Intrinsic image properties from deep learning

Victor Moyano Diaz

Resum– Són molts els algoritmes de visió per computador que no poden funcionar correctament quan les condicions d'il·luminació a les quals es troben exposades les imatges varien de forma substancial. Per aquest motiu, s'utilitza un model que proposa separar una imatge qualsevol en dues imatges diferents: Shading i Reflectància. El shading conté la informació dels il·luminants en qualsevol imatge, i la reflectància conté informació de les propietats físiques dels objectes. L'objectiu d'aquest treball ha sigut utilitzar deep learning per realitzar aquesta separació. Diferents arquitectures de deep learning han estat entrenades amb diferents bases de dades d'imatges sintètiques, que s'han anat confeccionant a mesura que aquest treball progressava. Per últim, es mostren els resultats obtinguts amb cadascuna de les arquitectures provades i utilitzant les diferents bases de dades generades.

Paraules clau– Imatges intrínseques, Deep learning, Reflectància, Shading, Unet, Segnet, Detecció d'objectes, Visió per computador, Intel·ligència artificial, Xarxes neuronals.

Abstract– A lot of computer vision algorithms have trouble performing as the should due the lighting conditions the images are exposed to. When the lighting conditions vary substantially, these algorithms are just not able to perform nicely. For this reason, other authors have proposed a model that separates any image into two different images: Shading and Reflectance. The shading contains information about the lighting conditions of the scene. On the other hand, the reflectance contains the physical characteristics of the objects. The goal of this work is to use deep learning to do this separation. We have used different deep learning architectures with different databases build as this work progressed. At the end of this paper we present the results obtained with each architecture tested and each database generated.

Keywords– Intrinsic images, Deep learning, Reflectance, Shading, Unet, Segnet, Object detection, Computer vision, Artificial intelligence, Neural networks.

1 Introducció

La detecció d'objectes és un punt clau per resoldre diferents problemes que actualment els enginyers a tot el món estan intentant solucionar. La conducció autònoma, videovigilancia o aplicacions que fan servir realitat augmentada són alguns dels casos on la detecció d'objectes i la visió per computador presenten un paper crític pel funcionament correcte dels algoritmes proposats. No obstant, el resultat que s'obté al capturar una imatge amb la càmera pot variar molt en funció de les condicions d'il·uminació d'aquell instant. Dues imatges capturades amb els mateixos objectes però amb il·luminacions diferents, poden semblar completament diferents, tal com podem veure a la figura (1). La il·luminació en la imatge de la esquerra es suficient per poder distingir 3 objectes clarament, mentre que a la imatge de la dreta ja és més difícil distingir els 3 objectes i la forma de cadascun.

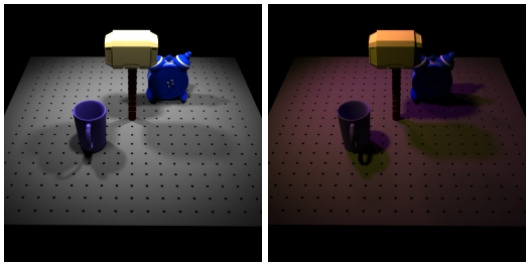


Fig. 1: Mateixa escena amb condicions de iluminació diferents

És per aquest motiu que s'utilitza un model d'imatge

• E-mail de contacte: victormoyano.d@gmail.com
• Menció realitzada: Computació
• Treball tutoritzat per: Ramón Baldrich (Computació)
• Curs 2017/2018

[1] que s'encarrega de separar els efectes que provoca la il·luminació. Aquest model estipula que qualsevol imatge pot ser separada en dos imatges diferents: Reflectància i Shading, segons la equació (1).

$$I = R * S \tag{1}$$

Com veiem, dividim qualsevol imatge en dos imatges diferents, que si les multipliquem píxel a píxel, ens retornen la imatge original. Aquestes dues imatges reben el nom de **imatges intrínseques**, i contenen propietats molt característiques i interessants de l'escena:

- Reflectància: Conté informació de les propietats físiques dels objectes. El color i forma del objecte degut al material del qual està construït es veurà reflectit en aquesta imatge.
- Shading: Conté informació de la il·luminació de la escena. Colors i intensitat de les llums, i les ombres produïdes per aquestes, les podrem veure en aquesta imatge.

La intenció és separar la il·luminació de la escena de la forma i color dels objectes. A continuació, en les figures (2) i (3) il·lustrem la separació de les dues imatges de la figura (1) en reflectància i shading.

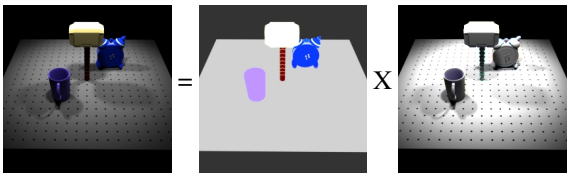


Fig. 2: Escena amb bona il·luminació

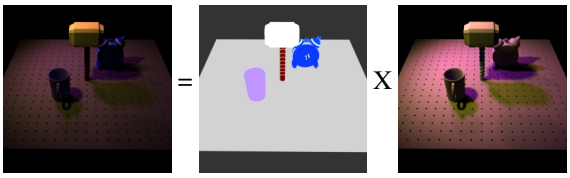


Fig. 3: Escena amb una il·luminació deficient

Com podem veure, la reflectància en la figura (3) i la figura (2) són exactament iguals. Això és degut a que ambdues imatges contenen exactament els mateixos objectes, en la mateixa posició i amb els mateixos materials. L'únic diferent en la escena és la intensitat i el color de les llums i, per tant, té sentit que la única imatge que canvia sigui el shading.

