, Koray Kavukcuoglu, George van den Driessche, Edward Lockhart, Luis C Cobo, Florian Stimberg, et al. Parallel wavenet: Fast high-fidelity speech synthesis. *arXiv preprint arXiv:1711.10433*, 2017.
- [133] OpenAI. Gpt-4 technical report, 2023.
- [134] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2023.
- [135] Boris Oreshkin, Pau Rodríguez López, and Alexandre Lacoste. Tadam: Task dependent adaptive metric for improved few-shot learning. In *NeurIPS*, pages 721–731, 2018.
- [136] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning, 2017. In *ACM Asia Conference on Computer and Communications Security*, 2016.
- [137] Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z Berkay Celik, and Ananthram Swami. The limitations of deep learning in adversarial settings. In *2016 IEEE European symposium on security and privacy (EuroS&P)*, pages 372–387. IEEE, 2016.
- [138] Eunbyung Park and Alexander C Berg. Learning to decompose for object detection and instance segmentation. *arXiv preprint arXiv:1511.06449*, 2015.
- [139] Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Łukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. Image transformer. *arXiv preprint arXiv:1802.05751*, 2018.

- [140] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.
- [141] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.
- [142] Martin Pawelczyk, Sascha Bielawski, Johannes van den Heuvel, Tobias Richter, and Gjergji Kasneci. Carla: A python library to benchmark algorithmic recourse and counterfactual explanation algorithms, 2021.
- [143] Martin Pawelczyk, Klaus Broelemann, and Gjergji Kasneci. Learning model-agnostic counterfactual explanations for tabular data. In *Proceedings of The Web Conference 2020*, pages 3126–3132, 2020.
- [144] Judea Pearl. *Causality*. Cambridge university press, 2009.
- [145] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.
- [146] Apostolos P Pysillos, Christos-Nikolaos E Anagnostopoulos, and Eleftherios Kayafas. Vehicle logo recognition using a sift-based enhanced matching scheme. *IEEE transactions on intelligent transportation systems*, 11(2):322–328, 2010.
- [147] Xiaoman Qi, Panpan Zhu, Yuebin Wang, Liqiang Zhang, Junhuan Peng, Mengfan Wu, Jialong Chen, Xudong Zhao, Ning Zang, and P Takis Mathiopoulos. Mlrsnet: A multi-label high spatial resolution remote sensing dataset for semantic scene understanding. *ISPRS Journal of Photogrammetry and Remote Sensing*, 169:337–350, 2020.
- [148] 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.
- [149] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9, 2019.
- [150] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *ICLR*, 2016.
- [151] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.
- [152] 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.
- [153] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.
- [154] Colorado J Reed, Ritwik Gupta, Shufan Li, Sarah Brockman, Christopher Funk, Brian Clipp, Salvatore Candido, Matt Uyttendaele, and Trevor Darrell. Scale-mae: A scale-aware masked autoencoder for multiscale geospatial representation learning. *arXiv preprint arXiv:2212.14532*, 2022.
- [155] Mengye Ren, Sachin Ravi, Eleni Triantafillou, Jake Snell, Kevin Swersky, Josh B. Tenenbaum, Hugo Larochelle, and Richard S. Zemel. Meta-learning for semi-supervised few-shot classification. In *ICLR*, 2018.
- [156] Mengye Ren and Richard S Zemel. End-to-end instance segmentation with recurrent attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6656–6664, 2017.
- [157] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NeurIPS*, pages 91–99, 2015.
- [158] Hamid Rezaatofghi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE*

