HELLENIC
MEDITERRANEAN
UNIVERSITY

Semantic Segmentation of Diabetic Retinopathy Lesions, using a UNET with Pretrained Encoder.

Dimitrios Theodoropoulos
B.A, Department of Physics, University of Crete, 2003

Submitted as partial fulfilment of the
requirements for the degree of
Master of Science

Supervisor: Marias Kostas

Department of Electrical and Computer Engineering
School of Engineering

Heraklion, Crete

November 7, 2021

Abstract

With the emergence of Deep Learning nowadays, a lot of novel architectures have been devised to perform tasks such as classification, detection, segmentation etc. In Medical Imaging and especially in Ophthalmology, the robustness of Deep Learning is exploited in many studies. Most of the state of art papers nowadays use UNets or Fully Convolutional Networks (FCN) for segmentation tasks. UNets instantiate a modified version of the acknowledged Convolutional Neural Networks. In this study we focus on the segmentation of Diabetic retinopathy lesions. In the real world, this task is very difficult because a good algorithm is based on a robust dataset. The special annotated datasets for segmentation tasks are pretty rare and comprise fewer images compared to other larger datasets. Additionally, images dedicated to segmentation are imbalanced as far as the ratio of lesions and normal pixels are concerned. Those were the main reasons we chose not to train a model from scratch and confront possible difficulties. Instead we exploited transfer learning and utilized a pretrained network (MobileNetV2) as encoder of the Unet. We segmented four kinds of Diabetic Retinopathy lesions, surpassing the existing state of art models in the case of two lesions: Hemorrhages and Soft Exudates. More specifically, Sensitivity in Hemorrhages reached 0.89 whilst in Soft Exudates reached 0.97. One of the novelties of the thesis is that the algorithm could be further easily applied on mobile phones, something that MobileNetV2 is intended to.

Acknowledgements

First of all I would like to thank the person who gave me the chance to change my life and fulfill my dream, my professor Kostas Marias. He supports me in every way.

Furthermore, I would like to thank George Manikis and Nikos Tsiknankis for the amazing collaboration we had.

Finally I would like to thank my wife for her support and my two adorable daughters for giving me motivation for all my effort.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Objective of the study

The revolution of Artificial Intelligence (AI) is phenomenal nowadays. We witness the utilities and the implementations of AI without really knowing it. Facebook, Instagram, etc. for example apply AI. Many scientific disciplines exploit this novelty and experimentally try to gain intuition about AI 's behavior on several tasks. On the other hand, there are open fields in Medicine that can be improved and AI could help. Medical Imaging is a field that asks for those improvements.

AI may not only be applied theoretically but also in practice. There are many underdeveloped countries, in Africa for example, that do not have a decent number of doctors at that time when children suffer and have no economical background to move to the closest hospital. In such countries, and especially in remote areas, the lack of doctors is a determinant factor for Diabetic Retinopathy evolution. Diabetic Retinopathy (DR) is a retinal vascular disease that affects the central vision. There are early signs of Diabetic Retinopathy. Those signs are the lesions which carry information about the DR stage. Thus, if a patient discovers those signs early, has a bigger chance to prevent DR evolution. In our study we segment 4 kinds of lesions: Hard Exudates, Hemorrhages, Microaneurysms and Soft Exudates. The work presented in this thesis has also humanitarian motivations. These related to the fact that in underdeveloped countries, and especially in remote areas, the lack of doctors is an important factor for Diabetic Retinopathy evolution [1]. DL based tools provide a relative novel approach that may substitute doctors in remote and rural areas, in addition to assisting them as decision support systems when diagnosis is very complex and in general "...AI-based systems will augment physicians and are unlikely to replace the traditional physician–patient relationship", as reported in [2]. The final algorithm we implemented can be further applied to a mobile phone, onto which a special fundus camera can be attached as seen in Figure 1.1.

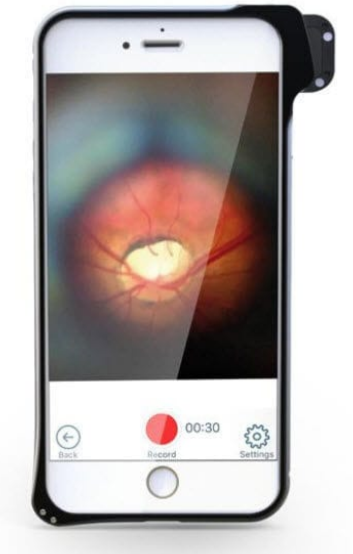


Figure 1.1: A special fundus camera attached on a mobile phone
Image taken from `https://www.d-eyecare.com/en_US/howtouse#assembly`

Several techniques and architectures have been devised to boost the automation of the previous human-based diagnosis. We instantiate an algorithm for Semantic Segmentation [3] and more specifically we use Deep Learning (DL). DL works well with images and is an exclusively supervised algorithm. This means that it needs annotated data. The annotation of those data can only be executed from doctors. Here lies the biggest problem we confronted: the small number of qualitative annotated datasets. The annotation of experts is subjective which means that for the same case, the annotations may differ. Another factor is that this task is fatigue and has complexity level, depending on each case. Annotation is very time consuming and consequently demands a high budget. On the other hand, training a DL network demands a huge database and furthermore images with descend quality. There are datasets with annotations of lesser lesions, fundus images of raw illumination conditions or not precise annotation borderlines. In [4] it was proved that the quality of the dataset influences the algorithm and the performance.

The goal of our study is to semantically segment those four types of lesions. We had to follow a safe way to avoid overfitting issues and generally bad performance. The way we confronted this pitfall was a double plan: a robust DL architecture combined with knowledge transferred from another gnostic domain. The only prerequisite was to have gained the maximum information from the dataset and that was accomplished with a good preprocessing.

UNETs [5] according to our literature review are very powerful in segmentation tasks. So it was a challenge to manage to exploit its power. Transfer Learning is a great solution when dealing with problematic datasets [6]. In our case the datasets comprised very few images and additionally the imbalance of healthy tissue and lesion was very big. So it was a requirement to exploit Transfer Learning and having a better starting point for excellent performance of our model. We did not use a fully trained model but instead we used MobinetV2 [7] as pretrained encoder.

The preprocessing was not based on well known Image Processing techniques ,but instead on gathering the most informative patches of the IDRiD dataset [8]. The size of the patches we worked, was 512x512. We increased the strength of the training by using augmentation and thus creating synthetic patches.

We executed four experiments, each one concerning a binary problem for each lesion. We did not confront any overfitting issues. The experiments showed that the results are pretty decent and surpass the state of art performances for segmenting Hemorrhages and Soft Exudates. More specifically, Sensitivity in Hemorrhages reached 0.89 whilst in Soft Exudates reached 0.97. Generally, both in pixel and lesion level analysis Sensitivity was over 0.83 except for Hemorrhages (lesion level=0.599).

In conclusion, the objective of our study is to use transfer learning to segment retinal lesions and achieve better performance from the existing state of art studies. In addition, for humanitarian purposes we want to contribute to the earlier diagnosis of DR in regions that lack doctors.

Chapter 2

Diabetic Retinopathy lesions — Terminology

2.1 Diabetic Retinopathy

Insulin is secreted from pancreas to regulate blood sugar levels of the body. The disease, either when the pancreas fails to produce a decent amount of insulin, or when insulin is not handled correctly, is named Diabetes Mellitus or Diabetes. There are two types of Diabetes: Type 1 and Type 2. Type 1 is also called "Insulin dependent diabetes" and the reason is the insufficient amount of insulin produced from pancreas. Type 2, is called "non-insulin dependent diabetes" and is due to the ineffective handling of insulin produced in the body. According to WHO in 2000, 2.8% of the total world population suffered from diabetes [4]. Diabetes is a cause of many diseases in the human body: kidney failure, strokes, heart diseases and vision loss.

Diabetic Retinopathy (DR) is a retinal vascular disease that affects the central vision. DR is expected to reach 191 million by 2030 [9]. If the disease evolves without treatment, patients are in danger of becoming blind. Unfortunately people with DR in early stages have no warning signs of their vision. Only when the disease worsens, patients become aware of the situation. Those warning signs that can save many patients from blindness lie in fundus images. Non-mydriatic digital color fundus cameras can acquire such fundus images. Those digital imaging procedures are non invasive and friendly to patients.

2.2 Types of lesions

Fundus means the base of anything. Especially in Medicine, it indicates the inner line of an organ. Sensory Retina, Retinal Pigment Epithelium, Bruch's Membrane and the Choroid form the inner line of the eye, which is the ocular fundus. Such fundus images carry information about the eye, and if any lesions occur inside. Studying existing lesions, leads to conclusions about the disease. Consequently, it is important to identify not only the type of the lesions but also their magnitude. There are 3 main types of lesions: Microaneurysms (MAs), hemorrhages (HMs) and Exudates (EX)s.

Generally Microaneurysms **(MAs)** are the first signs that can be detected from fundus images, indicating DR evolution. **MAs** are a dilation of microvasculature, as a result of disruption of internal elastic lamina. Their size is normally less than 125μm and they look like red spots with distinguishable borders.

When the capillaries collapse, leaking blood forms **HMs**. They look like MAs but they are bigger in size and have random shapes. Splinter hemorrhage occurs in the superficial surface layer and causes more superficial bleeding-shaped flame.

EXs are formed when capillaries collapse and leak much more blood. In contrast with former lesions they are yellowish and have random shapes. EXs comprise of two types: Hard exudates **(HEs)** and soft exudates **(SEs)**. In terms of biology, HEs are proteins and lipoproteins, which escape from abnormal vessels. Their color is close to yellow or white, have distinguishable margins and form blocks or ring-like regions. Central vision depends on macula and fovea. Consequently if the position of hard exudates coincides with those regions, the central vision of the patient is in danger. Detection of the position of hard exudates is very important in DL algorithms. On the other hand, SEs have hues close to white and grey and resemble small clouds. They are the result of the occlusion of the arteriole. Generally MAs and HMs have different brightness from EXs. They are dark whilst EXs are brighter. Neovascularization (NV) are new generated vessels which occur due to the failure to use glucose by existing blood paths. Finally, Macular edema (ME) occurs when leakage of retinal capillaries lies around the macula.

In figure 2.1 we can get an idea about how the lesions look like.

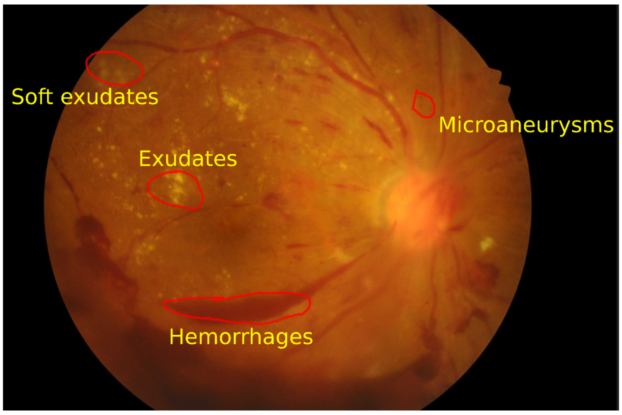


Figure 2.1: An image exhibiting all lesions

2.3 Correlation of lesions with DR

The knowledge of the exact numbers of lesions a patient has, can help in DR grading. MESSIDOR research program [10] grades DR according to the type and number of lesions found in fundus image. Besides lesions, changes in vessel's anatomy or new generated vessels (neovascularizations), can reveal DR. So, in this way segmentation of lesions can implicitly indicate in which stage a patient is. Table 2.1 shows the correlation of DR grading with the number and the type of lesions.

Table 2.1: The correlation of DR Grading with the number and the type of lesions according to MESSIDOR research program

DR Grade	Microaneurysms	Hemorrhages	Neovascularization
0	0	0	0
1	<5	0	0
2	5–15	0–5	0
3	<15	<5	1

Chapter 3

Literature review on segmentation of DR lesions

There are many studies which try to segment different lesions each time, following their own strategies. In this part we will make a literature review, gathering the most representative studies. Table 3.1 summons the main strategies found in this literature review. With the term strategy we imply the methodology that studies tackle the problem of retinal segmentation.

Table 3.1: The main strategies found in this literature review

Strategy	Reference
Selective Sampling (SeS)	[11]
Extraction of probability map	[9, 12, 13, 11, 14, 15]
Class Activation Map (CAM)	[16]
Transfer learning	[17, 18, 19, 20]
Membrane system	[21]
Fixed CNN backbones	[21, 22, 23, 16, 24, 25]
Multiple cohort study	[19]
Fusion of DL and handcrafted features	[14]
Active learning	[25]
DRU-Net	[26]
Adversarial learning	[27, 28]
Additional DR Grading	[29, 30]
GANs as generator of synthetic data	[31, 32]
Attention mechanism	[33, 34, 35]
Ensemble of CNNs	[20]
Reinforcement sample learning	[36]

Softmax as a classifier, provides the benefit to produce for every pixel a probability ranging from zero to one. The original image can be turned into a probability map. In [9, 12, 13, 11, 15] they created such probability maps. In order to proceed with segmentation of the lesions, an appropriate threshold was set (this is necessary in order to turn probability maps into binary images). Post image processing was necessary due to problems generated by this binarization in order to have a fine result with accuracy in the boundaries of the lesions.

Eftekhari et al [15] created patches and trained a CNN to extract probability maps. Afterwards, they passed the thresholded probability map into a CNN to create a smoother probability map. Kushwaha and Balamurugan [29] created two binary maps from U-Nets, instead of CNNs. Each map represented a lesion and was superimposed with the original image. Afterwards they used Resnet-101 to classify the DR grade corresponding to the image.

Another alternative for visualization, besides probability maps, is the Class Activation Map (CAM). This method has a major limitation as far as the architecture concerns: it needs a Global Average Pooling (GAP) as the final layer to work. In case of absence, final Dense Layers are replaced by GAP. Gondal et al [16] created such CAMs. They observed that the removal of the final Dense Layer decreased the overall classification accuracy of the network. The extracted maps had a low resolution while upsampling. They made modifications in order to increase the resolution.

In some studies, architectures were based on fixed CNNs such as AlexNet, Googlenet etc. Perdomo et al [23] trained patches in LeNet CNN to segment the exudates. The model in [24] was based on o_O architecture, which ranked second in the Kaggle Diabetic Retinopathy competition. It comprised two networks, A and B. They also evaluated the model by comparison with AlexNet. The o_O architecture was also used in [16]. Otálora et al [25] segmented exudates using active learning. The CNN which was used was LeNet. Xue et al [21] selected ResNet101 as the classification network in their study.

Harangi et al [20] created an ensemble of three fixed networks: AlexNet, GoogleNet and VGGNet. Fully-connected layers were removed from each independent network and a joint fully connected layer merged them, followed by Softmax. Besides from image-level classification the network localized MAs. They used the weights from the pretrained networks.

Furtado [37] tested three state of art models to segment all lesions of the IDRiD dataset. He experimented with the famous DeepLabV3 model [38], with FCN [39] and with a Unet. The results showed that DeepLabV3 achieved better performance generally. The Microaneurysm's class seems to have the worst performance. He also tested in DeepLabV3 [40] and SegNet [41], two very popular models. DeepLabV3 seemed to segment better but both confused parts of the background as lesions.

Saha et al [22] utilized a FCN to segment all four lesions. Optic Disk was added as a class in the same segmentation problem as lesions, so that the model was able to distinguish exudates from Optic Disk. The proposed network differed from SegNet in essence that in the last layer instead of a pixel wise classification layer (following the architecture of VGG16), had a sigmoid layer to produce class probabilities for each pixel independently in all channels. Exudates had the best performance in contrast with the Microaneurysms.

Contrary to the common approach that Transfer Learning is useful for large datasets, Chudzik et al [17] trained a U-Net with Microaneurysms dataset and fine tuned it with Exudates dataset.

Khojasteh et al [19] executed an experiment with 3 cohorts to segment exudates:

1. they trained a CNN
2. they trained a Discriminative Restricted Boltzmann Machines (DRBM)
3. they used a pre trained residual network and used 3 different classifiers: Support Vector Machine (SVM), Optimum- Path Forest (OPF), and k-Nearest Neighbors (KNN).

The results showed that the best performance was obtained when using ResNet-50 with SVM classifier. The Sensitivity was 0.99.

Another very promising approach was proposed from Xue et al [21] with accuracy 99.7% in detecting lesions of Microaneurysms (IDRiD dataset). They proposed a dynamic membrane system with hybrid structure and implemented efficient CNNs to perform pixel level multitask segmentation.

Orlando et al [14] fused Deep Learning with handcrafted features. Specifically, they trained a CNN to extract features. Moreover they used image processing to extract intensity based and shape based features, resulting in a combined vector. This vector was classified by a Random Forest and a probability map was extracted.

Kou et al [26] proposed a network obtained by combining the deep residual model and recurrent convolutional operations into U-Net. The resulting model was named DRU-Net. The accuracy in the study was 0.9999 and outperformed existing methods such as U-Nets, FCN and ResU-Net. DRU-Net was used for segmenting Microaneurysms.

A trend in the literature review was to exploit the existing limited datasets with the best way. Grinsven et al [11] presented a method to speed up the training process by selecting more informative samples, the Selective Sampling (SeS). A dynamic weight was assigned to each pixel and was updated in each epoch training. The training stopped when there was saturation in updating. The result was integrated with a probability map. The AUC reached was 0.972.

Otálora et al [25] introduced active learning in training a CNN with an algorithm called expected gradient length(EGL) to classify between healthy and exudate patches. They chose to train Le-Net as CNN due to its shallow architecture, in order to prevent convergence issues. EGL was used to select and sort the most informative patches to feed the network.

Budak et al [36] applied Reinforcement Sample Learning for training a Deep Convolutional Neural Network(DCNN). This technique is applied on samples with poor performance in the training procedure. The paper proposed a 3 stage system to detect Microaneurysms. In the first stage they preprocessed the input images. Next followed a selection of candidate lesions based on a spiral like algorithm. Finally, a DCNN was used to train the system.

Generative Adversarial Networks (GANs) [42] have been characterized as "one of the most interesting ideas in the last 10 years in Machine Learning". The rationale behind adversarial learning relies on GANs. A convolutional segmentation network is trained along with an adversarial network, which discriminates segmentation maps coming from the ground truth or from the segmentation network. Gullón [27] in her thesis presented such a system to detect multiple lesions. The loss function was modified and the segmentation of exudates had the best performance compared to the other lesions. Xiao et al [28] utilized Holistically-Nested Edge Detection(HED-Net) for semantic segmentation by incorporating it in an Conditional Generative Adversarial Network (cGAN) to enhance the results. Although HEDNEt was originally proposed to solve edge detection for images, this study showed that it is capable of solving segmentation problems as well. They also modified the loss function for better performance and the results showed that the segmentation of exudates was the most successful among the other lesions.

Zhang et al [33] applied a special technique called Attention Mechanism over a DNN to detect Microaneurysms. Specifically they preprocessed the images and passed them to the DNN which was enhanced with an attention mechanism. Finally a secondary screening was obtained by utilizing the spatial relations of MAs and vessels. The Sensitivity obtained was 0.868.

Si et al [35] utilized a modified FCN with Attention Mechanism to segment Exudates. To address the imbalance of the dataset they invented dice cross entropy loss function. The sensitivity reached 89%

A strategy for tackling limited datasets is creation of synthetic images generated from GANs. Besides augmentation as described has excellent results, GANs can offer an alternative with promising results. [31] proposed a method to segment exudates by utilizing cGAN to generate images. The Specificity reached at 99.99%. Zheng et al [32] proposed a method to segment Hemorrhages by feeding a U-Net with both real and synthetic images and labels. The Sensitivity was 92.47% by utilizing both synthetic and augmented images to train the network.