Doncs, la utilitat de separar reflectància i shading queda demostrada. Per exemple, qualsevol algoritme de detecció d'objectes funcionarà millor fent servir la imatge de reflectància, que la imatge original, degut a que diferents condicions d'il·luminació no faran variar la reflectància.

Ara que la utilitat de separar una imatge qualsevol en reflectància i shading queda explicada, podem passar a definir els objectius d'aquest treball en concret.

1.1 Objectius

Per clarificar quin és l'objectiu principal d'aquest treball, començarem dient que **aquest treball no pretén crear una nova arquitectura de deep learning**, ni proposar una innovació molt gran en els mètodes utilitzats per entrenar xarxes neuronals fent servir imatges. Ja n'hi han diverses arquitectures que es conegut que funcionen bé per determinats problemes, així que no pretenem crear un tipus d'arquitectura completament nova.

El objectiu principal d'aquest treball és **determinar quines dades són més adients per tal d'entrenar un model que sigui capaç de separar reflectància i shading en una imatge qualsevol**. Les arquitectures són conegudes, el que no està clar és amb quin tipus de dades el model aprendrà millor. De fet, la recerca en els últims anys ha estat molt enfocada en millorar i proposar noves arquitectures (cada cop més complicades) pensant que el problema és que l'arquitectura podria ser millor. No s'ha posat tant èmfasi en crear una base de dades que generalitzi i representi bé les diferents situacions possibles. **Aquest treball pretén demostrar que les dades utilitzades són tant importants com l'arquitectura escollida**. Degut al caràcter d'aquest treball i que haurem de treballar tant en l'arquitectura escollida com en les dades utilitzades, queden definits dos objectius ben diferenciats:

- **Predir la reflectància de qualsevol imatge fent servir deep learning**: L'objectiu és confeccionar un software que sigui capaç d'extreure imatges de reflectància a partir d'una imatge qualsevol. Hem escollit dues arquitectures que han donat bons resultats en els últims anys (Segnet i U-net). Provarem aquestes dues arquitectures amb les bases de dades generades i escollirem la que funcioni millor pel nostre problema determinat.
- **Generar les bases de dades sintètiques necessàries**: Per tal de poder entrenar arquitectures de deep learning, necessitem molts exemples d'imatges originals amb les seves corresponents reflectàncies. La forma de generar aquests 'datasets' ha estat utilitzant Blender per generar tantes imatges com siguin necessàries. De fet, s'han generat dos 'datasets' diferents, el primer d'ells amb 18000 imatges i el segon amb 24000.

Com podem veure, n'hi han dos camps ben diferenciats, però per poder realitzar aquest treball correctament, ambdós s'han de dur a terme. Per assolir un dels objectius s'hauran de tenir coneixements en intel·ligència artificial i xarxes neuronals, i per assolir l'altre objectiu es necessitaran nocions en disseny 3D i generació d'imatges sintètiques.

1.2 Estat de l'art

El concepte de imatges intrínseques va ser introduït per primera vegada per [10]. Els mètodes convencionals per obtenir imatges intrínseques estan basades en la teoria Retinex, tractada en [11], [12], [13]. En aquesta teoria, s'associen gradients grans en la imatge a canvis en la reflectància, i gradients petits al shading.

Més recentment, la descomposició d'imatges en imatges intrínseques s'ha intentat fer mitjançant CNNs, com farem en aquest mateix treball. Alguns dels articles que ho han fet han sigut [14], [15], [16]. Per exemple, a [16] van utilitzar una CNN que directament aprenia a predir reflectància

i shading utilitzant una imatge RGB, que és el que farem nosaltres en el nostre treball.

Respecte les arquitectures utilitzades, s'han anat proposant diverses en els últims anys, ja que les CNNs són un camp on molts investigadors es troben treballant actualment. Per exemple, a [2] (2017) proposen l'arquitectura Segnet, que és un tipus de CNN que van utilitzar per fer detecció d'objectes en imatges. Dos anys abans (2015, a [3] van proposar U-Net, una CNN utilitzada per segmentació d'imatges en biomedicina. Aquestes són les dues arquitectures que nosaltres provarem en el nostre treball.

2 METODOLOGIA

Es pretén entrenar un model que aprengui a extreure la reflectància de qualsevol imatge fent servir deep learning i es poden utilitzar diferents llenguatges de programació i llibreries per fer-ho. En quant a llenguatge de programació s'ha utilitzat Python 3 ja que és un llenguatge de programació amb el que ja estava familiaritzat i les llibreries de deep learning per python s'actualitzen constantment i tenen bon suport de la comunitat investigadora. Respecte quina llibreria de deep learning utilitzar, s'ha escollit entre aquestes: TensorFlow, Theano, PyTorch, Keras + Theano, Keras + TensorFlow i Caffe.