-
- Conference on Computer Vision and Pattern Recognition*, pages 658–666, 2019.
- [159] S Hamid Rezatofghi, Anton Milan, Ehsan Abbasnejad, Anthony Dick, Ian Reid, et al. Deepsetnet: Predicting sets with deep neural networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 5257–5266. IEEE, 2017.
- [160] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.
- [161] Pau Rodríguez, Massimo Caccia, Alexandre Lacoste, Lee Zamparo, Issam Laradji, Laurent Charlin, and David Vazquez. Beyond trivial counterfactual explanations with diverse valuable explanations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1056–1065, October 2021.
- [162] Pau Rodriguez, Guillem Cucurull, Jordi Gonzàlez, Josep M Gonfaus, Kamal Nasrollahi, Thomas B Moeslund, and F Xavier Roca. Deep pain: Exploiting long short-term memory networks for facial expression classification. *IEEE transactions on cybernetics*, 52(5):3314–3324, 2017.
- [163] Pau Rodríguez, Jordi Gonzalez, Guillem Cucurull, Josep M Gonfaus, and Xavier Roca. Regularizing cnns with locally constrained decorrelations. *ICLR*, 2016.
- [164] Pau Rodríguez, Issam Laradji, Alexandre Drouin, and Alexandre Lacoste. Embedding propagation: Smoother manifold for few-shot classification. In *European Conference on Computer Vision*, pages 121–138. Springer, 2020.
- [165] David Rolnick, Priya L Donti, Lynn H Kaack, Kelly Kochanski, Alexandre Lacoste, Kris Sankaran, Andrew Slavin Ross, Nikola Milojevic-Dupont, Natasha Jaques, Anna Waldman-Brown, et al. Tackling climate change with machine learning. *ACM Computing Surveys (CSUR)*, 55(2):1–96, 2022.
- [166] Stefan Romberg and Rainer Lienhart. Bundle min-hashing for logo recognition. In *Proceedings of the 3rd ACM conference on International conference on multimedia retrieval*, pages 113–120, 2013.

- [167] Stefan Romberg, Lluís Garcia Pueyo, Rainer Lienhart, and Roelof Van Zwol. Scalable logo recognition in real-world images. In *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, pages 1–8, 2011.
- [168] Bernardino Romera-Paredes and Philip Hilaire Sean Torr. Recurrent instance segmentation. In *European conference on computer vision*, pages 312–329. Springer, 2016.
- [169] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *2011 International conference on computer vision*, pages 2564–2571. Ieee, 2011.
- [170] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *IJCC*, 2015.
- [171] Andrei A Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. Meta-learning with latent embedding optimization. In *ICLR*, 2018.
- [172] Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. *arXiv preprint arXiv:1606.04586*, 2016.
- [173] Tim Salimans and Durk P Kingma. Weight normalization: A simple reparameterization to accelerate training of deep neural networks. In *NeurIPS*, pages 901–909, 2016.
- [174] Amaia Salvador, Miriam Bellver, Victor Campos, Manel Baradad, Ferran Marques, Jordi Torres, and Xavier Giro-i Nieto. Recurrent neural networks for semantic instance segmentation. *arXiv preprint arXiv:1712.00617*, 2017.
- [175] Axel Sauer and Andreas Geiger. Counterfactual generative networks. *arXiv preprint arXiv:2101.06046*, 2021.
- [176] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *International conference on machine learning*, pages 3145–3153. PMLR, 2017.

-
- [177] Patrice Y Simard, David Steinkraus, John C Platt, et al. Best practices for convolutional neural networks applied to visual document analysis. In *Icdar*, volume 3. Citeseer, 2003.
- [178] Sumedha Singla, Brian Pollack, Junxiang Chen, and Kayhan Batmanghelich. Explanation by progressive exaggeration. *arXiv preprint arXiv:1911.00483*, 2019.
- [179] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *NeurIPS*, pages 4077–4087, 2017.
- [180] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *JMLR*, 15(1):1929–1958, 2014.
- [181] Russell Stewart, Mykhaylo Andriluka, and Andrew Y Ng. End-to-end people detection in crowded scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2325–2333, 2016.
- [182] Hang Su, Xiatian Zhu, and Shaogang Gong. Deep learning logo detection with data expansion by synthesising context. In *Applications of Computer Vision (WACV), 2017 IEEE Winter Conference on*, pages 530–539. IEEE, 2017.
- [183] Hang Su, Xiatian Zhu, and Shaogang Gong. Open logo detection challenge. In *British Machine Vision Conference*, 2018.
- [184] Gencer Sumbul, Marcela Charfuelan, Begüm Demir, and Volker Markl. Bigearthnet: A large-scale benchmark archive for remote sensing image understanding. In *IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium*, pages 5901–5904. IEEE, 2019.
- [185] Qianru Sun, Yaoyao Liu, Tat-Seng Chua, and Bernt Schiele. Meta-transfer learning for few-shot learning. In *CVPR*, pages 403–412, 2019.
- [186] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *CVPR*, pages 1199–1208, 2018.
- [187] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.