In conclusion, while earlier works for retinal lesion segmentation use traditional image processing techniques [43], current works use mostly patch based deep learning approaches [44, 16, 24, 14, 9, 23, 4] with CNNs as dominant architecture. For example in [44] the CNN is trained with patches of a centered pixel. The output is an image with every pixel's value representing the probability of a pixel being EX. A fixed threshold is applied for obtaining a binary image. Generally when using CNNs we cannot talk about segmentation of lesions as the exact boundaries of the lesions are concerned. We talk about localization of the lesions and that is because we classify the patches and then produce probability maps.

The state-of-art models for semantic segmentation are inferentially Fully Convolutional Networks (FCN) and the U-NETs. Towards more precise segmentation, novel designs are continuously proposed. For instance dilated convolutions are introduced in [34]. Attention mechanisms are introduced in [33, 34, 35]. The state-of-art DeeplabV3+ [37, 45] uses both dilated convolutions and spatial pyramid pooling in its contracting path and is one of the most promising models in semantic segmentation nowadays.

Table 3.2 shows the metrics of the papers according to the literature review.

Table 3.2: Metrics according to the literature review

Reference	Year	Dataset	Architecture	Lesion	Sensitivity	Specificity	Accuracy	Precision (PPV)	AUC
[9]	2018	DIARETDB1/ Ophtha	CNN	Exudates	0.96	0.98	0.98	0.94	-
				Hemorrhages	0.84	0.92	0.90	0.85	-
				Microaneurysms	0.85	0.96	0.94	0.83	-
[21]	2019	IDRiD/ MESSIDOR/ e-Ophtha	Combination Of Networks	Exudates	0.779	0.996	0.992	-	-
				Microaneurysms	0.746	0.998	0.997	-	-
[17]	2018	e- Ophtha	UNET	Exudates	0.8458	0.9997	-	-	0.967
[12]	2017	Kaggle/ e-Ophtha	CNN	Microaneurysms	-	-	-	-	0.94
				Exudates					0.95
[22]	2019	IDRiD	FCN	Exudates	-	-	-	0.5498 (PPV vs SE)	-
				Hemorrhages				0.0829 (PPV vs SE)	
				Microaneurysms				0.0059 (PPV vs SE)	
				Soft Exudates				0.1823 (PPV vs SE)	
[13]	2016	ROC/ Messidor/ Diaretdb1	DNN	Microaneurysms	0.97	0.95	-		0.988
[46]	2017	CLEOPATRA/ MESSIDOR, DIARETDB1	CNN	Exudates	0.8758	0.9873	-	-	-
				Hemorrhages	0.6257	0.9893			
				Microaneurysms	0.4606	0.9799			

Continued in next page

Reference	**Year**	**Dataset**	**Architecture**	**Lesion**	**Sensitivity**	**Specificity**	**Accuracy**	**Precision (PPV)**	**AUC**
[47]	2015	STARE/ DR1/ Diaretdb1	Machine Learning	Exudates	0.927	0.8102	0.8723	-	-
[18]	2018	E-Ophtha, ROC, DIARETDB1	UNET	Microaneurysms	0.562	-	-	-	-
[4]	2018	IDRiD	CNN	Exudates	0.9829	0.4135	0.966	-	-
[48]	2018	E-Ophtha, ROC, DIARETDB1	Multilayer Perceptron	Exudates	0.564	0.999	0.998	-	-
[11]	2016	Kaggle/ Messidor	CNN	Hemorrhages	-	-	-	-	0.972 (ROC)
[49]	2016	DIARETDB1	SSAE	Microaneurysms	-	0.9160	0.9138	0.9157	0.9620
[50]	2017	E-Ophtha	CNN	Exudates	0.8885	0.96	0.9192	-	-
[44]	2015	DriDB	CNN	Exudates	0.77	0.77	0.77	-	-
[23]	2017	E-Ophtha	Le-Net CNN	Exudates	0.998	0.996	0.996	-	-
[16]	2017	Kaggle/ DiaretDB1	CNN	Exudates Hemorrhages Red Small Dots Soft Exudates	0.87 0.91 0.52 0.89	-	-	-	-
[24]	2017	Kaggle/ DiaretDB1/ E-Optha	CNN	Exudates Hemorrhages Red Small Dots Soft Exudates	0.735 0.614 0.500 0.809	-	-	-	-
[19]	2019	DiaretDB1/ E-Optha	CNN	Exudates	0.98	0.99	0.98	-	-

Continued in next page

Reference	Year	Dataset	Architecture	Lesion	Sensitivity	Specificity	Accuracy	Precision (PPV)	AUC
[14]	2018	Diaretdb1/ E-Ophtha/ MESSIDOR	CNN	Hemorrhages Microaneurysms	0.4883 0.4883	-	-	-	-
[25]	2017	e-Optha	Le-Net CNN	Exudates	0.99	-	-	-	-
[31]	2018	e-Ophtha, DiaretDB1, HEI_MED, MESSIDOR	DCNNs	Exudates	0.9094	0.9999	0.9997	0.9472	-
[26]	2019	eOphtha/ IDRiD	DRU-NET	Microaneurysms	-	-	0.9999	-	0.9943
[27]	2018	IDRiD	UNET	Exudates Hemorrhages Microaneurysms Soft Exudates					0.718 0.438 0.358 0.456
[29]	2019	IDRiD	UNET	Exudates Hemorrhages	- -	0.9977 0.9985	0.9965 0.9977	0.7888 0.8630	- -
[32]	2018	DRiDB	UNET	Hemorrhages	0.927	-	-	0.80	0.912
[15]	2019	ROC/e-Ophtha	CNN	Microaneurysms	0.800	-	-	-	-
[33]	2019	IDRiD/ IDRiD_VOC	DCNN	Microaneurysms	0.868	-	-	-	-
[20]	2018	ROC/ e-Ophtha, DIARETDB1/ MESSIDOR	AlexNet, GoogleNet, VGGNnet	Microaneurysms	0.6458	0.8800	0.6942		0.8335

Continued in next page

Reference	Year	Dataset	Architecture	Lesion	Sensitivity	Specificity	Accuracy	Precision (PPV)	AUC
[36]	2017	ROC	CNN	Microaneurysms	0.394	-	-	-	-
[28]	2019	IDRiD	HEDNet	Exudates	-	-	-	0.8405	-
				Hemorrhages				0.4812	
				Microaneurysms				0.4392	
				Soft Exudates				0.4839	
[30]	2017	Kaggle	DCNN	Exudates	-	-	-	0.8380	-
				Hemorrhages				0.7445	
				Microaneurysms				0.5678	
[37]	2021	IDRiD/ DIARETDB1	DeepLabV3, FCN, UNET	Exudates	0.94	-	-	-	-
				Hemorrhages	0.87				
				Microaneurysms	0.48				
				Soft Exudates	0.875				
[45]	2016	IDRiD	DeepLabV3/ SegNet	Exudates	-	-	-	-	-
				Hemorrhages					
				Microaneurysms					
				Soft Exudates					
[40]	2020	Kaggle	Lesion-Net	8 types of lesions	-	-	-	-	-
[51]	2018	IDRiD	UNET	Exudates	0.886	0.999	-	0.8268	-
				Hemorrhages	0.7214	0.9977	-	0.5415	-
				Microaneurysms	0.8715	0.9999	-	0.7553	-
				Soft Exudates	0.7791	0.9999	-	0.4915	-
[35]	2019	e-Optha & HEI-MED	FCN	Exudates	0.89	0.99	-	0.81	-

Chapter 4

Machine Learning

Machine Learning (ML) is a discipline of computer science, in which the main purpose is the improvement of algorithms through experience and processing of data. ML is a subfield of Artificial Intelligence (AI). ML has the rationale of training a part of the available data, known as "training data", and further using the model to predict new unseen data without being explicitly programmed for this purpose. ML algorithms can be applied in many fields. For example they are used in Google to predict if an email is spam. ML algorithms can recognize human speech and translate to another language. Computer vision is a field where ML is very useful for many tasks.In many applications, ML can count the number of cars crossing a road,or identify the name of a person who wants to enter a door. The part of interest for our paper is the application in Medicine. ML can be applied in medical images ,extract useful data and further use them for predictions. Thus ML can predict for example if a lesion is malignant or benign. It can also segment a lesion and maybe later use its volume for the appropriate dose of radiation. In all cases the strength and the dominance of ML over conventional algorithms is more than obvious. In many cases it is not feasible to create such conventional algorithms due to the nature of the task. In tasks concerning business, ML is also known as predictive analytics.

ML has a lot of subsets such as computational statistics, data mining etc. Computational statistics predicts new data by exploitation of computational power. Data mining uses supervised learning in order to make exploratory data analysis. Mathematical optimization, is a tool for helping ML not only in theory but also in practise.

ML has a huge difference from a subfield which is called Deep Learning (DL) and we will cover it in the next pages. ML needs human intervention to extract the most informative data as we see in Figure 4.1. In other words, there is no automatisation in this stage,something which is present in DL. For example, in a medical image task where there are mammograms,there must be an image analysis to acquire the features and afterwards decide which of them are the most important. Thus, there is a plethora of ML algorithms and finally the best is chosen.

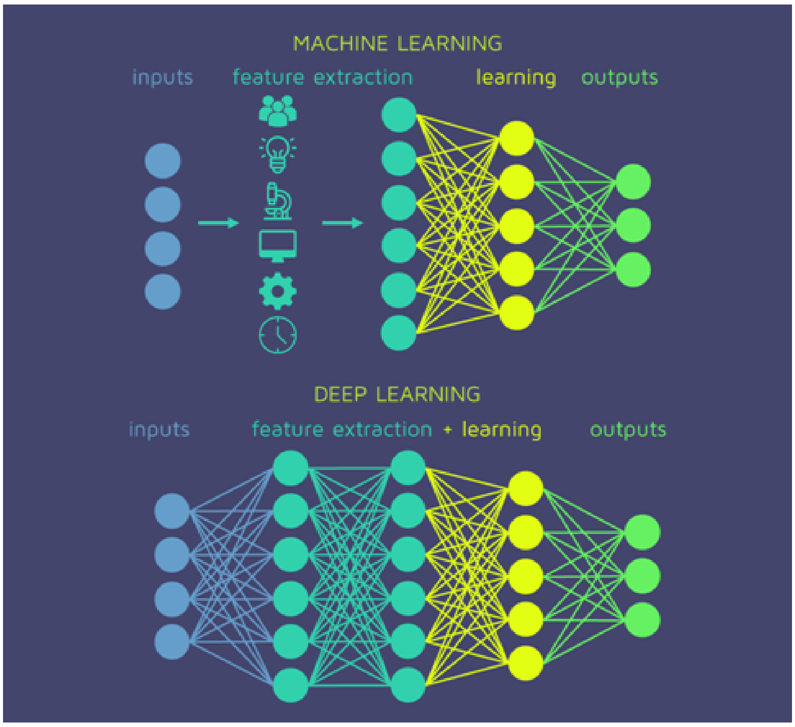


Figure 4.1: There is an obvious difference between ML and DL and this lies in the way that the features are extracted. In ML a human factor determines them, whilst in DL features are extracted automatically.
Image taken from `https://quantdare.com/what-is-the-difference-between-deep-learning-and-machine-learning/`

4.1 Overview of ML

Traditional algorithms work on the basis that what worked well in the past, is likely to continue well in the future. For example, the inference that the sun will rise tomorrow, since it rises for the last billions of years, is more than obvious.

Machine learning algorithms can fulfil tasks even if they are not explicitly meant to do so. It entails computers learning from data in order to do specific tasks. It is possible to build algorithms that teach the machine on how to complete all steps required to solve the problem at hand for basic jobs left to computers; no learning is required on the computer's behalf.

A human may find it challenging to manually create the algorithms required for more complex occupations. In fact, rather than having human programmers explaining each essential step, supporting the computer in designing its own algorithm can be more productive.

There are many approaches to ML algorithms in order to choose the best and teach the computer. This is something common where no algorithm can be satisfactory enough to fulfil a task. When there are a large number of approaches , one option is to mark some of the correct answers as valid. This approach with the best performance can be used for training.

4.2 Historical Retrospection of ML

ML was invented back in 1959 by Arthur Samuel, an American programmer of Artificial Intelligence (AI) and computer games [52]. Another synonym explaining ML rationale was self-teaching computers. Nilsson wrote a representative book for ML during the 1960s ,where he was dealing mostly with pattern recognition [53]. In 1981 there was a report given ,using neural networks to recognise 40 characters (26 letters, 10 digits, and 4 special symbols) from a computer.

Tom M. Mitchell [54] gave a more widely known phrase explaining the ML rationale: "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E.". This sentence is rather a definition explaining the operations during ML training , rather than a strictly compact term. On the other hand, Alan Turing, in his paper *Computing Machinery and Intelligence* [55] proposed an alternative to the common question if machines were able to think. He shifted this question to "Can machines do what we (as thinking entities) can do?"

Today ML tasks may have two different objectives: classification task or regression task.In the first category the algorithm is trained to classify data. For example in the MNIST dataset [56] the algorithm can classify the digits. In contrast, in a regression task the objective is to predict continuous values. For example, given an annual stock of a company, the algorithm can predict future prices.

4.3 Artificial intelligence

Ml was developed under the quest for Artificial Intelligence (AI). In the first era of AI some scientists focused on creating algorithms that computers could learn from data.The approaches to this problem were many including Neural Networks perceptrons and other models which mimic the generalised models of statistics. In Medical diagnosis probabilistic reasoning was also used [57].

However there was a gap between AI and ML. Probabilistic systems were plagued by theoretical and practical problems of data acquisition and representation. Statistics was losing the battle with AI. Neural networks had been abandoned. Backpropagation theory [57] came in the 1980s and gave a boost to ML.

ML was organised again and set as an independent field back in the 1990s. The goal of ML was reexamined and shifted its purpose from tackling solvable problems to problems which have a practical purpose. This detachment from AI , moved ML to models and methods that flirt with probability theory and statistics.

Many sources maintain that machine learning is still a subfield of AI [58]. The key point of contention is whether all ML is part of AI, as this would imply that anyone using ML might claim to be using AI. Others argue that not all ML is part of AI [59] and that only a subset of ML that is ’intelligent’ is part of AI.

ML is trained and can predict based on data, whereas AI implies an interaction with the environment to be trained and consequently achieve the goal [60]

In Figure 4.2 we see that ML is a subfield of AI ,and Deep Learning (DL) which we will analyze in the next section belongs to ML.

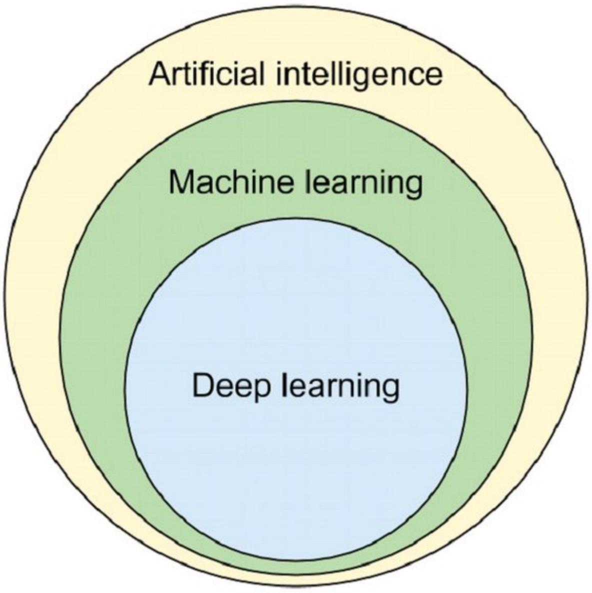


Figure 4.2: ML is a subfield of AI.
`https://en.wikipedia.org/wiki/Machine_learning#cite_note-Definition_of_AI-35`

4.4 Categories of ML

Machine learning is categorized in 4 main groups: Supervised learning, unsupervised learning, semi-supervised learning and reinforcement learning. Each of these algorithms are analysed in the following part.

4.4.1 Supervised learning

Supervised learning algorithms create a model with the data belonging to inputs and outputs. These data are used for training the model and consist of samples. These samples may have more than one input. For example, if we choose to insert the pixels of an image, then we use many inputs. The output is the label onto which the model is trained. This is also called supervisory signal. Tacking the mathematical aspect of the problem, each training sample is depicted by an array or vector (feature vector) and the training data is depicted by a matrix. Through the inner processes of the algorithm the trained model can further be used to predict new unseen samples. Synoptically the model learns from scratch to predict the known data and by optimisation techniques the accuracy is continuously improved until it reaches a saturation point. Support Vector Machine (SVM) is a powerful supervised algorithm that uses hyperplanes to distinguish the available labeled data. In Figure 4.3 we see the classification of black and white dots with SVM.

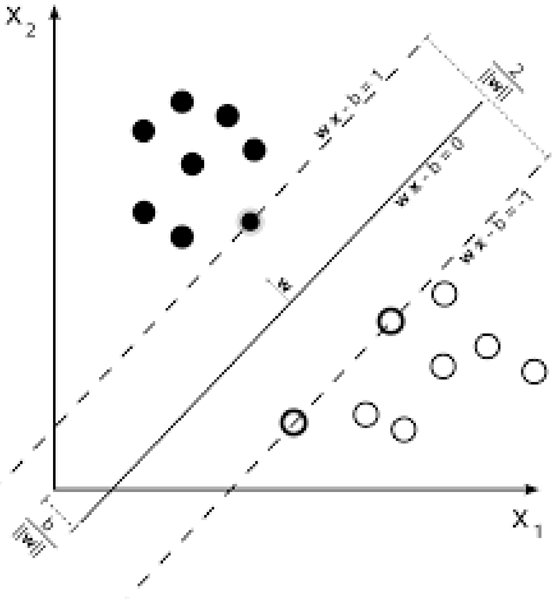


Figure 4.3: A Support Vector Machine (SVM) is a supervised algorithm that is used to classify samples. Here the black dots are separated with a hyperplane (thick line) from the white dots.
Image taken from `https://upload.wikimedia.org/wikipedia/commons/2/2a/Svm_max_sep_hyperplane_with_margin.png`

Active learning, regression and classification are supervised learning algorithms. When the outputs are restricted to a limited set of values we are talking about classification. In contrast when the outputs have any numerical value within a range we are referring to regression.

Another algorithm is similarity learning, which is related to both classification and regression. Here the purpose is to train the model from samples using a similarity function. This function measures how similar or related two objects are. Most common applications are in recommendation systems, in ranking, speaker verification, face verification etc.

4.4.2 Unsupervised learning

Unsupervised learning is a machine learning algorithm in which there is not any provided label for training. Algorithms who belong to the unsupervised learning field accept only inputs. The rationale behind this algorithm is to manage to find possible clusters or groups of the data by finding interrelationships between training data and then try to capture such patterns in the new unseen data. Samples belonging to the same cluster have commonalities and there is a stronger relationship between them. In Figure 4.4 we see how the algorithm divides the data in 3 clusters.

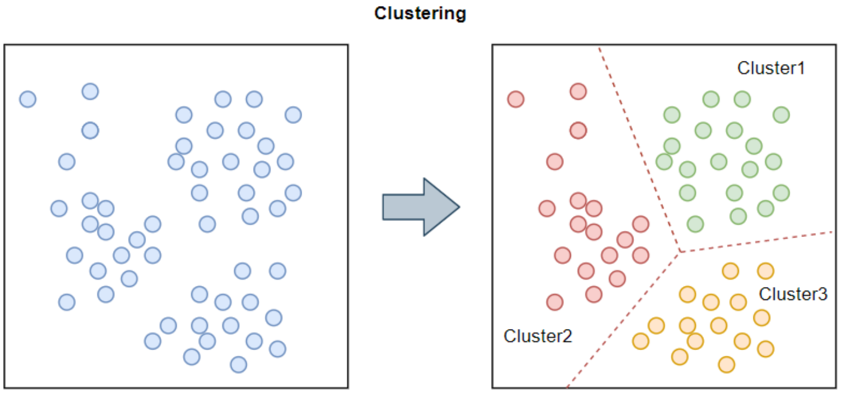


Figure 4.4: The algorithm divides the data in 3 clusters.
Image taken from `https://www.ecloudvalley.com/mlintroduction/`

An application of unsupervised learning is in the field of density estimation such as finding the probability density function. Other examples of this algorithm include clustering where the algorithm divides the samples into clusters with similar features, principal component analysis where the algorithm compresses the data. This is achieved by identifying the most informative ones and discarding the others. In contrast supervised learning works by providing labels for the corresponding samples.