S'ha escollit treballar amb tensorflow per sota de keras, degut a que keras ens dona un nivell d'abstracció que simplifica molt la construcció dels models (encara que degut a això, perdrem llibertat i nivells de customització). Programar xarxes neuronals amb Tensorflow o Theano és més complicat ja que hem de controlar més a baix nivell les capes de la xarxa neuronal. El motiu pel qual s'ha escollit TensorFlow per sota de Keras en lloc de Theano, és que sembla que Theano serà discontinuat en el futur recent i, per tant, s'ha aprofitat per aprendre TensorFlow que sembla que seguirà sent actualitzat en un futur recent.

Les arquitectures utilitzades en deep learning solen tenir moltes capes amb moltes neurones i , per tant, necessitem un hardware amb una bona capacitat. Un servidor amb varies gràfiques NVIDIA de l'estat de l'art(NVIDIA GTX 1080 Ti amb 11 GB) serà utilitzat, de forma que no hauríem de tenir problemes per potencia de hardware a l'hora d'entrenar els nostres models.

Per altra banda, degut a que volem utilitzar deep learning necessitarem una quantitat de dades important per tal de que el model pugui aprendre correctament. Per això, la base de dades utilitzada és tant important com l'arquitectura utilitzada. Si la base de dades és prou general i conté informació representativa del problema, el model aprendrà característiques interessants, però si les dades no són representatives, el model no estarà aprenent res important. Per aquest motiu, en aquest treball hem generat dos bases de dades diferents:

- Una base de dades sintètica, de 18000 imatges, simple i similar a la que es troba generada en el paper [2]. Aquesta base de dades serà generada amb Blender, fent servir molts tipus d'objectes diferents amb 'environment maps' (imatges de fons) diferents. **Aquesta base de dades no és fàcilment generable en el món real**, degut a que ha estat confeccionada fent servir milers d'objectes diferents amb imatges de fons aleatòries. Els objectes utilitzats són provinents de la base de dades Shapenet ([8]). Es poden veure imatges representatives d'aquesta base de dades en la figura 4

- Altre base de dades sintètica, també generada fent servir Blender. Aquesta base de dades conté 24000 imatges, però en aquest cas sí que ens assegurarem de que **aquesta base de dades pot ser replicada en el món real**. Hem reduït el número d'objectes utilitzats a 8 i hem replicat en el món sintètic una **plataforma de generació d'imatges** present al Centre de Visió per Computador.

  Aquesta plataforma consisteix d'una taula que pot rotar 360 graus, i pot inclinar-se fins 45 graus. Tenim 9 llums de les quals podem canviar el color i la intensitat, assegurant que podem tenir molts tipus diferents d'il·luminants. La càmera i les llums es troben fixes a la estructura, de forma que l'únic que es mou és la plataforma amb els objectes. Els objectes es posen fixats als forats de la plataforma, i la plataforma va rotant de forma que podem capturar diferents punts de vista dels objectes. D'aquesta forma, ens assegurem que aquesta base de dades sí que pot ser replicada en el món real.

  S'han fet servir 20 configuracions d'objectes diferents, on ens hem assegurat de combinar les posicions dels objectes de forma que les ombres d'un estiguin a sobre d'un altre objecte, per tal de que la xarxa pugui aprendre configuracions complicades. Respecte les llums, tenim 6 configuracions diferents, on variem les llums que es troben enceses en cada cas. A més, per cada una d'aquestes 6 configuracions, ens assegurarem de canviar el color de les llums a vermell, verd, blau i un color aleatori, de forma que a la base de dades tindrem il·luminants molt variats. Es poden veure imatges representatives de la base de dades resultant en la figura 6.

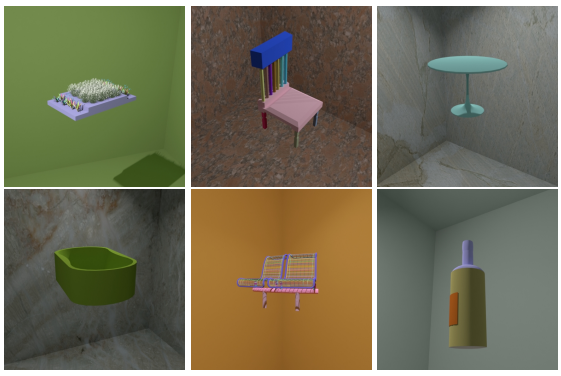


Fig. 4: Imatges de la primera base de dades (No és fàcilment replicable al món real

En principi la primera base de dades (Fig. 4) és més variada en quant a objectes i colors i això sempre ajuda quan estem entrenant una arquitectura de deep learning, ja que podem assegurar que pel nostre model passaran moltes imatges diferents i, per tant, aprendrà característiques interessants. No obstant, no la podem replicar en la realitat i no podrem tenir en compte fenòmens físics presents en