- [188] Gabriel Synnaeve, Qiantong Xu, Jacob Kahn, Edouard Grave, Tatiana Likhomanenko, Vineel Pratap, Anuroop Sriram, Vitaliy Liptchinsky, and Ronan Collobert. End-to-end asr: from supervised to semi-supervised learning with modern architectures. *arXiv preprint arXiv:1911.08460*, 2019.
- [189] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.
- [190] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NeurIPS*, pages 1195–1204, 2017.
- [191] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 9627–9636, 2019.
- [192] Yuji Tokozume, Yoshitaka Ushiku, and Tatsuya Harada. Between-class learning for image classification. In *CVPR*, pages 5486–5494, 2018.
- [193] Xin-Yi Tong, Gui-Song Xia, and Xiao Xiang Zhu. Enabling country-scale land cover mapping with meter-resolution satellite imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 196:178–196, 2023.
- [194] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In *Advances in Neural Information Processing Systems*.
- [195] Andras Tüzkö, Christian Herrmann, Daniel Manger, and Jürgen Beyerer. Open set logo detection and retrieval. *arXiv preprint arXiv:1710.10891*, 2017.
- [196] Andras Tüzkö, Christian Herrmann, Daniel Manger, and Jürgen Beyerer. Open Set Logo Detection and Retrieval. In *Proceedings of the 13th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications: VISAPP*, 2018.
- [197] Arnaud Van Looveren, Janis Klaise, Giovanni Vacanti, and Oliver Cobb. Conditional generative models for counterfactual explanations. *arXiv preprint arXiv:2101.10123*, 2021.
- [198] Vladimir N Vapnik. An overview of statistical learning theory. *IEEE transactions on neural networks*, 1999.

-
- [199] 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.
- [200] Diego Velazquez, Pau Rodriguez, Alexandre Lacoste, Issam H Laradji, Xavier Roca, and Jordi González. Explaining visual counterfactual explainers. *Transactions on Machine Learning Research*, 2023.
- [201] Vikas Verma, Alex Lamb, Christopher Beckham, Amir Najafi, Ioannis Mitliagkas, David Lopez-Paz, and Yoshua Bengio. Manifold mixup: Better representations by interpolating hidden states. In *ICML*, pages 6438–6447, 2019.
- [202] Vikas Verma, Alex Lamb, Juho Kannala, Yoshua Bengio, and David Lopez-Paz. Interpolation consistency training for semi-supervised learning. *stat*, 1050:19, 2019.
- [203] Vikas Verma, Minh-Thang Luong, Kenji Kawaguchi, Hieu Pham, and Quoc V Le. Towards domain-agnostic contrastive learning. *arXiv preprint arXiv:2011.04419*, 2020.
- [204] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103, 2008.
- [205] Oriol Vinyals, Samy Bengio, and Manjunath Kudlur. Order matters: Sequence to sequence for sets. *arXiv preprint arXiv:1511.06391*, 2015.
- [206] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *NeurIPS*, pages 3630–3638, 2016.
- [207] Li Wan, Matthew Zeiler, Sixin Zhang, Yann Le Cun, and Rob Fergus. Regularization of neural networks using dropconnect. In *International conference on machine learning*, pages 1058–1066. PMLR, 2013.
- [208] Bo Wang, Zhuowen Tu, and John K Tsotsos. Dynamic label propagation for semi-supervised multi-class multi-label classification. In *Proceedings of the IEEE international conference on computer vision*, pages 425–432, 2013.