The benefit of this algorithm lies in the minimal preprocessing of the training set, in contrast to supervised learning where there is a huge effort of the experts to assign labels. In this way there is much greater freedom for new patterns to be identified. This is the trade off for this algorithm in a way that it needs more data to be effective and more computational power to be supported.

4.4.3 Semi-supervised learning

Semi supervised learning lies between supervised and unsupervised learning. In such cases the training data are incomplete. Scientists have found that unlabelled data when mixed with labeled ones can increase the performance of the model. Weakly supervised learning [61] own data with noise or data with restricted amount. These data are easier to acquire, ergo cheaper.

4.4.4 Reinforcement learning

Reinforcement learning is a ML algorithm in which intelligent agents [62] interact with the environment in order to maximise the notion of cumulative reward. Reinforcement learning does not work with label data. Instead it gains knowledge by interacting with the environment and using it to confront new unseen conditions. Partially supervised Reinforcement algorithms merge the benefits of Reinforcement learning and supervised learning.

The generality of this algorithm makes it possible to be applied in many fields such as gaming, operation research, control theory, information theory ,statistics and genetic algorithms.In practise Reinforcement learning is used in autonomous vehicles or in learning to play a game against a human opponent. In Figure 4.5 we see an example of using reinforcement learning in gaming.

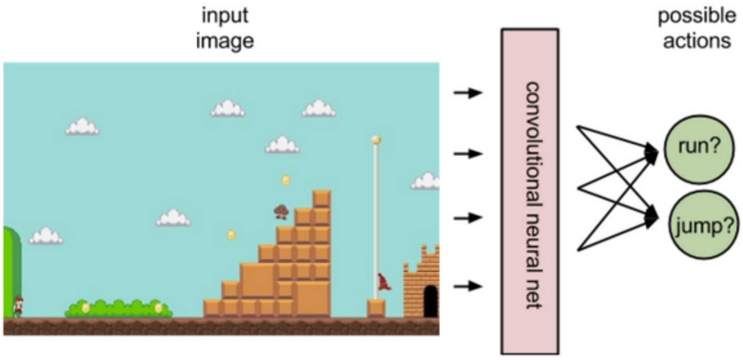


Figure 4.5: Reinforcement learning in gaming.The image illustrates how an intelligent agent works, by guiding a state to the best decision.
Image taken from `https://wiki.pathmind.com/deep-reinforcement-learning`

4.5 Deep Learning

Deep Learning (DL) is a subfield of ML ,which is mainly based on artificial neural networks. DL uses multiple layers which automatically extract features. In deeper layers the algorithm extracts higher-level features. The term "Deep" refers to this multiplicity of layers, DL.

The performance of a DL depends on its architecture. Examples of DL architectures are deep neural networks (DNNs), Deep Belief Networks, Recurrent Neural Networks (RNNs), Convolutional Neural Networks (CNNs), UNETs, Fully Convolutional Networks (FCNs) etc. Applications of DL can be found in Computer Vision, Natural Language Processing (NLP), medical imaging, bioinformatics, drug designing etc. The results of using DL networks in many cases surpass human experts.

The inspiration of Artificial Neural Networks (ANNs) lies in the way that information is processed and distributed in neurons. On the other hand, ANNs differ from human brains in a way that ANNs tend to be static in contrast with the human brains which are analogue [63].

DL is a contemporary variation in which an unlimited number of layers can be used. Theoretically DL may have a universality under no extreme conditions and thus be used in many applications. Heterogeneity of DL layers is allowed, and layers may deviate from biological models. This is due to the fact that DL models have to be trainable, understandable and efficient. Applications of DL are speech recognition, image recognition, visual art processing, natural language processing, drug discovery and toxicology, recommendation systems, bioinformatics, medical image analysis, military etc.

4.5.1 Overview of DL

ANNs and more specifically Convolutional Neural Networks (CNNs) form the basis of most DL models. In DL, each layer transforms the input data into a more conceptual and mixed depiction. For example, in a classification task with the image of some persons as inputs, the first layer may capture the edges, the second layer may encode arrangement of edges, the third layer may capture the nose and the eyes of a person and the next layer may capture holistically the existence of a face. A DL algorithm can learn the placement of the features to the appropriate level on its own. This does not exclude human intervention and this is necessary in fine tuning of the hyperparameters of the model. For example a human must determine the number of layers, the architecture of the model, the learning rate, the loss function etc.

DL systems have an ample credit assignment path (CAP) depth. The CAP is the path of transformations from input to output. CAPs depict conceivably connections among input and output. A feedforward neural network has depth of the CAPs equal to the network and is the number of hidden layers plus one output layer. The majority of experts think that DL necessitates a CAP depth greater than 2. In the sense that it can simulate any function, CAP of depth 2 has been proved to be a universal approximator [64]. More layers, on the other hand, do not improve the network's function approximator ability. Extra layers help in learning the features successfully since deep models (CAP > 2) can extract better features than shallow models. A greedy layer-by-layer method is used to build DL architectures.

In supervised learning tasks there is elimination of feature engineering .This is achieved by translating the data into compact representations related to principal components. Thus redundancy is avoided.

Unsupervised learning tasks can benefit from deep learning algorithms. This is a significant advantage because unlabelled data is more plentiful than labeled data. Neural history compressors and deep belief networks [65] are two examples of deep structures that can be trained in unsupervised manner.

4.5.2 Interpretations

Deep Neural Networks are explained with probabilistic inference or the universal approximation theorem. The capacity of feedforward neural networks with a single hidden layer simulates continuous functions. George Cybenko published the first evidence for sigmoid activation functions in 1989 [66], and Kurt Hornik generalised it to feed-forward multilayer architectures in 1991 [67]. The capacity of networks with bounded width but the depth is allowed to grow according to the universal approximation theorem for deep neural networks. In [68] Lu et al suggested that if the depth of the deep neural network ,using RELU as activation function, is larger than the dimensions of the input then any Lebesgue integrable function may approach the network. In the opposite case, with a depth of the network smaller than the dimensions of the input, the deep neural network does not account for an universal approximator.

Machine learning gives a probabilistic explanation. It presents inference, as well as the training and testing optimisation methods, which are connected to fitting and generalisation. The probabilistic explanation takes the activation nonlinearity into account as a cumulative distribution function. This probabilistic aspect resulted in the appearance of dropout as regularizer. Hopfield [69], Widrow [70] and Narendra [71] were the first to talk about probabilistic interpretation as a concept.

4.5.3 History of DL

The first signs of DL rationale can be found in 1943. Walter Pitts and Warren McCulloch invented a computer model which mimicked the human brain, with the help of neural networks [72]. ”Threshold logic” was a mixture of algorithms which tried to capture the process of human thinking. DL has been evolving with steady steps since that time.

Back Propagation Model was developed by Henry J. Kelley in 1960, who gave a boost to DL evolution [73]. A simpler version based on the chain rule was invented by Stuart Dreyfus [74]. Many factors make it inefficient for the Back Propagation Model to be applied until 1985, although the theory existed since 1960, as mentioned earlier.

Alexey Grigoryevich Ivakhnenko and Valentin Grigor’evich Lapa, authors of Cybernetics and Forecasting Techniques [75], were the first to make efforts in developing DL algorithms back in 1965. The activation functions they used in their models were polynomials. Models were analysed statistically for each layer and the best statistically chosen features were passed to the next layer. This was a manual and slow process.

The first AI winter began in the 1970s, as a result of promises that could not be kept. The lack of funding had a negative impact on both DL and AI development. Fortunately, there were others who continued the research even if they didn't have any money. Kunihiko Fukushima was the first to use Convolutional Neural Networks (CNNs) [76].

Fukushima created numerous pooling and convolutional layers in his neural networks. He created the Neocognitron artificial neural network in 1979, which featured a hierarchical, multilayered design. The computer was able to "learn" to recognise visual patterns thanks to this architecture. The networks had similarities with contemporary versions, although they were trained with reinforcement strategy, something that was evolving during the upcoming years.

Back propagation evolved during the 1970s and its theory concerned the exploitation of errors for further correction of the performance. Seppo Linnainmaa used FORTRAN language to code for backpropagation [77]. Until 1985 backpropagation was not applied as a concept. Rumelhart, Williams, and Hinton proved that backpropagation could have interesting results [78]. From a philosophical aspect, this evolution of backpropagation enlightened the question if human understanding is based on distributed representations (connectionism) or symbolic logic (computationalism) . Yann LeCun in 1989 demonstrated the practical aspect of backpropagation in computer terminals.He used backpropagation in his CNNs to classify handwritten digits [79].

1999 was the year where a major evolutionary step for DL was accomplished. The computational power was growing because of the appearance of GPU (graphics processing units). This was a novel and determinant step for the evolution of DL.GPUs could process images really faster and generally the computational speed increased by 1000 times over a 10 year period [80]. During this period Support Vectors Machines (SVMs) competed with neural networks. While a neural network is slower than a support vector machine, it produces better results when working with the same data. The advantage of neural networks is that they improve as more training data is added.

The “Vanishing Gradient Problem” appeared around 2000 [81]. In this case, features learned in lower layers could not be passed to the upper layers due to the vanishing of the signal. This wasn’t a problem with all neural networks.Instead, it was only with those that used gradient-based learning methods. Certain activation functions were found to be the root of the problem. A number of activation functions compressed their input, resulting in a fairly chaotic reduction in output range. As a result, enormous amounts of input were mapped over a very limited range. A substantial change in the input will be reduced to a small change in the output for certain areas, resulting in a vanishing gradient. Layer-by-layer pre-training and the formation of long short-term memory were two ways utilized to tackle this challenge.

ImageNet was founded in 2009 by Fei-Fei Li, an AI professor at Stanford, who collected a free database of over 14 million tagged photos [82]. There are a lot of unlabeled photographs on the Internet. To “train” neural nets, labeled images are required.

The 2009 NIPS Workshop on Deep Learning for Speech Recognition [65] was prompted by the constraints of deep generative models of speech and the prospect that deep neural nets (DNN) could become practical given more capable hardware and large-scale data sets. It was thought that pre-training DNNs using generative models of deep belief networks (DBN) would solve the neural nets’ fundamental problems [83].

In 2010, researchers used extensive output layers of the DNN based on context-dependent HMM states created by decision trees to expand deep learning from TIMIT [84] to large vocabulary speech recognition [85].

By 2011, GPUs had substantially improved in speed, allowing convolutional neural networks to be trained ”without” layer-by-layer pre-training. With the increased computing speed, it became evident that DL had considerable efficiency and speed advantages. AlexNet, a convolutional neural network, won multiple international competitions in 2012 due to its architecture [86]. To improve the speed and dropout, rectified linear units were utilised.

Google Brain also published the results of an interesting experiment called The Cat Experiment in 2012 [87]. The informal project looked into the limitations of ”unsupervised learning.”. DL employs ”supervised learning”, which entails training the convolutional neural network using labeled data. A convolutional neural network (CNN) is given unlabelled data as input and asked to look for recurring patterns through unsupervised learning. The “Cat Experiment” used a neural network that was distributed over 1,000 computers. The training software was allowed to run after ten million ”unlabelled” images were randomly selected from YouTube and given to the system. One neuron in the top layer was shown to respond strongly to images of cats at the end of the training.In the discipline of DL, unsupervised learning remains a major priority. In terms of processing unlabelled images, the “Cat Experiment” performs about 70% better than its predecessors. However, it only detected only 16% of the objects used in training, and it performed significantly worse when the objects were rotated or moved.

In 2012, a team led by George E. Dahl won the ”Merck Molecular Activity Challenge” by predicting the biomolecular target of one drug using multi-task deep neural networks [88].

DL is being used in both the processing of Big Data and the advancement of Artificial Intelligence. DL is still growing, and is in search of novel ideas.

4.6 Transfer Learning

Stevo Bozinovski and Ante Fulgosi back in 1976 published a paper introducing transfer learning in the training stage of a neural network [89]. This work gives a mathematical model of transfer learning. Lorien Pratt in 1993 presented the discriminability-based transfer (DBT) algorithm in a paper in ML field [90]. Cognitive science was a field for transfer learning application, as Pratt edited an issue of Connection Science, having to do with the use of transfer learning in neural networks [91].

One of the most novel potentials of Deep Learning is that a model may not be trained from scratch in some demanding situations. Instead it can inherit knowledge from another relevant gnostic domain, as seen in Figure 4.6, ImageNet consists of over 14 million real life images, such as trees, cars, food, people, machines etc. Medical images and especially annotationed from doctors, are very difficult to acquire as previously mentioned and in many studies transfer learning is used when the dataset is poor [6, 92].

The main benefit when using transfer learning is that the algorithm converges faster and by this way the whole training lasts significantly shorter than training from scratch. Moreover the accuracy of a model increases compared to a trained model from scratch. In conclusion, in circumstances when the available dataset is problematic both in quantity and in quality, transfer learning is an excellent remedy. In our study we exploited a pretrained network (MobileNetV2) which is trained on ImageNet, to overcome the problems of our available dataset.

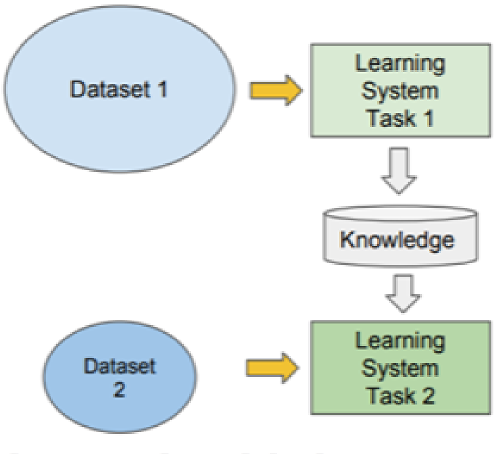


Figure 4.6: Knowledge acquired from training in task 1 with dataset 1 is exploited for training in task 2 with another dataset 2.
Image taken from `https://towardsdatascience.com/a-comprehensive-hands-on-guide-to-transfer-learning-with-real-world-applications-in-deep-learning-212bf3b2f27a`

Chapter 5

Most used architectures in semantic segmentation

5.1 Semantic Segmentation

Image segmentation is an important part of Computer Vision. Nowadays, Computer Vision has many applications in the Automotive Industry, Disaster Relief and Emergency Situations, Agriculture, Healthcare, Security, Finance, Retail and Inventory Management, Advertising.

Semantic segmentation is the task of decomposing images into classes. With the term 'class' we mean each individual entity of an homogeneous ensemble.

Segmentation is useful because it converts an image into something that is more meaningful and easier to analyse. More precisely, semantic segmentation is the assignment of a label to every pixel of an image. Thus each individual class contains pixels with the same characteristics such as colour, intensity, or texture. It differs from image classification, where one or more labels are assigned to the whole image. When applied to a stack of images, for example in Magnetic Resonance Imaging (MRI), the subsequent contours after image segmentation may be used to reconstruct 3D images.

There is also a more informative way to segment images and is called Instance segmentation. This type of segmentation carries more information because it segments and can distinct objects belonging to the same class.

Binary segmentation is the simplest category of semantic segmentations. In this case the pixels may belong either to one (positive) label or to the other (negative) label. In cases where the classes are more than two, we are talking about a multi-class segmentation problem. In our study we will utilise binary segmentation and will repeat the experiment as many times as the number of the lesions we examine.

5.2 Convolutional Neural Networks (CNNs)

The term “convolution” implies the existence of a mathematical operation. A Convolutional Neural Network is an artificial neural network. The major usage and applications concern feeding the network with images as input. Their alternative name is also shift invariant or space invariant artificial neural networks (SIANN). This name is based on the feature maps or translation equivariant responses that the networks provide, based on the sliding convolution kernels. Thus most CNNs are equivariant, in contrast to invariant, to translation.Their applications are many: image and video recognition, image classification, recommender systems, financial time series, natural language processing, medical image analysis, image segmentation and many others.

CNNs are versions of multilayer perceptrons with regularisation. Multilayer perceptrons are fully connected networks where each neuron is connected to all neurons of the next layer. The advantage they have against overfitting lies in the fully connected layer that they have in the final levels of their architecture. Traditional remedies against overfitting are regularisation methods such as penalty of parameters (weight decay), skipping connections or dropout. CNNs tackle overfitting with an alternative remedy :they exploit the hierarchical patterns of data, constructing more complex patterns with the use of filters.

CNNs were created in a way that they mimic biological processes [93]. Animal’s visual cortex has an infrastructure which the CNNs try to mimic with the connectivity between the neurons. Individual cortical neurons can only respond to stimuli in a small area of the visual field called the receptive field. By covering the whole optic field ,the receptive fields of different neurons may overlap. The major advantage compared to traditional ML is that the features are extracted automatically and not by human intervention.

If CNNs are used in image level, they cannot reveal where a candidate lesion is hidden. Probability maps come into play, by giving an intuition about “where” does the model detect the class (lesion) which searchers for. Many studies [9, 12, 11, 50, 44, 23, 16, 14, 25, 15, 36], create probability maps and often fuse them with the original image to get an integrated visualization.

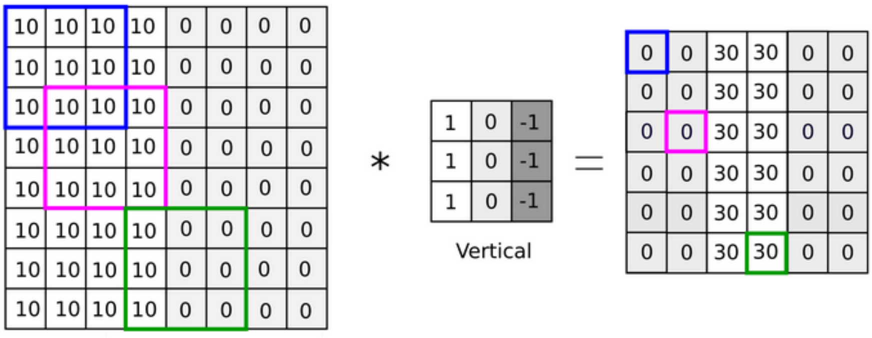
5.2.1 Architecture of CNNs

In traditional neural networks , neurons accept signals from specific regions of the prior layer. In contrast, in CNNS there is a limited area from which neurons accept the signals of the previous layer. This area is the neuron's receptive field. The shape of the area is normally a square. In a fully connected layer ,the area is the whole prior layer. As a result, each neuron in each convolutional layer accepts input from a greater region in the input than in prior levels. This is due to the repeated use of the convolution, which considers the value of a pixel as well as its surrounding pixels. The number of pixels in the receptive field remains constant while utilizing dilated layers, but the field becomes increasingly sparsely populated as the dimensions of the field increase.

A CNN comprises an input layer, the middle hidden layers and the final output layer. They are called hidden because the activation function and the convolution masks the input and the output. The convolutions are executed in the hidden layers. After passing the input image the hidden layers perform a dot product of the kernel with the image. Frobenius inner product is the terminology for this product and the activation function used is the RELU. The convolution kernel passes over the input image and simultaneously feature maps are created through this step. This feature map is the input for the next layers. The following layers are pooling layers and fully connected layers.