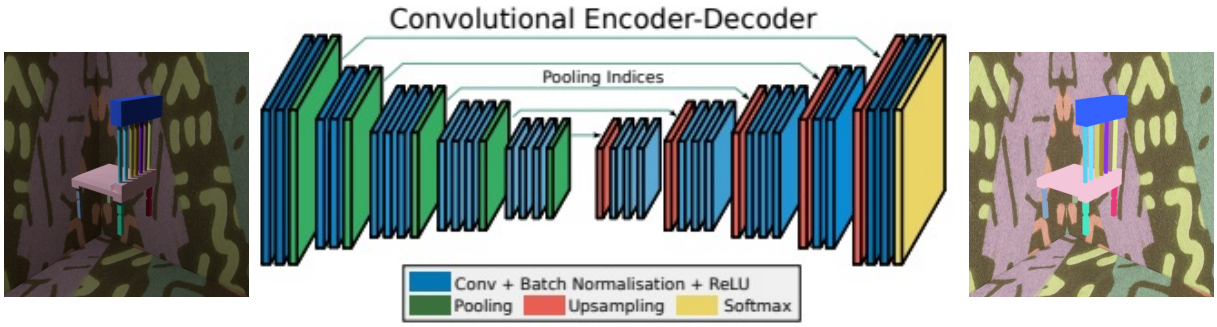


Fig. 5: Esquemàtic de l'arquitectura segnet

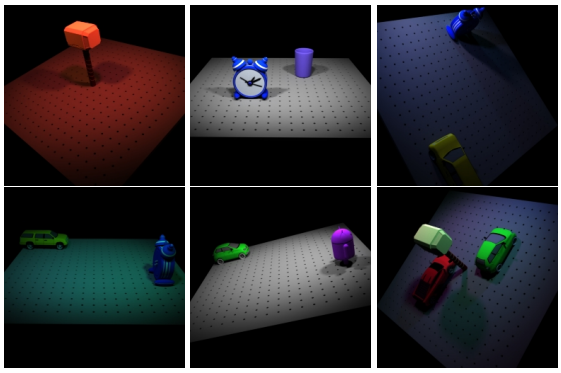


Fig. 6: Imatges de la segona base de dades (Replicable al món real)

la realitat però no en el món sintètic (per exemple, efecte Fresnel).

Doncs, es disposen dues bases de dades diferents amb les quals entrenarem les arquitectures de deep learning. A continuació s'expliquen les dues arquitectures de deep learning utilitzades en l'ordre cronològic que es van provar.

2.1 Segnet [2]

Aquesta arquitectura va ser proposada a l'any 2017. Podem veure un esquemàtic de la xarxa a la figura 5. La imatge del esquemàtic s'ha agafat del article original [2]. És una arquitectura que està formada per capes de convolucions, normalització, activació, 'Pooling' i 'Upsampling'. A continuació expliquem la funció de cadascuna:

- **Convolució**: S'encarrega de avaluar una determinada zona de píxels (normalment 3x3) i d'aquesta forma obtenim nous valors, mitjançant la combinació d'un píxel i els píxels veïns.
- **Normalització**: S'encarrega de que els valors de la imatge es trobin sempre normalitzats, és a dir, en el rang [0,1]
- **Relu**: Decideix quines neurones s'han d'activar i quines no.
- **'Pooling'**: Aquestes capes són les encarregades de reduir la mida de la imatge quan sigui necessari. D'aquesta forma compactem la informació en imatges cada cop més petites.
- **'Upsampling'**: Amb aquestes capes tornem a augmentar la imatge. D'aquesta forma passem de informació compactada(imatges petites) a informació més difusa(imatges grans).

Bàsicament, la idea d'aquesta arquitectura és compactar la informació de la imatge fent servir convolucions i poolings, i després tornar a fer gran la imatge. És important tenir en compte que en el article original la van fer servir per detecció d'objectes i, per tant, la última capa de la xarxa neuronal és una capa softmax que retorna una classe detectada (en realitat retorna una llista de probabilitats de detecció de les diferents classes). En el nostre cas, no estem realitzant cap classificació, així que la capa softmax ha estat eliminada.

2.2 Unet [3]

L'altre arquitectura de deep learning que hem utilitzat en aquest treball ha sigut U-net. El article original va ser publicat al 2015. Les capes utilitzades són les següents:

- **Convolució**: També utilitza capes de convolució com Segnet.
- **Relu**: Exactament igual que Segnet.
- **'Pooling'**: Igual que Segnet. Amb aquestes capes anem fent la imatge més petita.
- **'Copy and crop'**: Amb aquestes capes copiem la meitat d'una imatge que trobem en una capa, i aquesta meitat serà utilitzada posteriorment per augmentar la mida de la imatge més endavant.

Com podem veure ara quan hem d'ampliar la imatge, utilitzem porcions de les imatges obtingudes en les capes anteriors. És a dir, en lloc de d'ampliar una imatge extrapolant valors pels píxels que estem creant, utilitzarem els píxels de la imatge que vam obtenir en la capa oposada. Aquesta arquitectura és pot veure representada a la figura (7). En aquest cas, hem modificat lleugerament l'arquitectura proposada en l'article original, ja que l'arquitectura original era molt gran i no cabia a la memòria ram de la gràfica proporcionada.