- [209] Jiaqi Wang, Kai Chen, Shuo Yang, Chen Change Loy, and Dahua Lin. Region proposal by guided anchoring. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2965–2974, 2019.
- [210] Liya Wang and Alex Tien. Remote sensing scene classification with masked image modeling (mim). *arXiv preprint arXiv:2302.14256*, 2023.
- [211] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, pages 9929–9939. PMLR, 2020.
- [212] Xiaolong Wang and Abhinav Gupta. Unsupervised learning of visual representations using videos. In *Proceedings of the IEEE international conference on computer vision*, pages 2794–2802, 2015.
- [213] Yan Wang, Wei-Lun Chao, Kilian Q Weinberger, and Laurens van der Maaten. Simpleshot: Revisiting nearest-neighbor classification for few-shot learning. *arXiv preprint arXiv:1911.04623*, 2019.
- [214] Xinye Wanyan, Sachith Seneviratne, Shuchang Shen, and Michael Kirley. Dino-mc: Self-supervised contrastive learning for remote sensing imagery with multi-sized local crops. *arXiv preprint arXiv:2303.06670*, 2023.
- [215] Mike Wu, Chengxu Zhuang, Milan Mosse, Daniel Yamins, and Noah Goodman. On mutual information in contrastive learning for visual representations. *arXiv preprint arXiv:2005.13149*, 2020.
- [216] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. <https://github.com/facebookresearch/detectron2>, 2019.
- [217] Zhong Wu, Qifa Ke, Michael Isard, and Jian Sun. Bundling features for large scale partial-duplicate web image search. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 25–32, 2009.
- [218] Louis-Pascal Xhonneux, Meng Qu, and Jian Tang. Continuous graph neural networks. In *International Conference on Machine Learning*, pages 10432–10441. PMLR, 2020.
- [219] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

-
- [220] Jiahao Xie, Xiaohang Zhan, Ziwei Liu, Yew Soon Ong, and Chen Change Loy. Delving into inter-image invariance for unsupervised visual representations. *arXiv preprint arXiv:2008.11702*, 2020.
- [221] Chen Xing, Negar Rostamzadeh, Boris Oreshkin, and Pedro OO Pinheiro. Adaptive cross-modal few-shot learning. In *NeurIPS*, pages 4848–4858, 2019.
- [222] Fan Yang, Ninghao Liu, Mengnan Du, and Xia Hu. Generative counterfactuals for neural networks via attribute-informed perturbation. *ACM SIGKDD Explorations Newsletter*, 23(1):59–68, 2021.
- [223] Tong Yang, Xiangyu Zhang, Zeming Li, Wenqiang Zhang, and Jian Sun. Metaanchor: Learning to detect objects with customized anchors. *arXiv preprint arXiv:1807.00980*, 2018.
- [224] Han-Jia Ye, Hexiang Hu, De-Chuan Zhan, and Fei Sha. Few-shot learning via embedding adaptation with set-to-set functions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8808–8817, 2020.
- [225] Mang Ye, Xu Zhang, Pong C Yuen, and Shih-Fu Chang. Unsupervised embedding learning via invariant and spreading instance feature. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6210–6219, 2019.
- [226] Sung Whan Yoon, Jun Seo, and Jaekyun Moon. Tapnet: Neural network augmented with task-adaptive projection for few-shot learning. In *International Conference on Machine Learning*, pages 7115–7123. PMLR, 2019.
- [227] Zhongjie Yu, Lin Chen, Zhongwei Cheng, and Jiebo Luo. Transmatch: A transfer-learning scheme for semi-supervised few-shot learning. In *CVPR*, pages 12856–12864, 2020.
- [228] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *BMVC*, pages 87.1–87.12. BMVA Press, September 2016.
- [229] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *ICLR*, 2018.
- [230] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *European conference on computer vision*, pages 649–666. Springer, 2016.

- [231] Shifeng Zhang, Cheng Chi, Yongqiang Yao, Zhen Lei, and Stan Z Li. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9759–9768, 2020.
- [232] Dengyong Zhou, Olivier Bousquet, Thomas N Lal, Jason Weston, and Bernhard Schölkopf. Learning with local and global consistency. In *NeurIPS*, pages 321–328, 2004.
- [233] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019.
- [234] Yilun Zhou, Serena Booth, Marco Tulio Ribeiro, and Julie Shah. Do feature attribution methods correctly attribute features? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 9623–9633, 2022.
- [235] G. Zhu and D. Doermann. Automatic document logo detection. In *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, volume 2, pages 864–868, 2007.
- [236] Lingxuan Zhu, Jiaji Wu, Wang Biao, Yi Liao, and Dandan Gu. Spectralmae: Spectral masked autoencoder for hyperspectral remote sensing image reconstruction. *Sensors*, 23(7):3728, 2023.
- [237] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020.
- [238] Imtiaz Ziko, Jose Dolz, Eric Granger, and Ismail Ben Ayed. Laplacian regularized few-shot learning. In *International Conference on Machine Learning*, pages 11660–11670. PMLR, 2020.