In a CNN the way that the network accepts the inputs is in the form of a tensor with shape equal to (number of inputs × input height × input width × input channels). The convolutional layer shifts the image in a conceptual way, by forming the feature maps with shape equal to (number of inputs × feature map height × feature map width × feature map channels). Convolutional layers perform the convolution praxis and pass the feature map to the next layers. This resembles the response of a neuron to a stimulus [94]. Despite the fact that fully connected feedforward neural networks can classify data, they are not appropriate for images with high resolution. Practically it would require a very huge number of neurons, no matter if the architecture is shallow. For example a relatively small image of size 100 x 100 has 10,000 weights for every neuron in the second layer. In contrast convolution minimizes the parameters in a way that the network can increase in depth [95]. With the use of regularized weights , problems such as the vanishing gradient and exploding gradient disappear unlike in the backpropagation stage of classic neural networks. Additionally , CNNs are appropriate for data distributed in grid-like form such as images , as they take into consideration the spatial connections of features.

In Figure 5.1 we see an original image on the left and the 3×3 kernel in the middle. After the convolution the resulting feature map is on the right.The blue region on the image is finally converted to the blue region on the feature map.



10	10	10	10	0	0	0	0
10	10	10	10	0	0	0	0
10	10	10	10	0	0	0	0
10	10	10	10	0	0	0	0
10	10	10	10	0	0	0	0
10	10	10	10	0	0	0	0
10	10	10	10	0	0	0	0
10	10	10	10	0	0	0	0

1	0	-1
1	0	-1
1	0	-1

0	0	30	30	0	0
0	0	30	30	0	0
0	0	30	30	0	0
0	0	30	30	0	0
0	0	30	30	0	0
0	0	30	30	0	0

Figure 5.1: The convolution process of an image on the left with a 3×3 kernel results in the feature map on the right.
Image taken from `https://anhreynolds.com/blogs/cnn.html`

Pooling layers may follow the convolutional layers in many architectures. The rationale behind using these layers is the dimensionality reduction. This is achieved by combining the output of neuron clusters into a single neuron in the following layer. There are 2 common categories of pooling layers: Max Pooling and Average Pooling. Max Pooling layers finds the maximum value of each local group of neurons, whilst Average Pooling finds the average value.

In Figure 5.2 we see the effect of using a 2×2 kernel in a Max Pooling layer. The result is a smaller image keeping the same characteristics of the original image.

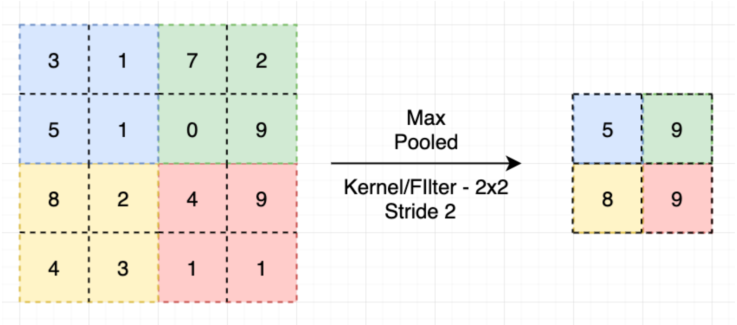


Figure 5.2: The effect of the Max Pooling Layers on an image using a 2×2 kernel. The reduction of the image size is obvious, keeping though the characteristics of the original image. Image taken from `https://ai.plainenglish.io/pooling-layer-beginner-to-intermediate-fa0dbdce80eb`

The final layer is the Fully Connected Layer. Every neuron of a previous layer is connected with all the neurons of the following layer. After this stage, the feature matrix is shifted to a one dimensional vector and can be classified as seen in Figure 5.3. SoftMax function is usually applied at this phase (most used in CNNs multi-class tasks) providing a list of probabilities for the candidate classes.

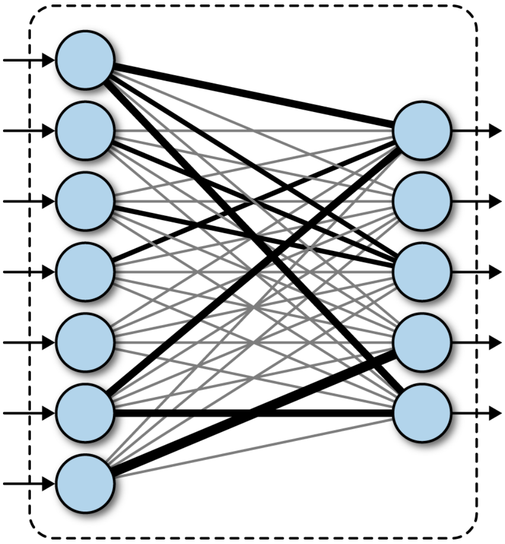


Figure 5.3: A fully connected layer.
Image taken from `https://www.oreilly.com/library/view/tensorflow-for-deep/9781491980446/ch04.html`

In Figure 5.4 we see all the layers described earlier as part of the architecture. In this example of DR lesion classification, the input image is passed through the layers and the result is a probability for each type of lesion. More precisely the lesion is predicted to be Hemorrhages as results with the highest probability amongst the others.

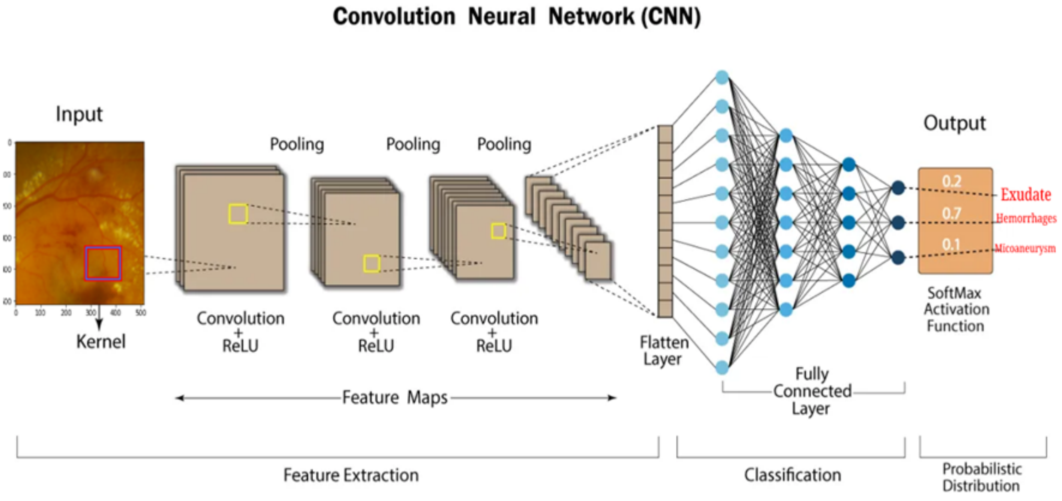


Figure 5.4: Classification of a fundus image with a CNN.
Modified Image from `https://www.analyticsvidhya.com/blog/2021/05/20-questions-to-test` `yourskills-on-cnn-convolutional-neural-networks`

5.3 UNETs

U-Nets were developed at the Computer Science Department of the University of Freiburg, Germany. They are a modification of a CNN which was initially targeted in image segmentation of biomedical images. The architecture of a UNET comprises two paths: the first path is the contracting path or the encoder or the path which is responsible for the analysis. It is exactly the same as CNN, providing information for classification. The other path is an expansion path or decoder or the path which is responsible for synthesis. It comprises up-convolution layers and also concatenations which arrive from the encoder, containing features. The decoder allows the network to access the spatial information lost from the encoding stage. Moreover the resolution of the output is increased due to the expansion path. The resulting output passes to a convolution layer to construct the segmented image. The architecture of the network is practically symmetric ,resembling the “U” letter. The objective of most CNNs is to perform an image level classification. On the other hand, they cannot provide pixel level classification, something determinant for medical imaging analysis. Historically speaking there were previous attempts for segmentation tasks, but Ronneberger et al. [96] made huge improvements in medical image segmentation tasks. The inspiration of UNET was based on previous works of Long et al. [39] who used fully convolutional networks. The performance of their networks surpassed the previous best on ISBI 2012 challenge.

The importance of UNETs lies in the fact that they can be trained with a very limited number of images and despite this fact they can create very detailed segmentation maps. This is extremely important in medical imaging due to the rarity of annotated datasets. This unique characteristic is accomplished by random elastic deformation of the training data [96]. If two instances of the same class have touching borders, then the segmentation is achieved by applying a weighted loss function that penalises the model in case of wrong separation of two instances. Moreover UNET has training time much faster compared to other segmentation models due to its methodology based on learning of the context.

Since the development of UNETs in 2015,there has been an outbreak in the medical community. New methods and approaches have been developed ,as it was expected, to enforce the power of their predecessors. In [26, 27, 29, 31, 32], they use U-nets or combinations for segmenting the lesions.

5.3.1 UNET architecture

UNET networks have two parts as mentioned earlier. The first one is the encoder or contracting which resembles a CNN architecture. We can see the architecture in Figure 5.5. Every single block in the encoder comprises two 3×3 convolution layers, a RELU activation function and a max-pooling layer. This motif is repeated 4 times. Something that makes UNETs unique is that the feature map after passing to the decoder is upsampled with the help of a 2×2 convolution layer. The corresponding feature map of the encoder is cropped and concatenated to the previous upsampled feature map of the decoder. Two 3×3 convolutional layers and a RELU activation function follow up. The last phase of the architecture has an 1×1 convolutional layer which is applied to the feature map for dimensionality reduction and thus the segmentation image is feasible. The reason for cropping is that the edges of the feature map include the least contextual information and need to be avoided. The whole architecture of the UNET looks like the “U” letter as mentioned earlier.

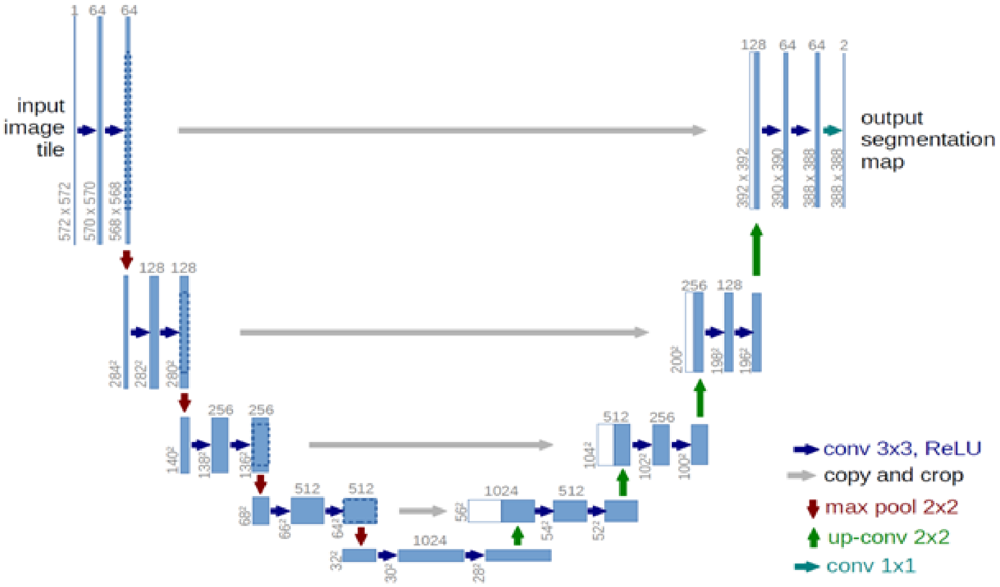


Figure 5.5: The architecture of a UNET.
Modified Image from `https://lmb.informatik.uni-freiburg.de/people/ronneber/u-net/`

The energy function for the network is given by the following equation:

$$E = \sum w(x) \log\left(p_k(x)\right), \tag{5.1}$$

where p_k is the pixel-wise SoftMax function applied over the final feature map,

$$p_k = \frac{e^{a_k(x)}}{\sum\limits_{k'=1}^{k} e^{a_{k'}(x)}} \tag{5.2}$$

and $a_k(x)$ denotes the activation in channel k.

5.4 FCNs

A Fully Convolutional Network (FCN) is a modified CNN where FC layers have been removed and deconvolution layers are added to undo the effect of down-sampling and get an output map of the same size as input image.

A difference between UNet and FCN relies on upsampling. In FCNs, a downsampling feature map of the same level and an upsampled feature map are simply added and upsampled. In contrast, in UNets, they are concatenated and then go through some convolutional layers for further processing. Figure 5.6 shows an FCN.

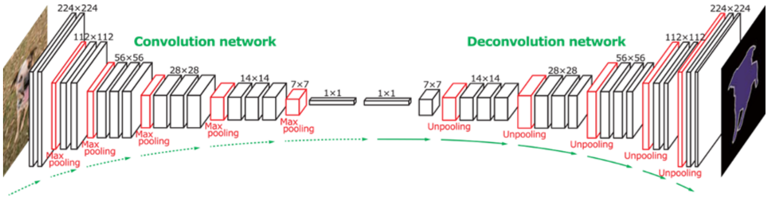


Figure 5.6: A Fully Convolutional Network.
Image taken from: `http://www.kafftjishqi.com/blog/fully-convolutional-neural-networks/index.html`

5.5 Stacked autoencoders

Stacked Autoencoders (SAEs) comprise of blocks. Each block is called "Autoencoder(AE)" and is simply a neural network with a single hidden layer with the same input and output. The training of a SAE has 2 phases: pre-training and fine tuning. The pre-training phase is achieved with unsupervised algorithms followed with supervised ones in the second phase.

There are two types of AE: sparse autoencoders and denoising autoencoders (DAE)s. The first category is used to extract sparse features from raw data whilst the second is used for recovering inputs (images) with noise.

5.6 Generative Adversarial Networks (GANs)

In 2014 Ian Goodfellow devised Generative Adversarial Networks (GANs) which belong to the ML field [42]. The philosophy behind the design lies in a contest amongst two networks,the generator and the discriminator , with one trying to deceive the other. Synthetic data are generated given a training set. If a GAN for instance is trained on MRI images ,it can generate MRI ,at least plausible to the human eye taking advantage of the many pragmatic elements. Although GANs had been initially proposed for unsupervised learning tasks ,they are also powerful for supervised, semi-supervised and reinforcement learning tasks.

The main concept of GANs is that the generator instead of trying to minimise the loss function, it contrarily tries to fool the discriminator as shown in Figure 5.7. This kind of training is unsupervised learning.

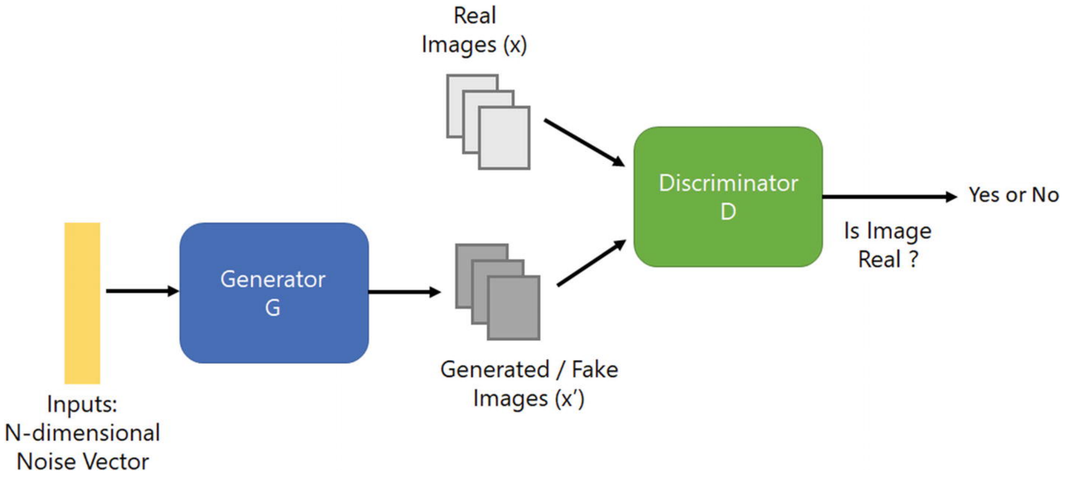


Figure 5.7: The discriminator tries to find out if the input image is real or synthetically generated from the generator.
Image taken from `https://link.springer.com/chapter/10.1007/978-1-4842-3679-6_8`

The generator generates fake or synthetic images whilst the discriminator checks them. The network which constitutes the generator is trained to correspond from a latent space to a data distribution similar to the original. The network which constitutes the discriminator, in contrast, identifies the source of the input, thus distinguishing whether it is real or synthetic. The role of the generator is to increase the error rate of the discriminator, in other words to fool the discriminator.

The discriminator is initially trained on a known dataset until it reaches the highest accuracy. On the other hand the training of the generator is based on the success of deceiving the discriminator. The training of the generator begins with random noise sampled from a predefined latent space. Afterwards, the candidate images are checked by the discriminator. Backpropagation algorithm is applied to both networks. Thus, the generator learns to produce more realistic data, whilst the discriminator learns to recognise the synthetic data better. In cases where GANs deal with images, a deconvolution network comprises the generator while a CNN the discriminator.

A limitation that GANS have is the "mode collapse", as they don't generalise properly and miss whole modes from the input. For instance, a GAN trained to classify 10 digits may not include a digit to its output. There are some theories about this problem, concerning the selection of loss function and others the poor training of the discriminator who fails to notice the omission of one digit. This problem is yet unsolved [97].

Table 5.1 below shows some of the architectures used according to previous literature review.

Table 5.1: Architectures used according to Literature Review

Architecture	Reference
CNN	[9, 12, 11, 50, 44, 23, 16, 14, 25, 15, 36]
FCN	[17, 18, 31, 22, 35, 37]
UNet	[31, 26, 27, 29, 32, 37, 51]
Stacked Sparse Auto-encoder (SSAE)	[49]
Generative Adversarial Networks (GANs)	[27, 28]
Discriminative Restricted Boltzmann Machines (DRBM)	[19]
DeepLabV3	[37, 45]
SegNet	[45]

Chapter 6

Available Public Datasets for DR

High performance in Deep Learning depends on the available datasets [98]. The networks are “data hungry” and the doubt of decent performance of the model arises due to the limited medical datasets. There are several reasons for this issue. First of all, the privacy of the patient’s data renders the access very difficult. Only with the patient’s consent, data can be accessed. Second, the annotation is very time consuming and expensive. Physicians must determine the exact boundaries of the lesions, something harder than an image based annotation. The problem of limited dataset leads to “overfitting”. Generally this means that although the algorithm may have high accuracy in train set, it cannot generalize in new unseen images. There are many remedies to tackle this difficult problem such as augmentation, use of Dropout layer, use of synthetic images, use of regularization in loss function etc.

Another limitation is the problem of imbalanced datasets both in image and pixel level. The images with lesions are much less than the normal in a dataset. And this happens clearly due to statistical reasons. In pixel level, the imbalance appears because the lesions occupy much less pixels compared to the whole image. In all cases the imbalanced datasets lead to biased weights, and thus to lower performances. [15] tackles this problem with a two stage approach of passing a probability map to a CNN. [31] tackles the same problem with the use of conditional adversarial networks (cGANs).

Finally, a common problem amongst the most datasets is their quality both in image as well as in annotation level.

Fundus images are acquired under “conditions of a real world”. This means that many of them do not have the appropriate conditions for descent acquisition. [11] proved that the quality of the image influences the algorithm and thus the performance depends on it.

These limitations make segmentation tasks very challenging because they demand the devise of new techniques based on the rarity of annotated datasets. Table 6.1 is a synaptic table with the most common public datasets that are used.