3 Resultats

El primer que hem de respondre és: Quina arquitectura funciona millor pel nostre problema? Per respondre a aquesta

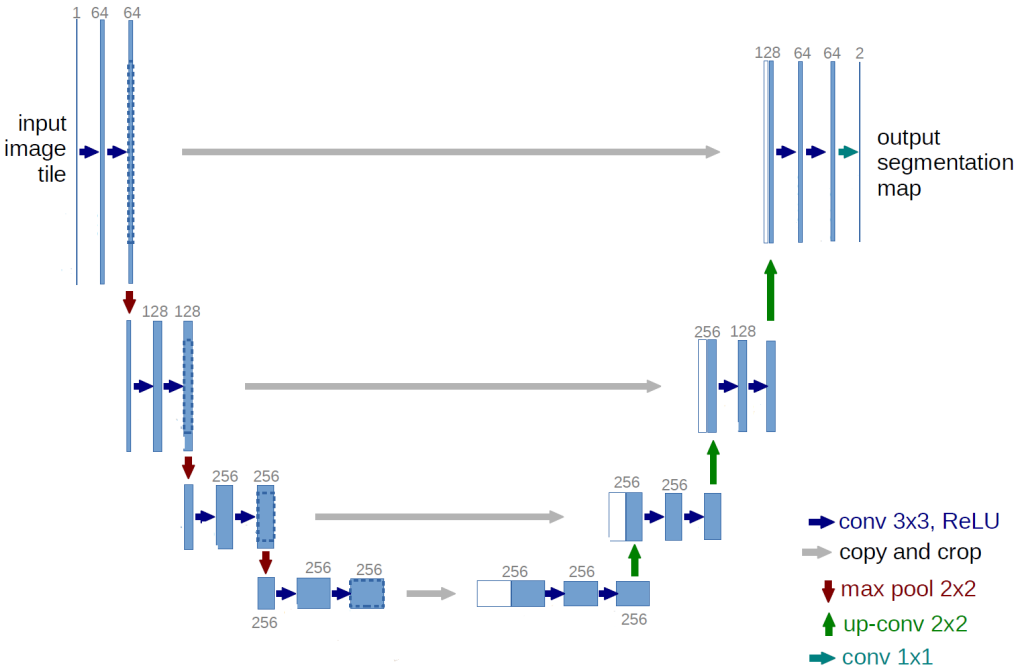
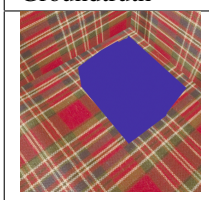
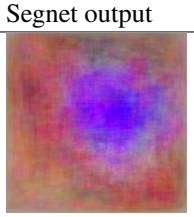
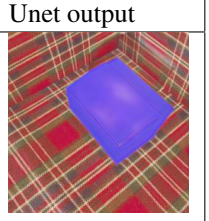
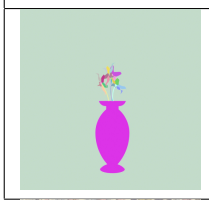
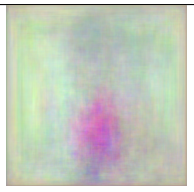
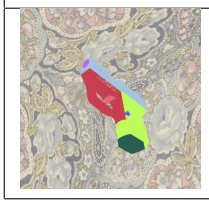
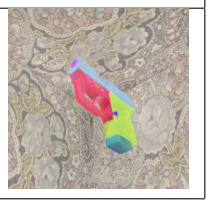


Fig. 7: Esquemàtic de l'arquitectura Unet

pregunta, vam entrenar cadascuna de les arquitectures fent servir la primera base de dades confeccionada (Fig. 4) i es van comparar els resultats obtinguts. Ambdues arquitectures es van entrenar per un total de 20 èpoques i els resultats obtinguts van ser els que es poden veure a la taula 1.

Groundtruth	Segnet output	Unet output
		
		
		

TAULA 1: RESULTATS DE SEGNET VS UNET

Com podem veure, tot i que Segnet sembla que prediu d'una forma aproximada la reflectància de la imatge, U-net presenta una resolució i definició molt més elevada. Aquesta resolució de U-net és degut a que quan la imatge s'està fent gran de nou per recuperar la mida original, s'utilitzen trossos de les imatges obtingudes en les capes anteriors. En canvi, en la xarxa Segnet simplement el que fem és extrapolar els valors del píxels, i per aquest motiu veiem aquestes taques difuses a les prediccions. De fet, Segnet va ser pensada per problemes de classificació d'objectes, on la resolució de la imatge no importa tant com en el nostre cas. Però, encara podem anar més enllà. Ara que hem vist que Unet clarament s'adapta molt millor al nostre problema, ens fixarem en si realment la xarxa neuronal està aprenent a separar la ombra (shading) i la reflectància en una imatge qualsevol.

Abans de continuar, és important comentar que totes els resultats presentats en aquest apartat s'han obtingut fent servir imatges de 'testing'. És a dir, que una petita part de les bases de dades d'imatges no es van utilitzar per entrenar el model i es van reservar per fer 'testing'. D'aquesta forma, comprovem que el model no està fent overfitting del conjunt d'entrenament.

En la figura 8 podem veure clarament com la xarxa neuronal està aprenent a eliminar les ombres de la imatge original, al menys amb imatges de la mateixa base de dades. A la primera i segona columna, al principi la ombra és clarament present, mentre que a mesura que augmenten les èpoques la xarxa neuronal ja ha aprés a eliminar la ombra correctament. Les dues primeres columnes representen les prediccions de imatges de la mateixa base de dades i la tercera columna representa la predicció en una imatge de l'altra base de dades. Quan aquest model intenta predir imatges de l'altra base de dades, podem veure com ho fa molt malament. El més important és que no ha aprés a eliminar la ombra, com veiem a la tercera columna d'imatges de la figura 8. A més, tampoc ha aprés a diferenciar el color dels objectes respecte del color de la llum. Per aquest motiu, podem afirmar que la base de dades 1 no és suficient per fer que el nostre model aprengui, segurament degut a que ha estat confeccionada només amb 1 tipus de il·luminant.