Table 6.1: The most common public datasets used

Name	Number of images	Resolution	Annotation	Task
MESSIDOR	1200	Varying	Image level	DR grading
MESSIDOR 2	1748	Varying	Image level	DR grading
IDRiD	516	4288x2848	Image & pixel level	DR grading HE, MA, SE, HM detection
e-Optha	463	Varying	Pixel level	EX, MA detection
DRiDB	50	Varying	Pixel level	MA, HM, HE, SE, Optic disc detection
STARE	400	605x700	Pixel level	Vessel segmentation, Optic nerve detection, 13 retinal diseases
DRIVE	40	768x584	Pixel level	Vessel segmentation
Eye-PACS	88702	Varying	Image level	DR grading
DIARETDB1	89	1500X1152	Pixel level	HE, HM, SE, red small dots
Kaggle	80,000	Varying	Image level	DR grading
CHASE	28	1280X960	Pixel level	Vessels segmentation
DRISHTI-GS	101	2896x1944	Pixel level	Optic disc, Optic Cup segmentation
ARIA	167	768X576	Pixel level	Optic disc, fovea, vessel segmentation
DRIONS-DB	110	600x400	Pixel level	Optic disc segmentation
ORIGA	650	720X576	Pixel level	Optic disc, Optic Cup segmentation
REVIEW	16	Varying	Pixel level	Vessel segmentation
SEED-DB	235	3504X2336	Pixel level	Optic disc, Optic Cup segmentation
RIM_ONE	169	-	Pixel level	Optic nerve segmentation
HRF	45	3504x2336	Pixel level	Vessel segmentation

Chapter 7

Dataset preprocessing

7.1 Preprocessing techniques

In the pipeline to implement a segmentation task, there are some stages which are crucial for the outcome. The retinal camera works as a fountain of images where there are several factors which determine their quality. In all cases bad quality of fundus images may hide some lesions and important information may be lost. In Figure 7.1 we can see images with several technical problems.

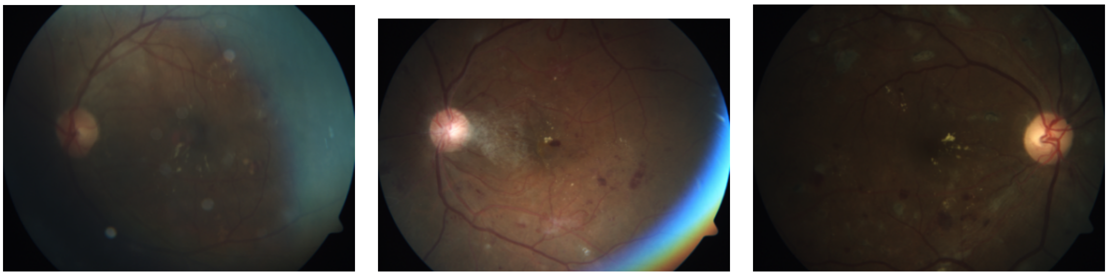


Figure 7.1: Several technical problems that are present in real world datasets.
Images taken from: https://www.kaggle.com/c/diabetic-retinopathy-detection

Generally with the term 'noise' we describe any cause which leads to loss of high frequency information and thus a rough appeal of the image. Noise may come from several reasons such as dust, bad quality of the camera, bad camera settings, bad pose of the subject, movement of the eye and generally not professional conditions of acquiring images. Thus, many artifacts due to unpredictable reasons may appear. In the real world, a small portion of an acquired dataset with such images is probable to be extracted and must be tackled as best as possible.

A common issue of problematic images is the lack of sufficient illumination. In such cases dark regions appear in images and generally there is an heterogeneity in pixel's intensity. In some other cases the opposite happens where the images seem "burned" and there are regions overexposed to light.

Many studies [17, 18, 99, 100, 49, 101, 50, 102, 14, 31, 29, 33, 36] begin the preprocessing by extracting the green channel because it provides the highest contrast.

There are several ways to get rid of the noise. In [28] they apply Non-local Means Denoising algorithm to reduce it as a starting point. In other cases [18] they use morphological operators.

Contrast enhancement (CE) [9, 101, 31] is used to enhance the contrast between the lesions and the background and is produced by subtracting the Gaussian filtered image from the original image while adding a baseline factor over the grayscale (γ).

Contrast Limited Adaptive Histogram Equalization (CLAHE) [103, 33, 28] is a technique used for contrast enhancement and affects small regions instead of the whole image. This leads to better results compared to normal CE.

In some cases [99, 50, 32] some studies believe that some false positives have to do with the existence of vessels close to lesions with similar characteristics so they remove the vessels. This is achieved with Image processing techniques such as morphological operators and in some other cases [33], Otsu's threshold was used to divide the image in binary regions for further processing.

Besides the issues having to do with Image Processing in many cases the processing comprises stages having to do with configuration of images. The main reason is because the models work better with some manipulations. For example in [101] they scaled the pixel intensities between zero and one. In [16] they standardized the pixel's intensities by subtracting mean and dividing by standard deviation.

Almost in all cases the most common step is to resize the images or patches to fit to the input of the upcoming network [99, 100, 11, 16, 24, 26].

The most common public datasets have a very small number of images and this is addressed in many ways. This problem is analyzed in the next section. But in some cases [9, 17, 12, 18, 4, 48] patches are created as a starting point. They may be augmented to address overfitting [14], or balanced by removing patches which create bias to the model [15].

We worked with IDRiD dataset, which is a very a qualitative dataset. In our study we did not apply any Image Processing technique because the images were very decent, except from scaling the data between zero and one and creating patches to increase the number of images. The only additional step as it will be later explained is that we did not use all the available patches but instead we set up a threshold to maintain the most informative ones. Those patches were further used in augmentation algorithm.

7.2 Augmentation

The most common problem in Deep Learning is overfitting. By this term we mean that the model may have a great performance in the train set but fails to generalize in new unseen data. In the next sketch we can see 3 different types of fitting the data. The first one shows the ideal fit of the data, in contrast with the next ones. The second sketch appeals overfitting whilst the third appeals underfitting. It is obvious that in case of overfitting the model tries to capture all possible training data. The result is the poor performance on test data. In underfitting case, the model has a bad performance on training dataset. Figure 7.2 depicts these types of data fitting.

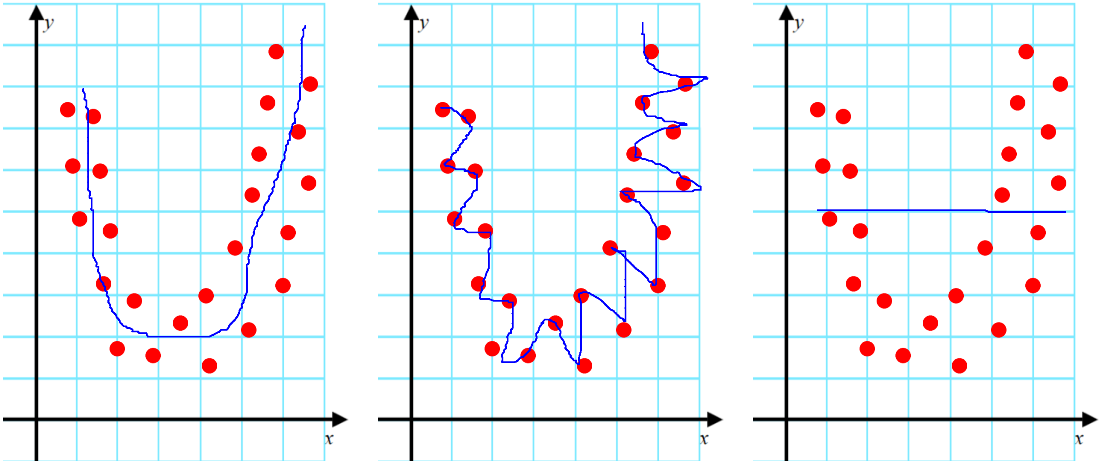


Figure 7.2: Types of data fitting: Left image shows an ideal fitting, the middle image has overfitting problem and the right one has underfitting problem

There are several reasons which lead to this bad performance and thus many ways to tackle it. In Deep Learning and especially in Medical Imaging the major problem is the lack of enough datasets. There are only a few datasets and they provide a small number of annotated images. So the model has not enough data to be trained and without addressing this issue with some techniques, this leads to poor performance.

The most common ways to confront these problems are augmentation, Drop Out, cross validation, regularization and early stopping.

By the term 'augmentation' we mean that we augment or increase the amount of data in order to reduce "overfitting". There are two ways to do this: By exploiting the existing data and altering them, and the second way is to create new synthetic data from scratch.

In [31, 32] they create synthetic data with Generative Adversarial Networks (GANs). GANs comprise two networks which cooperate in order to generate synthetic images as mentioned in 5.6.

There is a simpler way to avoid creating a whole network to tackle overfitting. Augmentation may be applied to already created patches or to the whole image. In Python (specifically Keras) there are functions (eg. ImageDataGeneretor) which can alter and distort the existing images. On the other hand, many studies create their own custom functions. The rationale behind this is that a model must learn to recognize a lesion no matter the pose it has.

In [9, 12, 22, 13, 11, 44, 16, 24, 14, 25, 29, 15, 36] they apply this technique. There are several ways to alter an image: An image can be cropped, zoomed in and out [12], mirrored [44] and the most common is to flip horizontal, vertical, 180, 270 degrees the image [9, 22, 47, etc.]. There are also cases that change the colors of the image for further augmentation.

Chapter 8

Metrics to evaluate the performance of a model

There are several metrics used to evaluate the performance of a model. Some of them are dedicated for image segmentation and others for general classification tasks.

The deviation of the prediction from ground truth helps in precise evaluation of the model. By knowing the ground truth, predictions are classified as following: When a true lesion has been classified correctly then it is a true positive (TP). In case the model believes that it is a lesion but in reality it is not, we are talking about a false positive (FP). On the other side, when a region has been predicted not to indicate a lesion, correctly, we are talking about a true negative (TN), and in case the model fails to detect a lesion in a region, we are talking about a false negative (FN).

Having in mind those definitions, we define as:

$$\text{Sensitivity, recall, True positive rate } (TPR) = \frac{TP}{FN + TP} \tag{8.1}$$

$$\text{Specificity, true negative rate } (TNR) = \frac{TN}{FP + TN} \tag{8.2}$$

$$\text{Precision, positive predictive value } (PPV) = \frac{TP}{FP + TP} \tag{8.3}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{8.4}$$

Those metrics are most used in more general classification tasks but they can also be used to evaluate the detection of lesions. Most studies don't rely on one metric but use several to have a better understanding about the model.

More dedicated to segmentation metrics are:

The Intersection-over-Union (IoU), reflects the degree of coincidence between the predicted values (P) and the original ground truth (G):

$$IoU = \frac{P \cap G}{P \cup G} \tag{8.5}$$

The dice coefficient is a measure of similarity that takes into account the overlap between the predicted values (P) and the ground truth (G). It is commonly used to assess segmentation performances

$$\begin{aligned}\text{Dice coefficient, F1 score} &= \frac{2 * |P \cap G|}{|P| + |G|} \\ &= \frac{2 * TP}{2 * TP + FP + FN}\end{aligned} \tag{8.6}$$

Besides the values provided by metrics, special curves can be plotted in order to have an intuition and visualization about how well the model works.

The ROC curve is created by plotting the true positive rate (TPR) against the false positive rate ($FPR = 1 - TNR$) at various threshold settings. “Area Under the ROC Curve” (AUC) provides an intuition about how well the model classifies. Specifically AUC estimates the probability that the model ranks a random positive example more highly than a random negative example. The AUC values range from zero to one, with zero and 100% accuracy respectively.

There special curves which are deserved for segmentation tasks: A Free-response Receiver Operating Characteristic (FROC) curve is a tool for characterizing the performance of a free-response system at all decision thresholds simultaneously. It displays the possible tradeoff between the sensitivity against the average number of false positive detection per image.

Chapter 9

Organizing our Experiment

The model in Deep Learning tasks, is trained on annotated data and predicts new unseen images. In our case we will follow the pipeline shown in Figure 9.1.

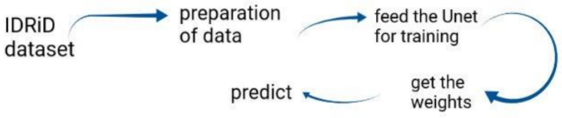


Figure 9.1: The pipeline we will follow to predict the lesions

We started by preparing our dataset.This is a crucial part of the experiment and dedicated the most time compared to the rest of the procedure..Afterwards as shown in Figure 9.1 we trained our model and we were able to predict the lesions.

9.1 Data preprocessing

9.1.1 IDRiD dataset

IDRiD dataset [8] is one of the best public datasets concerning annotations of DR. The fundus images come from an ophthalmologist at an Eye Clinic located in Nanded, Maharashtra, India. From all the acquired images only 516 were chosen as the best and most informative to be included in the dataset.

Images were acquired using a Kowa VX-10 alpha digital fundus camera with 50-degree field of view (FOV), and all are centered near to the macula. The images have a resolution of 4288 × 2848 pixels and are stored in jpg file format. The size of each image is about 800 KB.

The pixel level annotated data include 81 color fundus images which appear at least one DR lesion. A binary mask shows with pixel level accuracy the precise lesion. The annotation is executed for all four lesions and resulted in : 81 images for microaneurysms (MA), 40 images for soft exudates (SE), 81 images for hard exudates (EX) and 80 images for hemorrhages (HE), totally 282 images.

9.1.2 Rationale behind preprocessing

The main concern in our experiment was to achieve a good performance. One of the things we had to pay attention in preprocessing stage was the overfitting problem. We had to begin with the available 282 images. The size of the images (4288 × 2848) would not fit to the memory though. As mentioned in previous chapter UNETS are ideal to work with a limited number of training images. Nevertheless, we decided to increase the number of images synthetically, by creating patches ,taking as base those 282 images.

Moreover the patches were created from sliding windows so that the final number of images increased dramatically. We decided to work with 512×512 size of each patch, according to [51]. The sliding step was 64 pixels in all datasets except for SE which we reduced to 32 to have an equal number of final patches in each lesion.

According to this plan we ended up with 140,000 patches of EX, 140,000 patches of HM, 140,000 patches of MA and 142,000 patches of SE. From all those patches we set a threshold for each lesion to keep the most informative patches and discard the redundant ones. So for:

- Exudates (EX) the threshold was set to keep 5 % and more of the lesions in images. Thus the final number of informative EX patches was reduced to 17,016.
- Hemorrhages (HM) the threshold was set to keep 5 % and more of the lesions in images. Thus the final number of informative HM patches was reduced to 19,536.
- Microaneurysms (MA) the threshold was set to keep 1 % and more of the lesions in images. Thus the final number of informative MA patches was reduced to 17,168.

- Soft Exudates (SE) the threshold was set to keep 1 % and more of the lesions in images. Thus the final number of informative SE patches was reduced to 12,944.

Figure 9.2 shows an example on how the results we expect to get.

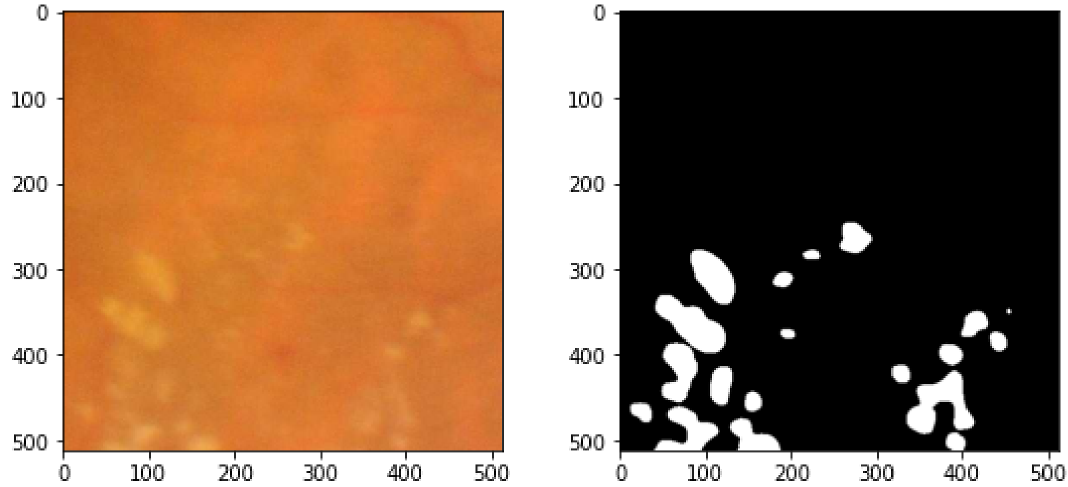


Figure 9.2: We can see an example of what kind of images we feed the network. The left one is the original fundus image, while the right one is the groundtruth corresponding to exudates.

So as an example we can see two patches from Exudates dataset. The left one is the image and the right one is the binary mask corresponding to exudates.

We split each individual dataset with a ratio of 80% training and 20% testing. As mentioned in the previous section we did not apply any special Image Processing technique. This was due to the fact that IDRiD was a qualitative dataset compared to the other publicly available datasets both in image as well as annotation level. The only additional step was to scale the data by dividing each pixel by 255.

9.2 Training the model

9.2.1 MobileNETV2

MobileNetV2 [7] is an architecture most used in mobile devices. It surpasses the state of the art performance of benchmarks and mobile models on multiple tasks.

In this model there are 2 types of blocks: residual blocks with stride of 1 and another block with stride of 2. There are three layers for these blocks. The first layer is a 1 × 1 convolution, the second is the depthwise convolution and the last is another 1 × 1 convolution without nonlinearity, as seen in Table 9.1.

Table 9.1: The architecture of the MobileNetV2, where t is the expansion factor, c is the number of the output channels, n is the repeating number and s is the stride.
Image taken from `https://medium.com/@luis_gonzales/a-look-at-mobilenetv2-inverted-residuals-and-linear-bottlenecks-d49f85c12423`

Input	**Operator**	t	c	n	s
$224^2 \times 3$	conv2d	–	32	1	2
$112^2 \times 32$	bottleneck	1	16	1	1
$112^2 \times 16$	bottleneck	6	24	2	2
$56^2 \times 24$	bottleneck	6	32	3	2
$28^2 \times 32$	bottleneck	6	64	4	2
$14^2 \times 64$	bottleneck	6	96	3	1
$14^2 \times 96$	bottleneck	6	160	3	2
$7^2 \times 160$	bottleneck	6	320	1	1
$7^2 \times 320$	conv2d 1 × 1	–	1280	1	1
$7^2 \times 1280$	avgpool 7 × 7	–	–	1	–
$1 \times 1 \times 1280$	conv2d 1 × 1	–	k	–	

Figure 9.3 shows the MobileNetV2 full architecture with all layers.

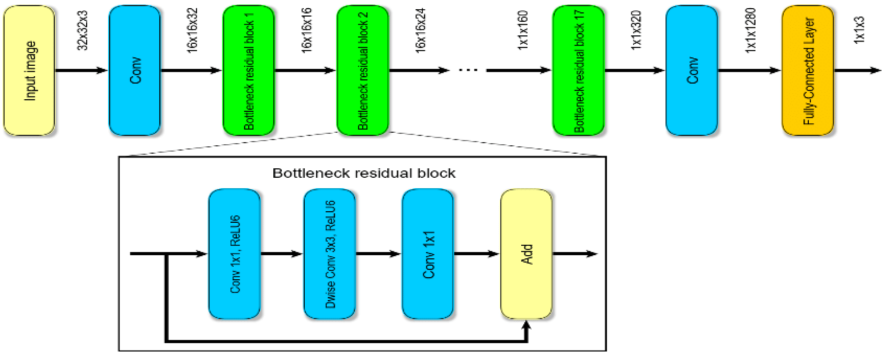


Figure 9.3: MobileNetV2 full architecture with all layers
Image taken from `https://www.mdpi.com/1424-8220/20/14/3856`

Generally MobileNetV2 is less time consuming to train because it has less parameters to train and thus the model converges faster and with higher accuracy compared with a model trained from scratch.

9.2.2 Architecture and hyper parameters

The model we use, as previously mentioned, is based on a Unet with the difference that the encoder is replaced with MobileNetV2 architecture, omitting the last Fully Connected Layer. Additionally the encoder is pretrained on ImageNet. Figure 9.4 shows the philosophy of our architecture.

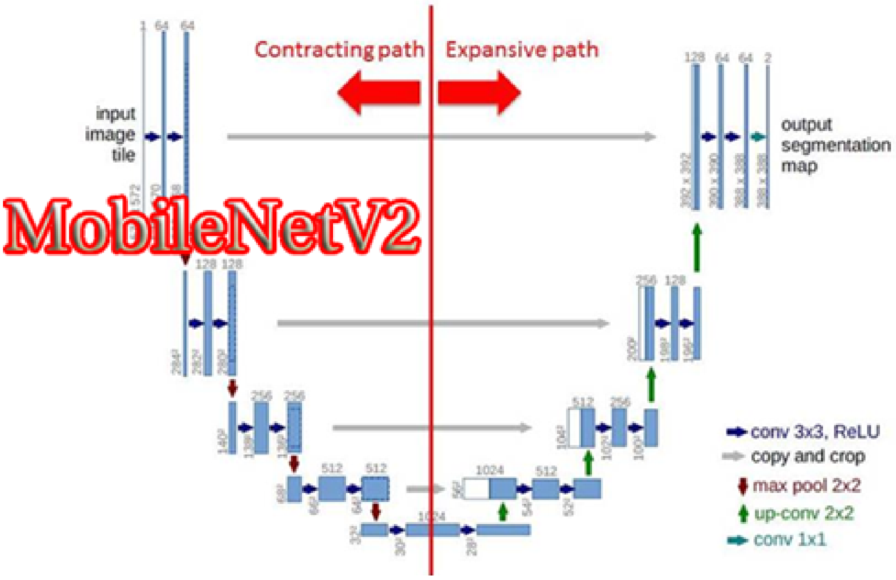


Figure 9.4: The philosophy of our architecture. The whole left part is which corresponds to the encoder of the U-Net, is replaced by MobiletV2

Except for the architecture, there are other hyperparameters which have been tuned to improve the performance.

Loss function measures the discrepancy of the predicted labels from the ground truth and it is used during training of the model. In every backpropagation the weights are updated and after the appropriate epochs, training stops because there is convergence of the loss function. We used binary cross entropy as a loss function which has the formula of the equation (9.1).

$$\text{Binary cross entropy} = -\frac{1}{N}\sum_{i=1}^{N} \widehat{y_i} + (1 - y_i)\log(1 - \widehat{y_i}) \tag{9.1}$$

where $\widehat{y_i}$ is the i-th scalar value in the model output, y_i is the corresponding target value, and output size is the number of scalar values in the model output.

As previously explained to prevent overfitting we used the augmentation technique but only for training set. Specifically we used the ImageDataGenerator class from Keras and set the following parameters: rescale = 1/255 to scale the data, shear range = 0.2 (to create sheared images), zoom range = 0.2 (to create images with zoom effect) and horizontal flips = True (to create horizontally flipped images).

The learning rate was set to 0.0001 for all the experiments. For the optimization of the loss function we used Adam optimizer with the default settings: $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The batch size we chose was set to 8 for all our experiments. The epochs we used differed for each experiment and did not use early stopping, to the contrary to reach the best possible accuracy. So in Ex the epochs were 70, in HM there were 50, in MA 100 and in SE 50.

Chapter 10

Results

The whole study consists of 4 experiments, each one concerning a lesion. In the following section we have gathered all the results and we present them synoptically.

10.1 Pixel level-Metrics

Table 10.1 shows the metrics from each experiment. A thing worth mentioning is that the specificity is high due to the imbalance of the dataset. Thus, it correctly predicts most of the negative class (normal tissue) in all test sets. Note that all the experiments are executed in the IDRiD test set.

Table 10.1: The metrics from each experiment.

Metric	Exudates	Hemorrhages	Microaneurysms	Soft Exudates
Dice coefficient	0.83	0.85	0.85	0.95
Recall	0.85	0.89	0.84	0.97
Precision	0.90	0.91	0.95	0.96
Sensitivity	0.86	0.89	0.844	0.97
Specificity	0.99	0.99	0.999	0.99

10.2 Pixel level -Confusion Matrices

The next tables show the confusion matrix in pixel level for each lesion. If we sum all the TP, TN, FP, FN, the result represents the total number of pixels of the test set.

Tables 10.2–10.5 show the confusion matrices of each experiment

Table 10.2: Confusion Matrix of Exudates experiment

Exudates Confusion Matrix	**Actually Positives**	**Actually Negatives**
Predicted Positives	True Positives= 58,951,560	False Positives= 5,894,475
Predicted Negatives	False Negatives= 9,734,064	True Negatives= 816,708,992

Table 10.3: Confusion Matrix of Hemorrhages experiment

Hemorrhages Confusion Matrix	**Actually Positives**	**Actually Negatives**
Predicted Positives	True Positives= 61,104,796	False Positives= 5,422,910
Predicted Negatives	False Negatives= 7,014,877	True Negatives= 951,964,864

Table 10.4: Confusion Matrix of Microaneurysms experiment

Microaneurysms Confusion Matrix	**Actually Positives**	**Actually Negatives**
Predicted Positives	True Positives= 4,661,535	False Positives= 226,894
Predicted Negatives	False Negatives= 857,627	True Negatives= 869,029,504

Table 10.5: Confusion Matrix of Soft Exudates experiment

Soft Exudates Confusion Matrix	**Actually Positives**	**Actually Negatives**
Predicted Positives	True Positives= 27,799,908	False Positives= 1,123,113
Predicted Negatives	False Negatives= 744,409	True Negatives= 649,809,728

10.3 Lesion level- Metrics

In our opinion it would be more informative to know how well the model predicts on lesion level and not on every single pixel. To this end, we set the threshold to 0.5. This means that if the model predicts more than 50 % of the pixels of a lesion, then we count as one lesion. In the opposite case we do not accept it as a lesion. So, we measure how many lesions the model has predicted (predicted labels) and how many of them are indeed lesions (true labels). Further we measure how many of the predicted lesions are indeed lesions (true positives), how many of the predicted lesions are indeed healthy tissues (false positives), how many of the lesions are not predicted (false negatives) and finally we measure the mean Intersection over Union (IoU) for all the test set.

So we with this rationale we have the following results shown in table 10.6:

Table 10.6: Metrics of the experiments in lesion level

Results	**Exudates** **(3400 patches)**	**Hemorrhages** **(3912 patches)**	**Microaneurysms** **(3440 patches)**	**Soft Exudates** **(2592 patches)**
Predicted labels	82,973	12,763	17,686	3,290
True labels	77,493	19,424	17,234	3,904
True positives	55,125	8,069	15,867	3,091
False positives	20,627	3,145	842	131
False negatives	4,392	5,391	424	592
Sensitivity	0.9262	0.5994	0.9739	0.8392
Precision	0.7277	0.7195	0.9513	0.9593
Mean IoU	0.7463	0.7794	0.7539	0.9269

10.4 Comparison of pixel and lesion level analysis

Comparing the pixel and lesion level analysis we can see that the results mainly converge in Precision metric : SEs and MAs have the best performance. In contrast in Sensitivity metric the results do not converge: In pixel level SEs have the best performance and MAs the worst. In contrast in lesion level MAs have the best performance.

This will be explained directly in the next paragraph. Finally dice coefficient and IoU mainly converge in both pixel and lesion analysis with SE having the best performance and EXs the worst.

The results in pixel and lesion level do not agree, in our opinion, due to the following reason.

Figure 10.1 shows two circles: A and B. Circle A is a lesion with the red area predicted which covers the 25% of the lesion. The dotted line shows the threshold of 50% of the area. Circle B is the groundtruth of the lesion, which covers 100% of the area as we see. In this case we have $TP = 25$, $TN = 0$, $FN = 75$, $FP = 0$.

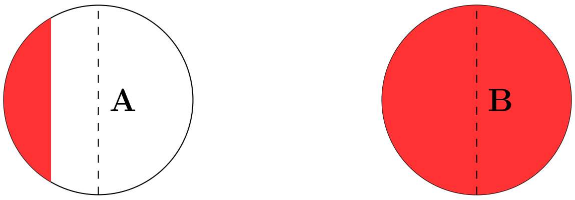


Figure 10.1: Circle A is a lesion with the red color showing the predicted area. Circle B is the groundtruth

In pixel level the sensitivity according to (8.1) is 0.25 while Precision, according to (8.3), is also 1.

In lesion level if we set the threshold to 50% then there is no lesion predicted, thus the Sensitivity is zero and Precision does not have any meaning. So if we have a large lesion it is difficult to predict more than 50 %, compared to a smaller lesion area.

Returning to our previous question why Sensitivity is better in smaller lesions, in lesion level, we can clearly understand now the answer. The areas in MAs are clearly smaller compared to other lesions, so the threshold does not affect Sensitivity so much. In contrast in larger lesions like HMs the threshold makes things harder.

10.5 Curves of the performances

Finally we can see the performance of the model during the whole training. The left figures show the dice coefficient, and the right show the model loss. As we can infer from all figures there is no overfitting during our training. Another phenomenon which is present in all our experiments is the sudden drops of the performance, which happened several times during training. The known "Black Box" issue which characterizes Deep Learning, does not let us have an intuition about the reason for this strange behaviour.

Figures 10.2, 10.3, 10.4 and 10.5 shows the performance in EX, HM, MA, SE experiment respectively. The left image shows the dice loss progress through the epochs, and the right show the loss in each experiment.

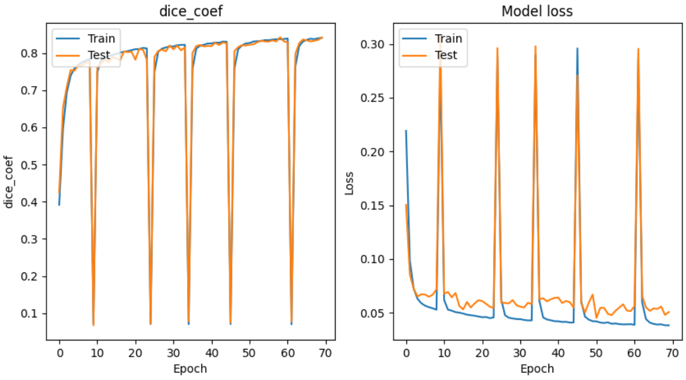


Figure 10.2: Exudates experiment

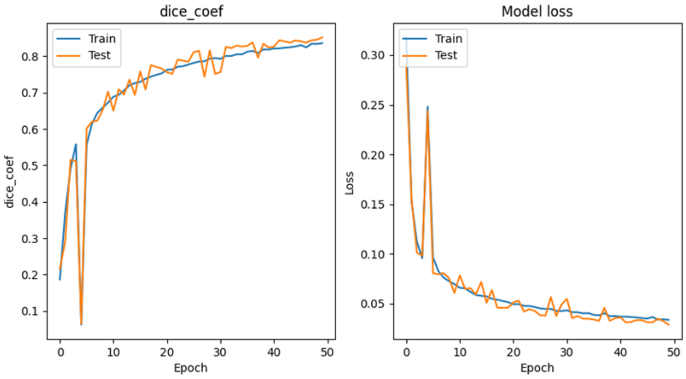


Figure 10.3: Hemorrhages experiment

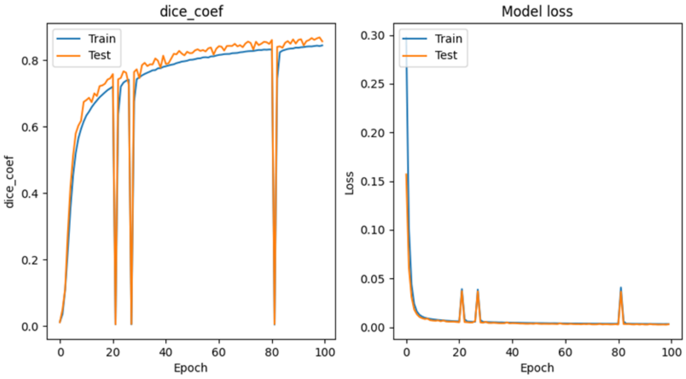


Figure 10.4: Microaneurysm experiment

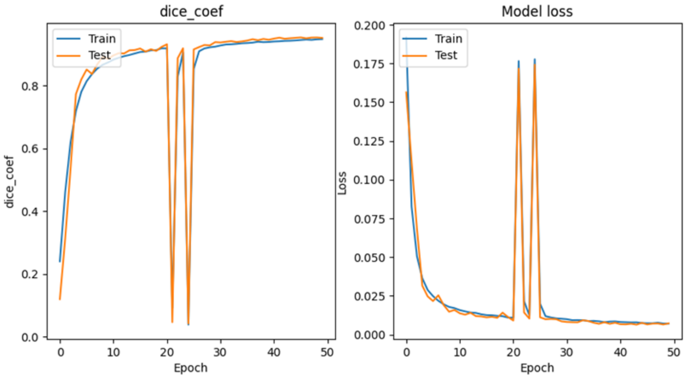


Figure 10.5: Soft Exudayes experiment

10.6 Visual representation of predictions

The performance of the dataset is examined visually in two datasets. The first one is the known test set IDRiD and the second is DIARETDB1, a dataset that the algorithm has never seen before. We will also see how the algorithm predicts both individual patches as well as whole images.

Figure 10.6 shows how the algorithm predicts patches of the known IDRiD dataset. The first image on the left is the image which will be passed to the algorithm, the second image is the predicted image and the last image is the ground truth. We can infer that the good performance of the model is also reflected on the visualization.

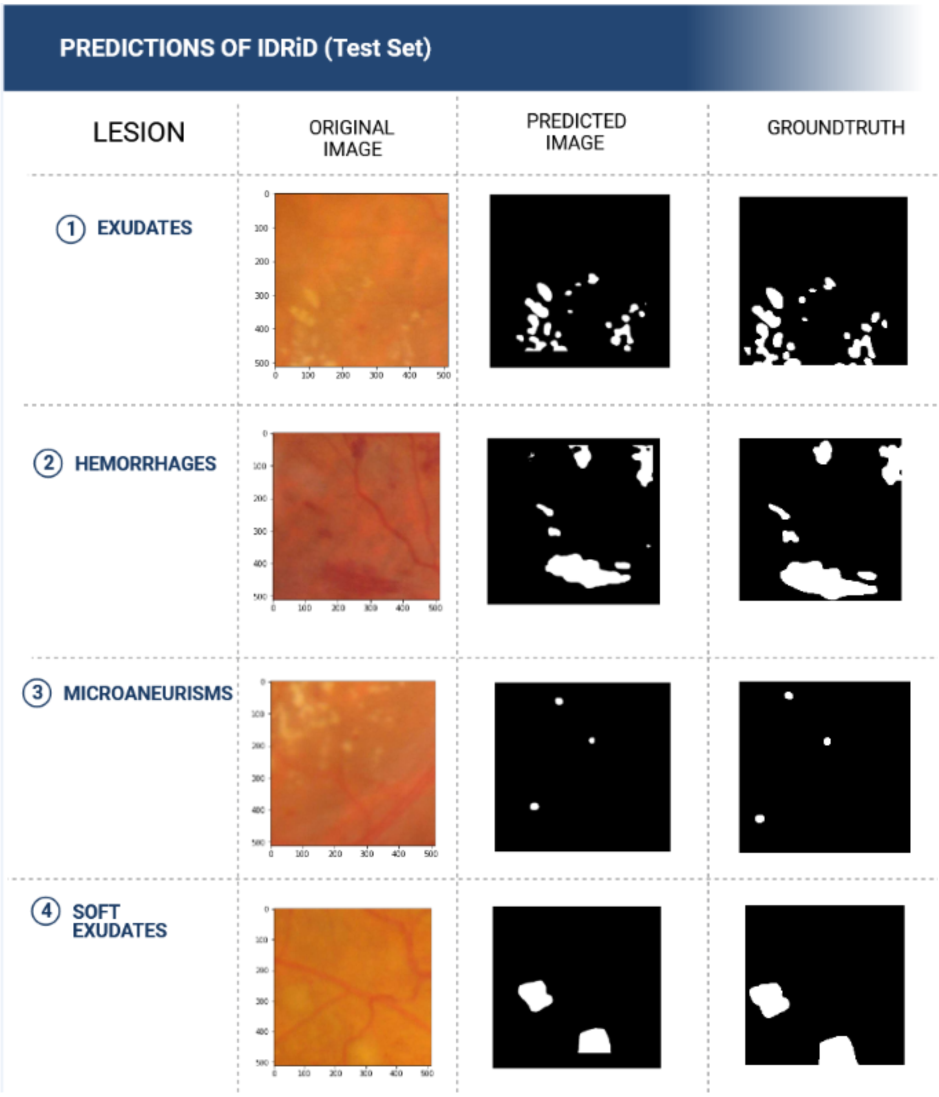


Figure 10.6: Visualization of our results: The image on the left is the fundus image, the second image is the predicted image and the last image is the ground truth. Note that the testing images belong to IDRiD, a dataset which our algorithm has already seen

Figure 10.7 shows the predictions on a new unseen dataset (DIARETDB1). The groundtruth is not with the same accuracy as the IDRiD dataset. There are not exact borders of the lesion but instead area which surrounds the lesion. Despite this fact, we can understand visually that out model has a good accuracy.

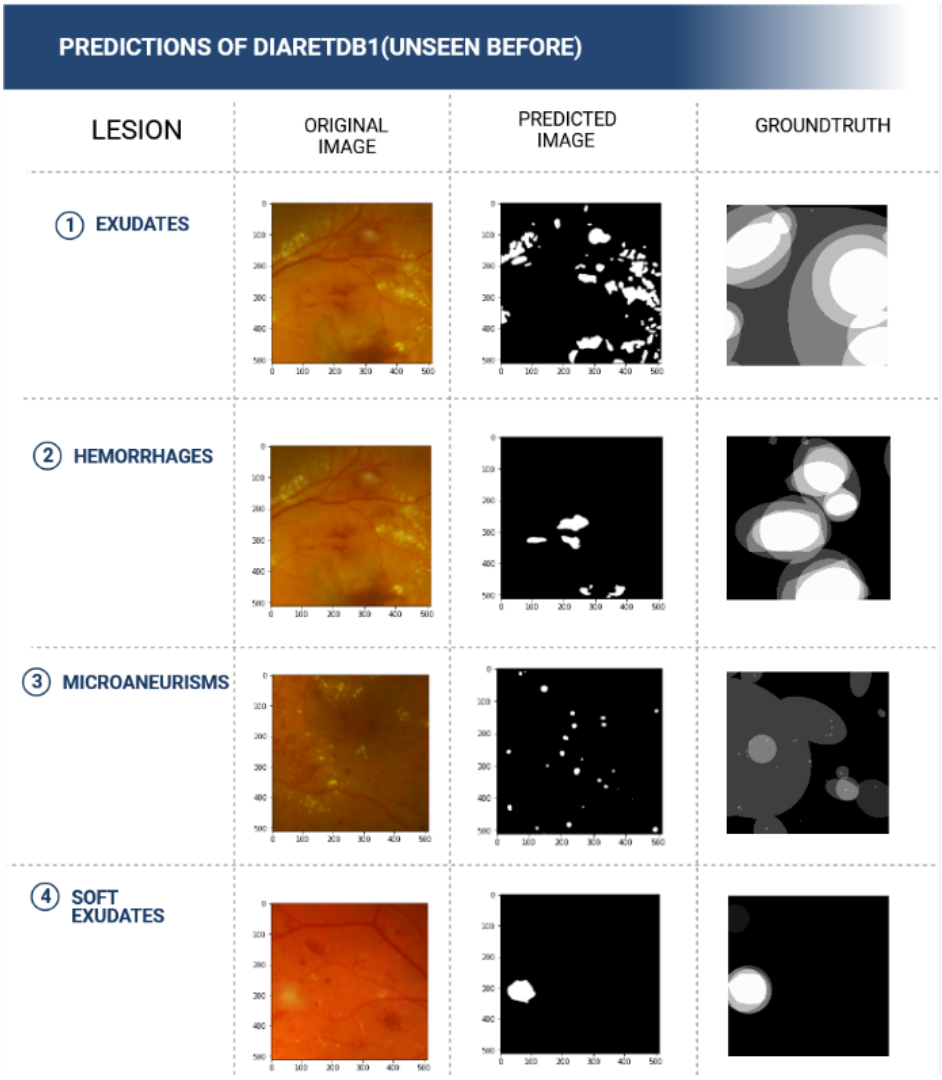


Figure 10.7: Visualization of our results in unseen dataset: The image on the left is the fundus image, the second image is the predicted image and the last image is the ground truth. The testing images belong to DIARETDB1 which our algorithm has never seen before. One can notice the weird groundtruth of this dataset, but our predictions belong to the area of groundtruth

Figure 10.8 shows the predictions holistically and not on single patches. The lesions we predict are the exudates of an entire image.