Ara que hem comprovat que Unet pot aprendre a separar correctament shading i reflectància, hem provat a entrenar un altre cop utilitzant la mateixa arquitectura, però fent ser-

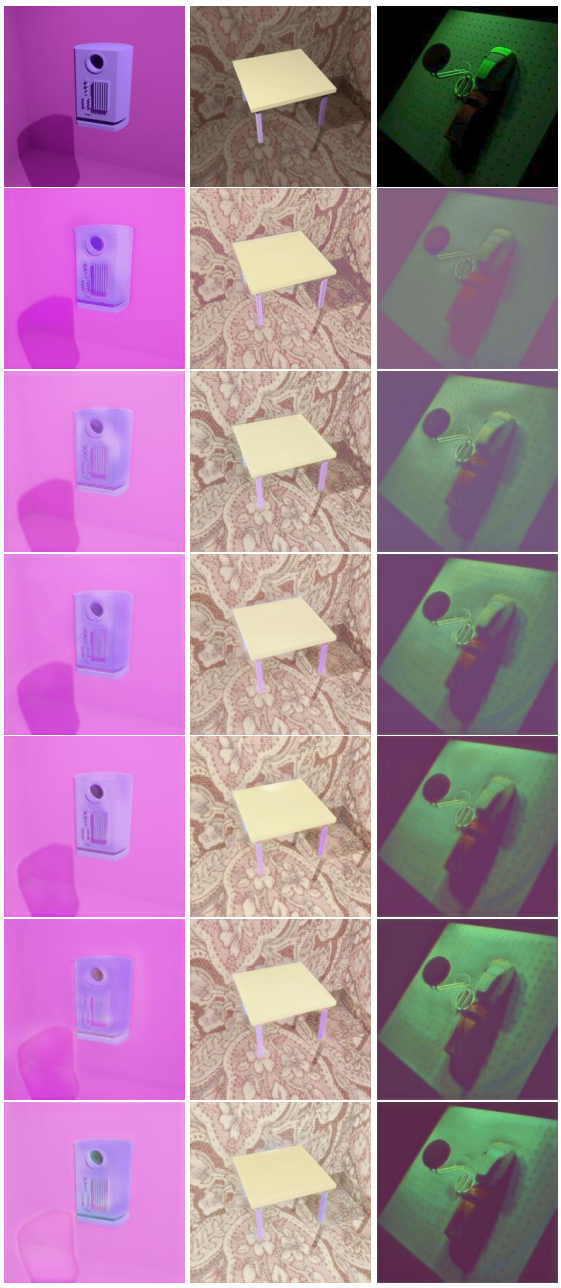


Fig. 8: Aprenentatge qualitatiu de la xarxa Unet amb la primera base de dades a mesura que passen les èpoques. El ordre cronològic augmenta d'adalt abaix. Les dues primeres columnes corresponen a imatges de la mateixa base de dades, mentre que la tercera columna correspon a una imatge de l'altre base de dades. La primera fila correspon a la imatge original.

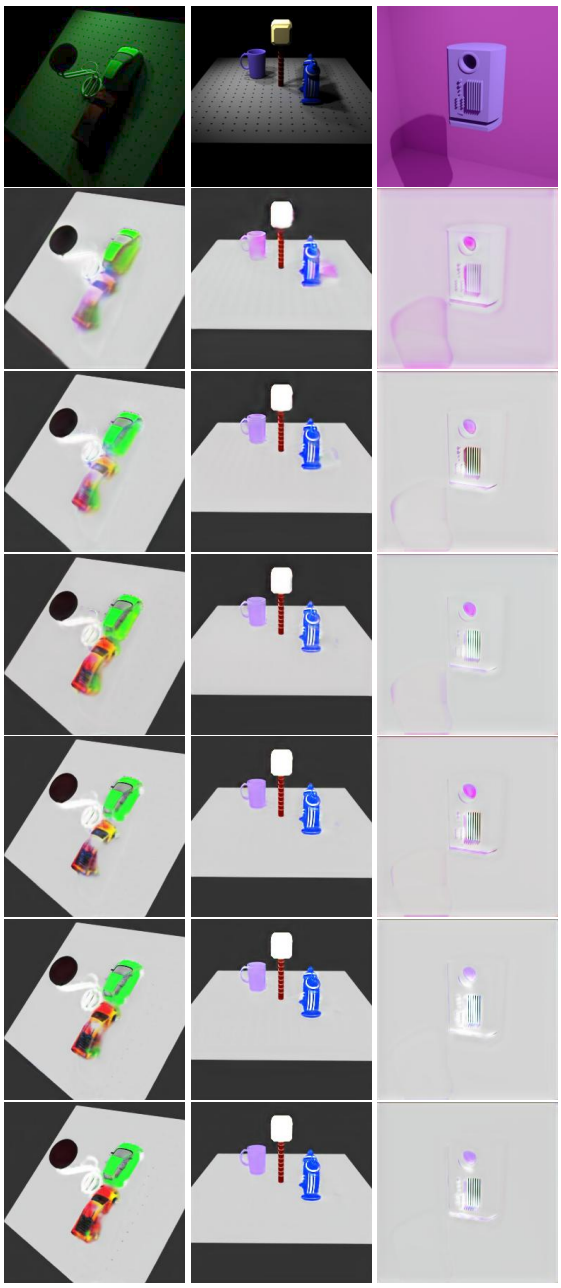


Fig. 9: Aprenentatge qualitatiu de la xarxa Unet amb la base de dades realista a mesura que passen les èpoques. El ordre cronològic augmenta d'adalt abaix. Les dues primeres columnes corresponen a imatges de la mateixa base de dades, mentre que la tercera columna correspon a una imatge de l'altre base de dades. La primera fila correspon a la imatge original.