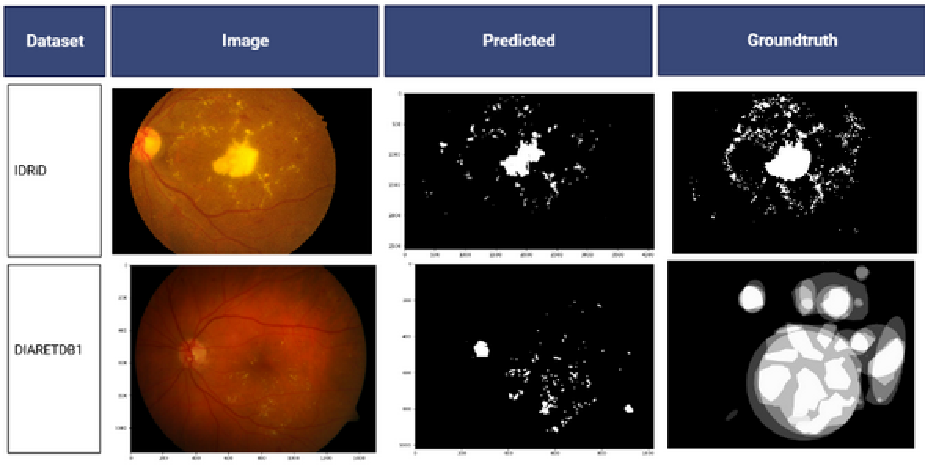


Figure 10.8: Predictions of Exudates in whole images instead of using patches

Finally we have combined all lesions and given a pseudo color for better visualization. Exudates are colored magenta, Hemorrhages cyan, Microaneurysms blue and Soft Exudates yellow. Final result emerges from this combination. It is important to note that this is not multiclass segmentation, where each pixel corresponds to one lesion only. It is just a visual representation of the whole experiment as seen in Figure 10.9.

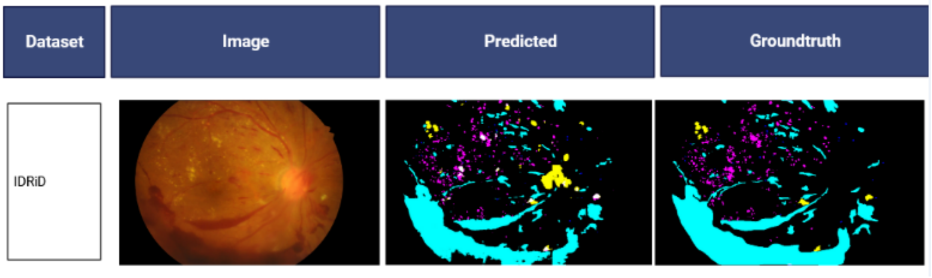


Figure 10.9: Visual representation of the results, combining all lesions in one image, using pseudocolors

10.7 Compare to state of art

Finally we compare our results with the results of our literature review. The comparison is in pixel level and not in lesion level. The inferences of our literature review, as earlier mentioned, showed that Unets and FCN are very robust models. Except from these models, some of the older ones based on traditional CNN architectures, showed very good performances too. So we did a comparison taking into account all those parameters. Table 10.7 shows that the Sensitivities of Hemorrhages and Soft Exudates of our model surpass the existing state of art models. More specifically, Sensitivity in HMs reached 0.89 whilst in SEs reached 0.97. In contrast, our Sensitivity in EXs (0.86) could not reach the 0.99 of [23], as well as in MAs our Sensitivity (0.844) could not surpass the 0.87 of [51].

Table 10.7: Metrics in the state-of-art papers according to our literature review. Red color indicates the highest sensitivity in each lesion

AUTHOR	**YEAR**	**EXUDATES SENSITIVITY**	**HEMORRHAGES SENSITIVITY**	**MICROANEYRISMS SENSITIVITY**	**SOFT EXUDATES SENSITIVITY**	**ARCHITECTURE**
OURS	2021	0.86	**0.89**	0.844	**0.97**	UNET WITH PRETRAINED ENCODER
[**37**]	2021	0.94	0.87	0.48	0.87	DEEPLABV3, FCN, UNET
[**31**]	2018	0.94	-	-	-	UNET + cGAN
[**32**]	2018	0.92	-	-	-	UNETs
[**23**]	2017	**0.99**	-	-	-	CNN(Le-Net)
[**9**]	2018	0.96	0.84	0.85	-	CNN
[**17**]	2018	0.84	-	-	-	UNET
[**35**]	2019	0.89	-	-	-	FCN
[**51**]	2018	0.88	0.72	**0.87**	0.77	UNET

Chapter 11

Discussion and future improvements

To the best of our knowledge, it is the first time that a pretrained network replaces the encoder of a UNET for semantic segmentation of DR lesions. The main goal of this study was to use state of art techniques and tackle a retinal lesions segmentation problem. This problem was substantially an automatisation of DR diagnosis.With the emergence of AI many human based tasks were substituted and in some cases like in countries where there is lack of doctors, AI can be beneficial. The algorithm we had to create could be further applied in mobile phones which have a special attached camera.

Semantic segmentation was our target, which means that we had to create an algorithm which could find and segment each type of four lesions. The problem was binary and this means that each time we can segment only one lesion. Such tasks are tackled with DL algorithms and more specifically we used UNETs which showed very promising results according to our literature review.

The major problem in many cases is the available datasets.In our case we were based on the IDRiD dataset,which is a public dataset. This dataset has qualitative images although it is imbalanced. We made a preprocessing step, based on most informative patches and used augmentation to be sure that we had a plethora of images.

Moreover due the peculiarity of our dataset we had to take it a step further and not train the model from scratch. Thus ,we used transfer learning but not in the whole part of the UNET. We used MobileNetV2 as the pretrained part of the UNET encoder.

The results of the experiment were very promising. The curves showed that there was no overfitting. This was achieved due to the correct preprocessing in our opinion.Confusion Matrices of all experiments showed that the model had generally very few FP or FN (FP in Exudates experiment were about 0.06% of the total pixels).

The Dice coefficient was over 0.83 in all lesions and in Soft Exudates was 0.95. Sensitivity is a metric which helps us understand how many of the pixels with true lesions were correctly labeled. And this is the most informative metric in our opinion. As far as Sensitivity concerns all lesions were over 0.844 and the best score was achieved in Soft Exudates (0.97).

Compared to the other state of art studies , in two lesions (SE,HM) we overpassed the existing best scores.In conclusion, gathering all the positive feedback the experiment showed that the algorithm was robust according to the comparison.

On the other hand, the results of the experiment could not be representative of the real world. The comparison between our experiment and the others was done based on different datasets in several cases. This means that this comparison may seem real but it is based upon wrong assumptions. Another problem is that the testing set is relatively small and not very representative. Finally, images acquired from fundus cameras have deficient illumination and bad image quality. So it is not obvious that our performance could be kept on testing with such mobile phones which acquire fundus images. To be more precise, in order to train an algorithm to be proficient on such image qualities, the training set must be based on such images and not on IDRiD, which contains qualitative image analysis.

Generally there is room for future improvements.If the case is to build exclusively a mobile application for semantic segmentation of DR lesions, then there is need for an appropriate dataset. Plenty of images must be acquired and skilled doctor must delineate the lesions and properly characterise them.

Exudates and Microaneurysm segmentation's performance could be further improved. A possible remedy is to use transfer learning from one lesion segmentation. In [17] they use knowledge from Microaneurysms and apply it in Exudates segmentation. In our case we could use transfer learning from Soft Exudates with very good metrics and train the model for segmenting Exudates or Microaneurysms.Another idea to improve the metrics would be to use other pretrained models besides MobilenetV2. Keras offers many pretrained models in the following table. We should choose models with as few parameters as possible in order to be light enough for further usage in mobile phones. MobileNetV2 has 3,538,984 parameters according to Table 11.1. So two candidate models could be NASNetMobile with 5,326,716 parameters and EfficientNetB0 with 5,330,571 parameters.

Table 11.1: Available mobels in keras Table taken from Keras documentation

Model	Size	Top-1 Accuracy	Top-5 Accuracy	Parameters	Depth
Xception	88 MB	0.790	0.945	22,910,480	126
VGG16	528 MB	0.713	0.901	138,357,544	23
VGG19	549 MB	0.713	0.900	143,667,240	26
ResNet50	98 MB	0.749	0.921	25,636,712	-
ResNet101	171 MB	0.764	0.928	44,707,176	-
ResNet152	232 MB	0.766	0.931	60,419,944	-
ResNet50V2	98 MB	0.760	0.930	25,613,800	-
ResNet101V2	171MB	0.772	0.938	44,675,560	-
ResNet152V2	232 MB	0.780	0.942	60,380,648	-
lnceptionV3	92 MB	0.779	0.937	23,851,784	159
lnceptionResNetV2	215 MB	0.803	0.953	55,873,736	572
MobileNet	16 MB	0.704	0.895	4,253,864	88
MobileNetV2	14 MB	0.713	0.901	3,538,984	88
DenseNet121	33 MB	0.750	0.923	8,062,504	121
DenseNet169	57 MB	0.762	0.932	14,307,880	169
DenseNet201	80 MB	0.773	0.936	20,242,984	201
NASNetMobile	23 MB	0.744	0.919	5,326,716	-
NASNetLarge	343 MB	0.825	0.960	88,949,818	-
EfficientNetB0	29 MB	-	-	5,330,571	-
EfficientNetB1	31 MB	-	-	7,856,239	-
EfficientNetB2	36 MB	-	-	9,177,569	-
EfficientNetB3	48 MB	-	-	12,320,535	-
EfficientNetB4	75 MB	-	-	19,466,823	-
EfficientNetB5	118 MB	-	-	30,562,527	-
EfficientNetB6	166 MB	-	-	43,265,143	-
EfficientNetB7	256 MB	-	-	66,658,687	-

References

[1] Yao Liu et al. “Factors influencing patient adherence with diabetic eye screening in rural communities: A qualitative study”. In: 13.11 (Nov. 2018). Ed. by Denis Bourgeois, e0206742. DOI: 10.1371/journal.pone.0206742 (page 1).

[2] Abhimanyu S. Ahuja. “The impact of artificial intelligence in medicine on the future role of the physician”. In: 7 (Oct. 2019), e7702. DOI: 10.7717/peerj.7702 (page 2).

[3] Jeremy Jordan. *An overview of semantic image segmentation.* https://www.jeremyjordan.me/semantic-segmentation/. Online; Accessed 20/08/2021. May 2018 (page 2).

[4] Avula Benzamin and Chandan Chakraborty. “Detection of Hard Exudates in Retinal Fundus Images Using Deep Learning”. In: *2018 Joint 7th International Conference on Informatics, Electronics Vision (ICIEV) and 2018 2nd International Conference on Imaging, Vision Pattern Recognition (icIVPR).* June 2018, pp. 465–469. DOI: 10.1109/ICIEV.2018.8641016 (pages 2, 4, 12, 14, 46).

[5] Nahian Siddique et al. “U-Net and Its Variants for Medical Image Segmentation: A Review of Theory and Applications”. In: *IEEE Access* 9 (2021), pp. 82031–82057. ISSN: 2169-3536. DOI: 10.1109/access.2021.3086020 (page 3).

[6] Beenish Zia, R. Illikkal, and Bob Rogers. “Use Transfer Learning for Efficient Deep Learning Training on Intel ® Xeon ® Processors”. In: 2018 (pages 3, 31).

[7] Mark Sandler et al. “MobileNetV2: Inverted Residuals and Linear Bottlenecks”. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* June 2018, pp. 4510–4520. DOI: 10.1109/CVPR.2018.00474 (pages 3, 53).

[8] Indian Diabetic Retinopathy Image Dataset (IDRiD). https://idrid.grand-challenge.org. Online; Accessed 20/08/2021. Oct. 2017 (pages 3, 51).

[9] Parham Khojasteh, Behzad Aliahmad, and Dinesh K. Kumar. "Fundus images analysis using deep features for detection of exudates, hemorrhages and microaneurysms". In: *BMC Ophthalmology* 18.1 (Nov. 2018). DOI: `10.1186/s12886-018-0954-4` (pages 4, 7, 8, 12, 13, 33, 42, 46, 48, 67).

[10] Decencière et al. "Feedback on a publicly distributed database: the Messidor database". In: *Image Analysis & Stereology* 33.3 (Aug. 2014), pp. 231–234. ISSN: 1854-5165. DOI: `http://dx.doi.org/10.5566/ias.1155` (page 6).

[11] Mark J. J. P. van Grinsven et al. "Fast Convolutional Neural Network Training Using Selective Data Sampling: Application to Hemorrhage Detection in Color Fundus Images". In: *IEEE Transactions on Medical Imaging* 35.5 (May 2016), pp. 1273–1284. DOI: `10.1109/tmi.2016.2526689` (pages 7, 8, 10, 14, 33, 42, 43, 46, 48).

[12] Carson Lam et al. "Retinal Lesion Detection With Deep Learning Using Image Patches". In: *Investigative Opthalmology & Visual Science* 59.1 (Jan. 2018), p. 590. DOI: `10.1167/iovs.17-22721` (pages 7, 8, 13, 33, 42, 46, 48).

[13] Mrinal Haloi. "Improved Microaneurysm Detection using Deep Neural Networks". In: *CoRR* abs/1505.04424 (2015). arXiv: `1505.04424 [cs.CV]`. URL: `http://arxiv.org/abs/1505.04424` (pages 7, 8, 13, 48).

[14] Jos Ignacio Orlando et al. "An Ensemble Deep Learning Based Approach for Red Lesion Detection in Fundus Images". In: *Comput. Methods Prog. Biomed.* 153.C (Jan. 2018), pp. 115–127. ISSN: 0169-2607. DOI: `10.1016/j.cmpb.2017.10.017` (pages 7, 10, 12, 15, 33, 42, 46, 48).

[15] Noushin Eftekhari et al. "Microaneurysm detection in fundus images using a two-step convolutional neural network". In: *BioMedical Engineering OnLine volume* 18.67 (May 2019). DOI: `10.1186/s12938-019-0675-9` (pages 7, 8, 15, 33, 42, 43, 46, 48).

[16] Waleed M. Gondal et al. "Weakly-supervised localization of diabetic retinopathy lesions in retinal fundus images". In: *CoRR* abs/1706.09634 (June 2017). arXiv:

1706.09634 [cs.CV]. URL: http://arxiv.org/abs/1706.09634 (pages 7, 8, 12, 14, 33, 42, 46, 48).

[17] Piotr Chudzik et al. "Exudate segmentation using fully convolutional neural networks and inception modules". In: *Medical Imaging 2018: Image Processing.* Ed. by Elsa D. Angelini and Bennett A. Landman. SPIE, Mar. 2018. DOI: 10.1117/12.2293549 (pages 7, 9, 13, 42, 46, 67, 69).

[18] Piotr Chudzik et al. "Microaneurysm detection using fully convolutional neural networks". In: *Computer Methods and Programs in Biomedicine* 158 (May 2018), pp. 185–192. DOI: 10.1016/j.cmpb.2018.02.016 (pages 7, 14, 42, 46).

[19] Parham Khojasteh et al. "Exudate detection in fundus images using deeply-learnable features". In: *Computers in Biology and Medicine* 104 (Jan. 2019), pp. 62–69. DOI: 10.1016/j.compbiomed.2018.10.031 (pages 7, 9, 14, 42).

[20] Balazs Harangi, Janos Toth, and Andras Hajdu. "Fusion of Deep Convolutional Neural Networks for Microaneurysm Detection in Color Fundus Images". In: *Annual International Conference of the IEEE Engineering in Medicine and Biology Society.* IEEE Engineering in Medicine and Biology Society, July 2018, pp. 3705–3708. DOI: 10.1109/embc.2018.8513035 (pages 7, 9, 15).

[21] Jie Xue et al. "Deep membrane systems for multitask segmentation in diabetic retinopathy". In: *Knowledge-Based Systems* 183 (July 2019), p. 104887. DOI: 10.1016/j.knosys.2019.104887 (pages 7, 8, 10, 13).

[22] Oindrila Saha, Rachana Sathish, and Debdoot Sheet. "Fully Convolutional Neural Network for Semantic Segmentation of Anatomical Structure and Pathologies in Colour Fundus Images Associated with Diabetic Retinopathy". In: *CoRR* abs/1902.03122 (2019). arXiv: 1902.03122 [cs.CV]. URL: http://arxiv.org/abs/1902.03122 (pages 7, 9, 13, 42, 48).

[23] Oscar Perdomo, John Arevalo, and Fabio A. González. "Convolutional network to detect exudates in eye fundus images of diabetic subjects". In: *12th International Symposium on Medical Information Processing and Analysis.* Ed. by

Eduardo Romero et al. SPIE, Jan. 2017. DOI: 10.1117/12.2256939 (pages 7, 8, 12, 14, 33, 42, 66, 67).

[24] Gwenolé Quellec et al. “Deep image mining for diabetic retinopathy screening”. In: *Medical Image Analysis* 39 (July 2017), pp. 178–193. ISSN: 1361-8415. DOI: 10.1016/j.media.2017.04.012 (pages 7, 8, 12, 14, 46, 48).

[25] Juan Sebastian Otálora et al. “Training Deep Convolutional Neural Networks with Active Learning for Exudate Classification in Eye Fundus Images”. In: *Intravascular Imaging and Computer Assisted Stenting, and Large-Scale Annotation of Biomedical Data and Expert Label Synthesis.* Vol. 10552. Sept. 2017, pp. 146–154. DOI: 10.1007/978-3-319-67534-3_16 (pages 7, 8, 10, 15, 33, 42, 48).

[26] Caixia Kou et al. “Microaneurysms segmentation with a U-Net based on recurrent residual convolutional neural network”. In: *Journal of Medical Imaging* 6.02 (June 2019), p. 1. DOI: 10.1117/1.JMI.6.2.025008 (pages 7, 10, 15, 38, 42, 46).

[27] Natàlia Gullón. “Retinal lesions segmentation using CNNs and adversarial training”. PhD thesis. Barcelona: Polytechnic University of Catalonia, July 2018 (pages 7, 11, 15, 38, 42).