vir l'altre base de dades que hem confeccionat, que conté diversos tipus d'il·luminants. Aquesta nova base de dades no és tan ideal com la primera (infinitat d'objectes backgrounds diferents), però s'acosta més a una base de dades que podem confeccionar en el món real.

A la figura 10 podem veure el procés d'entrenament de la xarxa neuronal U-net fent servir la nova base de dades. Com podem veure en el gràfic, la majoria d'aprenentatge es fa en les primeres èpoques, encara que a mesura que van augmentant el número de èpoques el MSE (mean squared error) va disminuint poc a poc. Aquesta és una senyal de que la nostra xarxa neuronal està aprenent prou bé.

Ara, passem a analitzar els resultats qualitatius. De la mateixa forma que amb la primera base de dades, primer mirarem si la mateixa xarxa està aprenent a eliminar shading de la imatge original. A la figura 9 podem veure els resultats obtinguts. Les dues primeres columnes corresponen a imatges de la mateixa base de dades, mentre que la tercera columna correspon a una imatge de l'altre base de dades. **Els resultats són molt satisfactoris**. És impressionant com **en imatges fosques**, on a simple vista tenim dificultats per distingir el color original i la mida dels objectes degut a una il·luminació deficient, **el model és capaç de separar shading i reflectància correctament**.

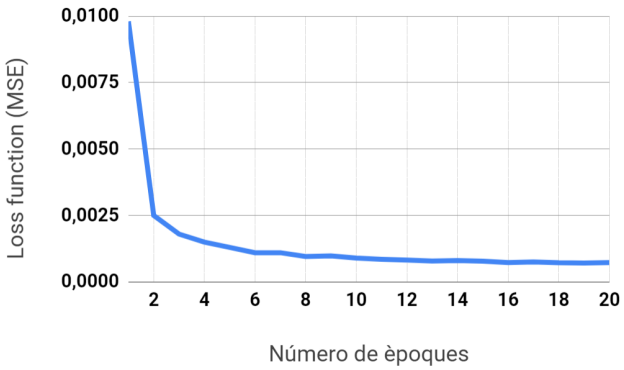


Fig. 10: Aprenentatge quantitatiu a mesura que passen les èpoques utilitzant l'arquitectura Unet amb la base de dades realista

Per exemple, en el cas de la primera columna (imatge verda), a la imatge original gairebé no podem apreciar el cotxe vermell. En canvi, podem veure com a mesura que augmenten les èpoques el nostre model aprèn a detectar el cotxe vermell correctament. A més, també es pot veure clarament en les columnes 1 i 2, que elimina tot efecte de shading present a la imatge original.

No obstant, com es pot veure a la tercera columna de la figura 9, les imatges de l'altre base de dades presenten un problema pel nostre model. Sí que és cert que en aquest cas eliminem els efectes de shading (veiem que no n'hi ha cap ombra en la tercera columna), però també s'elimina molt color de la reflectància. És a dir, el fons de les imatges de la tercera columna hauria de ser púrpura, com a la imatge original, però la nostra xarxa neuronal prediu que és gris.

Atribuïm aquest error a **l'absència de diferents colors de fons** en les imatges del nou dataset, degut a que a les imatges sempre és present un color gris que prové de la plataforma rotatòria amb la qual s'agafen els objectes. Aquest color gris s'hauria de substituir per color aleatoris o, al menys, diferents colors, per tal de que la xarxa no aprengui que el color de fons de la reflectància és sempre de color gris.

4 Conclusions

Al llarg d'aquest treball hem pogut familiaritzar-nos amb el ús de CNNs per extreure característiques interessants d'imatges. Hem aprés com entrenar diferents CNNs amb diferents bases de dades i s'han avaluat els resultats obtinguts.

La conclusió principal irrefutable que podem extreure és que **les dades utilitzades per entrenar qualsevol xarxa neuronal són tan importants (o fins i tot més) que l'arquitectura escollida**. Hem pogut veure com exactament el mateix model entrenat amb una base de dades o una altre, era capaç o no d'eliminar shading de les imatges. S'ha de recordar que un dels objectius principals del treball era precisament això: Demostrar que les dades són tan importants com el que fem amb elles. Doncs, un dels objectius principals d'aquest treball queda clarament assolit.

Per altra banda, altre dels objectius d'aquest treball era determinar quin tipus de base de dades seria necessària per entrenar un model capaç de predir reflectància i shading d'una imatge. A més, també volíem saber quina arquitectura funcionaria millor pel nostre problema: Segnet o Unet. Com s'ha demostrat, sens dubte, **Unet és l'arquitectura a utilitzar**, ja que presenta uns resultats molt més definits que Segnet. Això és degut a que Segnet va ser utilitzada per fer 'tagging' d'objectes en imatges, on no es necessita bona definició en la imatge de sortida.