[28] Qiqi Xiao et al. “Improving Lesion Segmentation for Diabetic Retinopathy Using Adversarial Learning”. In: *Image Analysis and Recognition.* Ed. by Fakhri Karray. Vol. 11663. Cham, Switzerland: Springer International Publishing, Aug. 2019, pp. 333–344. ISBN: 978-3-030-27271-5. DOI: 10.1007/978-3-030-27272-2_29 (pages 7, 11, 16, 42, 46).

[29] Ashutosh Kushwaha and P. Balamurugan. “Classifying Diabetic Retinopathy Images Using Induced Deep Region of Interest Extraction”. In: *2019 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI).* Oct. 2019, pp. 1–6. DOI: 10.1109/CISP-BMEI48845.2019.8965695 (pages 7, 8, 15, 38, 42, 46, 48).

[30] Yehui Yang et al. “Lesion Detection and Grading of Diabetic Retinopathy via Two-Stages Deep Convolutional Neural Networks”. In: *Medical Image Computing and Computer Assisted Intervention — MICCAI 2017*. Cham: Springer International Publishing, Sept. 2017, pp. 533–540. ISBN: 978-3-319-66179-7. DOI: 10.1007/978-3-319-66179-7_61 (pages 7, 16).

[31] Rui Zheng et al. “Detection of exudates in fundus photographs with imbalanced learning using conditional generative adversarial network”. In: *Biomedical optics express* 9.10 (Sept. 2018), pp. 4863–4878. DOI: 10.1364/boe.9.004863 (pages 7, 12, 15, 38, 42, 43, 46, 48, 67).

[32] Appan K. Pujitha and J. Sivaswamy. “Image analysis and recognition: 15th International Conference, ICIAR 2018, Póvoa de Varzim, Portugal, June 27-29, 2018, Proceedings”. In: *ICIAR 2018. Lecture Notes in Computer Science.* Ed. by A. Campilho, F. Karray, and B ter Haar Romeny. 10882. Cham, Switzerland: Springer, Cham, June 2018. ISBN: 978-3-319-92999-6. DOI: 10.1007/978-3-319-93000-8_70 (pages 7, 12, 15, 38, 42, 46, 48, 67).

[33] Lizong Zhang et al. “Detection of Microaneurysms in Fundus Images Based on an Attention Mechanism”. In: *Genes (Basel)* 10.10 (Oct. 2019), p. 817. DOI: 10.3390/genes10100817 (pages 7, 11, 12, 15, 46).

[34] Fisher Yu and Vladlen Koltun. *Multi-Scale Context Aggregation by Dilated Convolutions*. Version 3. Apr. 2016. arXiv: 1511.07122v3 [cs.CV] (pages 7, 12).

[35] Ze Si et al. “Hard exudate segmentation in retinal image with attention mechanism”. In: *IET Image Processing* 15.3 (July 2020), pp. 587–597. DOI: 10.1049/ipr2.12007 (pages 7, 11, 12, 16, 42, 67).

[36] Umit Budak et al. “A novel microaneurysms detection approach based on convolutional neural networks with reinforcement sample learning algorithm”. In: *Health Information Science and Systems* 5.14 (Nov. 2017). DOI: 10.1007/s13755-017-0034-9 (pages 7, 11, 16, 33, 42, 46, 48).

[37] Pedro Furtado. “Using Segmentation Networks on Diabetic Retinopathy Lesions: Metrics, Results and Challenges”. In: *Proceedings of the 14th International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2021)*. Vol. 2. BIOIMAGING. SCITEPRESS - Science and Technology Publications, 2021, pp. 128–135. ISBN: 978-989-758-490-9. DOI: 10.5220/0010208501280135 (pages 9, 12, 16, 42, 67).

[38] Liang-Chieh Chen et al. *Rethinking Atrous Convolution for Semantic Image Segmentation*. 2017. arXiv: 1706.05587 [cs.CV] (page 9).

[39] Jonathan Long, Evan Shelhamer, and Trevor Darrell. *Fully Convolutional Networks for Semantic Segmentation*. 2015. arXiv: 1411.4038 [cs.CV] (pages 9, 38).

[40] Qijie Wei et al. “Learn to Segment Retinal Lesions and Beyond”. In: *25th International Conference on Pattern Recognition (ICPR)*. IEEE, Dec. 2020. DOI: 10.1109/ICPR48806.2021.9412088 (pages 9, 16).

[41] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. *SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation*. 2016. arXiv: 1511.00561 [cs.CV] (page 9).

[42] Ian Goodfellow et al. “Generative Adversarial Nets”. In: *Advances in Neural Information Processing Systems*. Ed. by Z. Ghahramani et al. Vol. 27. Curran Associates, Inc., 2014. URL: https://proceedings.neurips.cc/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf (pages 11, 41).

[43] Alan Fleming et al. “The role of haemorrhage and exudate detection in automated grading of diabetic retinopathy”. In: *The British journal of ophthalmology* 94 (Sept. 2009), pp. 706–711. DOI: 10.1136/bjo.2008.149807 (page 12).

[44] Pavle Prentasic and Sven Loncaric. “Detection of exudates in fundus photographs using convolutional neural networks”. In: *2015 9th International Symposium on Image and Signal Processing and Analysis (ISPA)*. IEEE, Sept. 2015, pp. 188–192. DOI: 10.1109/ispa.2015.7306056 (pages 12, 14, 33, 42, 48).

[45] Pedro Furtado. “Segmentation of Diabetic Retinopathy Lesions by Deep Learning: Achievements and Limitations”. In: *7th International Conference on Bioimaging*. SCITEPRESS - Science and Technology Publications, Jan. 2020, pp. 95–101. DOI: `10.5220/0008881100950101` (pages 12, 16, 42).

[46] Jen Hong Tan et al. “Automated Segmentation of Exudates, Haemorrhages, Microaneurysms Using Single Convolutional Neural Network”. In: *Inf. Sci.* 420.C (Dec. 2017), pp. 66–76. ISSN: 0020-0255. DOI: `10.1016/j.ins.2017.08.050` (page 13).

[47] Syed Ali Gohar Naqvi, Muhammad Faisal Zafar, and Ihsan ul Haq. “Referral system for hard exudates in eye fundus”. In: *Computers in Biology and Medicine* 64 (Sept. 2015), pp. 217–235. ISSN: 0010-4825. DOI: `10.1016/j.compbiomed.2015.07.003` (pages 14, 48).

[48] Worapan Kusakunniran et al. “Hard exudates segmentation based on learned initial seeds and iterative graph cut”. In: *Computer Methods and Programs in Biomedicine* 158 (May 2018), pp. 173–183. DOI: `10.1016/j.cmpb.2018.02.011` (pages 14, 46).

[49] Juan Shan and Lin Li. “A Deep Learning Method for Microaneurysm Detection in Fundus Images”. In: *2016 IEEE First International Conference on Connected Health: Applications, Systems and Engineering Technologies (CHASE)*. IEEE, June 2016, pp. 357–358. DOI: `10.1109/CHASE.2016.12` (pages 14, 42, 46).

[50] Shuang Yu, Di Xiao, and Yogesan Kanagasingam. “Exudate detection for diabetic retinopathy with convolutional neural networks”. In: *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, July 2017. DOI: `10.1109/embc.2017.8037180` (pages 14, 33, 42, 46).

[51] Anmol Popli et al. *Automated hard exudates segmentation in retinal images using patch based UNet*. `https://github.com/apopli/diabetic-retinopathy/blob/master/segmentation-hard-exudates.pdf`. July 2018 (pages 16, 42, 52, 66, 67).

[52] Arthur Samuel. *Computer Scientist (1901 – 1990).* Online; Accessed 20/08/2021 (page 19).

[53] Nils Nilsson. *The quest for artificial intelligence : a history of ideas and achievements.* Cambridge New York: Cambridge University Press, 2010. ISBN: 9780521122931 (page 19).

[54] Tom Mitchell. Online; Accessed 20/08/2021 (page 19).

[55] A.M. Turing. "I.—COMPUTING MACHINERY AND INTELLIGENCE". In: *Mind* LIX.236 (Oct. 1950), pp. 433–460. ISSN: 0026-4423. DOI: `10.1093/mind/LIX.236.433`. eprint: `https://academic.oup.com/mind/article-pdf/LIX/236/433/30123314/lix-236-433.pdf` (page 19).

[56] Yann Lecunn, Corinna Cortes, and Christopher J.C. Burges. *THE MNIST DATABASE of handwritten digits.* `http://yann.lecun.com/exdb/mnist/`. Online; Accessed 20/08/2021 (page 20).

[57] Stuart Russell. *Artificial intelligence : a modern approach.* Englewood Cliffs, N.J: Prentice Hall, 1995. ISBN: 0-13-103805-2 (page 20).

[58] Dr Michael J. Garbade. *Clearing the Confusion: AI vs Machine Learning vs Deep Learning Differences.* `https://towardsdatascience.com/clearing-the-confusion-ai-vs-machine-learning-vs-deep-learning-differences-fce69b21d5eb`. Online; Accessed 25 August 2020. Sept. 2018 (page 20).

[59] Sebastian Raschka. *Python machine learning : machine learning and deep learning with python, scikit-learn, and tensorflow 2.* Birmingham: Packt Publishing, Limited, 2019. ISBN: 1789955750 (page 20).

[60] Satavisa Pati. *The Difference Between Artificial Intellignence and Machine Learning.* `https://www.analyticsinsight.net/the-difference-between-artificial-intelligence-and-machine-learning/`. Online; Accessed 20/08/2021. Aug. 2021 (page 20).

[61] Wikipedia Contributors. *Weak Supervision.* https://en.wikipedia.org/wiki/Weak_supervision. Online; last edited 23-October-2021 Accessed 20/08/2021 (page 24).

[62] Wikipedia Contributors. *Intelligent agent.* https://en.wikipedia.org/wiki/Intelligent_agent. Online; last edited 23-October-2021, Accessed 22/10/2021 (page 24).

[63] Adam H. Marblestone, Greg Wayne, and Konrad P. Kording. "Toward an Integration of Deep Learning and Neuroscience". In: 10 (Sept. 2016). DOI: 10.3389/fncom.2016.00094 (page 25).

[64] Shigeki Sugiyama. *Human behavior and another kind in consciousness : emerging research and opportunities.* Hershey, Pennsylvania (701 E. Chocolate Avenue, Hershey, Pennsylvania, 17033, USA: IGI Global, 2019. ISBN: 978-1522582175 (page 26).

[65] Li Deng, Dong Yu, and Geoffrey E Hinton. *Deep Learning for Speech Recognition and Related Applications.* Workshop, in Twenty-third Conference on Neural Information Processing Systems. Dec. 2009. URL: https://nips.cc/Conferences/2009/ScheduleMultitrack?event=1512 (visited on 08/29/2021) (pages 26, 29).

[66] G. Cybenko. "Approximation by superpositions of a sigmoidal function". In: 2.4 (Dec. 1989), pp. 303–314. DOI: 10.1007/bf02551274 (page 27).

[67] Kurt Hornik. "Approximation capabilities of multilayer feedforward networks". In: 4.2 (1991), pp. 251–257. DOI: 10.1016/0893-6080(91)90009-t (page 27).

[68] Zhou Lu et al. "The Expressive Power of Neural Networks: A View from the Width". In: *31st Conference on Neural Information Processing Systems.* Long Beach, CA, USA, 2017 (page 27).

[69] Wikipedia Contributors. *John Hopfield.* https://en.wikipedia.org/wiki/John_Hopfield. Online; last edited 3-October-2021, Accessed 22/10/2021 (page 27).

[70] Wikipedia Contributors. *Bernard Widrow.* https://en.wikipedia.org/wiki/Bernard_Widrow. Online; last edited 14-October-2021, Accessed 22/10/2021 (page 27).

[71] Wikipedia Contributors. *Kumpati S. Narendra.* https://en.wikipedia.org/wiki/Kumpati_S._Narendra. Online; last edited 11-April-2021, Accessed 20/08/2021 (page 27).

[72] Warren S. McCulloch and Walter Pitts. "A logical calculus of the ideas immanent in nervous activity". In: 5.4 (Dec. 1943), pp. 115–133. DOI: 10.1007/bf02478259 (page 27).

[73] Henry J. Kelley. "Gradient Theory of Optimal Flight Paths". In: 30.10 (Oct. 1960), pp. 947–954. DOI: 10.2514/8.5282 (page 27).

[74] Stuart E. Dreyfus. "Artificial neural networks, back propagation, and the Kelley-Bryson gradient procedure". In: 13.5 (Sept. 1990), pp. 926–928. DOI: 10.2514/3.25422 (page 27).

[75] C. M. Berners-Lee. "Cybernetics and Forecasting". In: 219.5150 (July 1968), pp. 202–203. DOI: 10.1038/219202b0 (page 27).

[76] Kunihiko Fukushima. "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position". In: 36.4 (Apr. 1980), pp. 193–202. DOI: 10.1007/bf00344251 (page 28).

[77] Seppo Linnainmaa. "Taylor expansion of the accumulated rounding error". In: 16.2 (June 1976). Corpus ID: 122357351, pp. 146–160. DOI: 10.1007/bf01931367 (page 28).

[78] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. "Learning representations by back-propagating errors". In: 323.6088 (Oct. 1986), pp. 533–536. DOI: 10.1038/323533a0 (page 28).

[79] Yann Lecun et al. "Learning algorithms for classification: A comparison on handwritten digit recognition". English (US). In: *Neural networks.* Ed. by J.H. Oh, C. Kwon, and S. Cho. World Scientific, 1995, pp. 261–276. URL: https:

//nyuscholars.nyu.edu/en/publications/learning-algorithms-for-classification-a-comparison-on-handwritte (page 28).

[80] Special Report. *From not working to neural networking.* https://www.economist.com/special-report/2016/06/23/from-not-working-to-neural-networking. Online; Accessed 25 August 2020. June 2016 (page 28).

[81] S. Hochreiter et al. "Gradient flow in recurrent nets: the difficulty of learning long-term dependencies". In: *A Field Guide to Dynamical Recurrent Neural Networks.* Ed. by S. C. Kremer and J. F. Kolen. IEEE Press, 2001. URL: https://www.bibsonomy.org/bibtex/279df6721c014a00bfac62abd7d5a9968/schaul (page 29).

[82] *ImageNet.* Mar. 2021. URL: https://www.image-net.org/index.php (visited on 08/29/2021) (page 29).

[83] Geoff Hinton. *Recent Developments in Deep Neural Networks.* UBC Department of Computer Science's Distinguished Lecture Series. Youtube. May 2013. URL: https://www.youtube.com/watch?v=vShMxxqtDDs (page 29).

[84] Garofolo, John S. et al. *TIMIT Acoustic-Phonetic Continuous Speech Corpus.* 1993. DOI: 10.35111/17GK-BN40 (page 29).

[85] Frank Seide, Gang Li, and Dong Yu. "Conversational speech transcription using context-dependent deep neural networks". In: *Proc. Interspeech 2011.* 2011, pp. 437–440. DOI: 10.21437/Interspeech.2011-169 (page 29).

[86] *Large Scale Visual Recognition Challenge 2012.* Stanford Vision Lab. 2012. URL: https://image-net.org/challenges/LSVRC/2012/results.html (page 29).

[87] John Markoff. June 2012. URL: https://www.nytimes.com/2012/06/26/technology/in-a-big-network-of-computers-evidence-of-machine-learning.html?_r=1&hpw&pagewanted=all (page 30).

[88] *Merck Molecular Activity Challenge, Help develop safe and effective medicines by predicting molecular activity.* Featured Prediction Competition. Kaggle. 2012. URL: https://www.kaggle.com/c/MerckActivity (page 30).

[89] Stevo Bozinovski. "Reminder of the First Paper on Transfer Learning in Neural Networks, 1976". In: 44.3 (Sept. 2020). DOI: 10.31449/inf.v44i3.2828 (page 30).

[90] L. Y. Pratt. "Discriminability-Based Transfer between Neural Networks". In: *Advances in Neural Information Processing Systems*. Ed. by S. Hanson, J. Cowan, and C. Giles. Vol. 5. Morgan-Kaufmann, 1993. URL: https://proceedings.neurips.cc/paper/1992/file/67e103b0761e60683e83c559be18d40c-Paper.pdf (page 30).

[91] *Connection Science*. Vol. 08(02). Taylor & Francis Online, 1996. URL: https://www.tandfonline.com/toc/ccos20/8/2 (page 30).

[92] Nikos Tsiknakis et al. "Deep learning for diabetic retinopathy detection and classification based on fundus images: A review". In: *Computers in Biology and Medicine* 135 (2021), p. 104599. ISSN: 0010-4825. DOI: https://doi.org/10.1016/j.compbiomed.2021.104599 (page 31).

[93] D. H. Hubel and T. N. Wiesel. "Receptive fields and functional architecture of monkey striate cortex". In: 195.1 (Mar. 1968), pp. 215–243. DOI: 10.1113/jphysiol.1968.sp008455 (page 33).

[94] Santiago A. Cadena et al. "Deep convolutional models improve predictions of macaque V1 responses to natural images". In: 15.4 (Apr. 2019). Ed. by Wolfgang Einhäuser, e1006897. DOI: 10.1371/journal.pcbi.1006897 (page 34).

[95] Hamed Habibi Aghdam and Elnaz Jahani Heravi. *Guide to Convolutional Neural Networks. A Practical Application to Traffic-Sign Detection and Classification.* Cham, Switzerland: Springer International Publishing, 2017. ISBN: 978-3-319-86190-6. DOI: 10.1007/978-3-319-57550-6 (page 35).

[96] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. "U-Net: Convolutional Networks for Biomedical Image Segmentation". In: Springer International Publishing, 2015, pp. 234–241. DOI: 10.1007/978-3-319-24574-4_28 (page 38).

[97] Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. *Which Training Methods for GANs do actually Converge?* 2018. arXiv: `1801.04406 [cs.LG]` (page 42).

[98] Jason Brownee. *An overview of semantic image segmentationImpact of Dataset Size on Deep Learning Model Skill And Performance Estimates.* `https://machinelearningmastery.com/impact-of-dataset-size-on-deep-learning-model-skill-and-performance-estimates/`. Online; Accessed 20/08/2021. Jan. 2019 (page 43).

[99] Elaheh Imani and Hamid-Reza Pourreza. "A novel method for retinal exudate segmentation using signal separation algorithm". In: *Computer Methods and Programs in Biomedicine* 133 (Sept. 2016), pp. 195–205. ISSN: 0169-2607. DOI: `10.1016/j.cmpb.2016.05.016` (page 46).

[100] T. P. Udhaya Sankar, R. Vijai, and R. M. Balajee. "Detection and Classification of Diabetic Retinopathy in Fundus Images using Neural Network". In: *International Research Journal of Engineering and Technology (IRJET)* 5.4 (Apr. 2018), pp. 2630–2635. ISSN: 2395-0072 (page 46).

[101] Sohini Roychowdhury, Dara D. Koozekanani, and Keshab K. Parhi. "DREAM: Diabetic Retinopathy Analysis Using Machine Learning". In: *IEEE Journal of Biomedical and Health Informatics* 18.5 (Sept. 2014), pp. 1717–1728. DOI: `10.1109/jbhi.2013.2294635` (page 46).

[102] Pavle Prentašić and Sven Lončarić. "Detection of exudates in fundus photographs using deep neural networks and anatomical landmark detection fusion". In: *Computer Methods and Programs in Biomedicine* 137 (Dec. 2016), pp. 281–292. ISSN: 0169-2607. DOI: `10.1016/j.cmpb.2016.09.018` (page 46).

[103] Bo Wu et al. "Automatic detection of microaneurysms in retinal fundus images". In: *Computerized Medical Imaging and Graphics* 55 (Jan. 2017). Special Issue on Ophthalmic Medical Image Analysis, pp. 106–112. ISSN: 0895-6111. DOI: `10.1016/j.compmedimag.2016.08.001` (page 46).