La forma d'intentar saber quin tipus de base de dades és la ideal ha sigut generar dos tipus de bases de dades diferents: una amb un únic il·luminant però amb molts objectes i imatges de fons aleatòries i l'altre amb diversos il·luminants però amb pocs objectes i un sistema físic real (sense diferents imatges de fons). Hem vist com entrenant l'arquitectura Unet amb la primera base de dades, no érem capaços de predir bé imatges de la segona base de dades. És a dir, l'arquitectura no havia aprés a separar el shading de la reflectància. Per altre banda, si utilitzàvem la segona base de dades per entrenar, sí que es separava el shading de la reflectància en imatges que no són provinents de la base de dades utilitzada per entrenar. No obstant, els diferents colors de fons no es podien estimar a la reflectància, degut a que precisament aquesta base de dades no tenia diversos colors de fons degut a que és un sistema físic real.

Doncs, quina base de dades és la més adient per entrar un model d'aquest tipus? Tot sembla indicar que **la base de dades ideal seria una mescla de les dues generades**. Es necessita una base de dades amb molts il·luminants diferents, però assegurant-nos que no n'hi ha cap element físic que sigui sempre igual a les imatges. En aquest cas, tenim la taula gris a sobre de la qual es troben els objectes, que sempre és de color gris pel fet de que és una taula real i, per tant, la xarxa aprèn que el fons (la part que no és objecte) sempre és de color gris. Això ha quedat demostrat clarament quan hem testejat la base de dades realista amb imatges de l'altre base de dades. Com a continuació d'aquest treball, quedaria pendent pensar com introduir imatges de fons aleatòries en un sistema d'adquisició real com el que disposen al Centre de Visió per Computador i, llavors, tornar a entrenar l'arquitectura U-net per veure si generalitza bé per les dues bases de dades.

En resum, creiem que s'han assolit tots els objectius desitjats amb aquest treball i els resultats obtinguts han estat molt satisfactoris.

Agraïments

Sens dubte, aquest treball no es podria haver dut a terme sense la interminable paciència i l'ajut del grup d'investigació 'Colour in Context', que es troba al Centre de Visió per Computador a la UAB. Especialment 2 integrants: Ramon Baldrich, tutor d'aquest treball, per transmetre molts dels seus coneixements en deep learning i visió per computador, i Hassan Ahmed, estudiant de PhD, que amb una paciència infinita m'ha ajudat a entendre les imatges intrínseques i la generació d'aquestes en bases de dades sintètiques.

REFERÈNCIES

[1] Tappen, M. F. (2002). Recovering shading and reflectance from a single image (Doctoral dissertation, Massachusetts Institute of Technology).

[2] Badrinarayanan, V., Kendall, A., Cipolla, R. (2017). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE transactions on pattern analysis and machine intelligence, 39(12), 2481-2495.

[3] Ronneberger, O., Fischer, P. Brox, T. (2015, October). U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention (pp. 234-241). Springer, Cham.

[4] Matt Bell and William T Freeman. Learning local evidence for shading and reflection. In Proceedings International Conference on Computer Vision, 2001.

[5] Tappen, M. F. (2002). Recovering shading and reflectance from a single image (Doctoral dissertation, Massachusetts Institute of Technology).

[6] Badrinarayanan, V., Handa, A., Cipolla, R. (2015). Segnet: A deep convolutional encoder-decoder architecture for robust semantic pixel-wise labelling.

[7] Adelson, E. H., Pentland, A. P. (1996). The perception of shading and reflectance. Perception as Bayesian inference, 409-423.

[8] Chang, A. X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., ... Xiao, J. (2015). Shapenet: An information-rich 3d model repository.

[9] Kim, S., Park, K., Sohn, K., Lin, S. (2016, October). Unified depth prediction and intrinsic image decomposition from a single image via joint convolutional neural fields. In European conference on computer vision (pp. 143-159). Springer, Cham.

[10] Barrow, H., Tenenbaum, J. (1978). Recovering intrinsic scene characteristics. Comput. Vis. Syst., A Hanson, E. Riseman (Eds.), 3-26.

[11] Land, E. H., McCann, J. J. (1971). Lightness and retinex theory. Josa, 61(1), 1-11.

[12] Shen, L., Tan, P., Lin, S. (2008, June). Intrinsic image decomposition with non-local texture cues. In Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on (pp. 1-7). IEEE.

[13] Zhao, Q., Tan, P., Dai, Q., Shen, L., Wu, E., Lin, S. (2012). A closed-form solution to retinex with nonlocal texture constraints. IEEE transactions on pattern analysis and machine intelligence, 34(7), 1437-1444.

[14] Shelhamer, E., Barron, J. T., Darrell, T. (2015). Scene intrinsics and depth from a single image. In Proceedings of the IEEE International Conference on Computer Vision Workshops (pp. 37-44).

[15] Zhou, T., Krahenbuhl, P., Efros, A. A. (2015). Learning data-driven reflectance priors for intrinsic image decomposition. In Proceedings of the IEEE International Conference on Computer Vision (pp. 3469-3477).

[16] Narihira, T., Maire, M., Yu, S. X. (2015). Direct intrinsics: Learning albedo-shading decomposition by convolutional regression. In Proceedings of the IEEE international conference on computer vision (pp. 2992-2992).