Deep Attention Model for Diabetic Retinopathy Grading

DISSERTATION SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

Master of Technology
in
Computer Science

by

Sourav Aich
[Roll No: CS-1720]

under the guidance of

Dr. Sushmita Mitra
Head
Machine Intelligence Unit

Indian Statistical Institute
Kolkata-700108, India

July 2019

To my Guide and my Family

Certification

This is to certify that the dissertation entitled **"Deep Attention Model for Diabetic Retinopathy Grading"** submitted by **Sourav Aich (CS1720)** to Indian Statistical Institute, Kolkata, in partial fulfillment for the award of degree **Master Of Technology (M. Tech.) in Computer Science** is a bonafide record of work carried out by him under my supervision and guidance.
The dissertation has fulfilled all the requirements as per the regulations of this institute and, in my opinion, has reached the standard needed for submission.

Prof. Sushmita Mitra
Machine Intelligence Unit
Indian Statistical Institute
Kolkata 700 108, INDIA

Acknowledgments

I would like to express my gratitude to my advisor, **Prof. Sushmita Mitra**, Head, Machine Intelligence Unit, Indian Statistical Institute, Kolkata, for her guidance and continuous support and encouragement. She has literally taught me how to do good research, and motivated me with great insights and innovative ideas.

I would also like to thank **Subhashish Bannerjee**, Senior Research Fellow, Machine Intelligence Unit, Indian Statistical Institute, Kolkata, for his valuable suggestions and discussions.

Finally, I am very much thankful to my parents and my family for their everlasting support.

Last but not the least, I would like to thank all of my friends for their help and support.

Sourav Aich
Indian Statistical Institute
Kolkata - 700108 , India.

Abstract

Diabetic Retinopathy is the leading cause of blindness in today's modern world. Early detection of Diabetic Retinopathy is crucial for it's prevention. To speed up the process of detection, automated systems needs to be developed, which can grade a Fundus Image for DR, without any human intervention. In this project we have used an advanced variant of CNN(Convolutional Neural Network) integrated with Visual Attention Mechanism, for grading the Fundus Image for DR. We have also detected the lesions such as Microaneurysms, Haemmorhage, Hard Exudates and Soft Exudates in the Fundus images, and delineated their boundaries in the Fundus image. Finally we have developed a joint segmentation and classification pipeline, which mimics a pathologists action while grading a Fundus image. The system detects all the pathologies in the Fundus Images, marks them and with this pathological information grades the image.

Contents

1 Introduction **6**

2 Related Work **8**

3 Background **10**

3.1 Convolutional Neural Networks(CNN) 10

3.2 Semantic Segmentation 12

3.3 Diabetic Retinopathy 14

3.4 Problem Statement 15

3.5 Proposed Method 15

4 DR Severity grading through Attention based CNN **16**

4.1 Introduction 16

4.1.1 Attention Mechanism 16

4.1.2 Problem Statement 17

4.1.3 Proposed Approach 17

4.2 Attention Map Generation 17

4.3 Deep Attention Architecture 19

4.3.1 Ordinal Regression 19

4.4 Experimental Details 20

4.5 Results 21

5 Attention based DR Pathology Segmentation **23**

5.1 Introduction 23

5.1.1 Problem Statement 23

5.1.2 Challenges 23

5.1.3 Proposed Solution 24

5.2 Multi Scale Attention UNET with Deep Supervision 24
5.2.1 Attention Mechanism in Deep Segmentation Architectures . . 24
5.2.2 Attention Gates 25
5.2.3 Network Architecture 25
5.2.4 Experimental Details 26
5.3 Capsule Network based Segmentation Architecture 26
5.3.1 Introduction 26
5.3.2 Capsule Networks 26
5.3.3 Drawbacks of the original Dynamic Routing Algorithm 27
5.3.4 Vectorized Convolutional Operation 27
5.4 Network Architecture 28
5.4.1 Experimental Details 28
5.5 Results 29

6 DR classification through Deep Pathological Feature 31
6.1 Introduction 31
6.2 Deep Pathological Features 31
6.3 Network Architecture 32
6.4 Experimental Details 33
6.5 Results 33

7 Conclusion and Future Work 35

List of Figures

1.1 Fundus Image with pathological marking 7

3.1 Example of a network with many convolutional layers. Filters are applied to each training image at different resolutions, and the output of each convolved image is used as the input to the next layer. 11

3.2 U-Net: An encoder-decoder architecture. 13

4.1 Schematic showing Attention Mechanism 18

4.2 Deep Attention Architecture 19

4.3 Stages of DR 19

4.4 Attention Maps from multiple layers overlayed on a Fundus Image . . 21

4.5 $\tau = 0.4$ 22

4.6 $\tau = 0.5$ 22

4.7 $\tau = 0.6$ 22

5.1 Attention Gate 25

5.2 Multi Scale Attention UNET with deep supervision 25

5.3 Vectorized Convolution Operation 28

5.4 CapsUnet Architecture 29

5.5 Segmentation Result 30

6.1 Classification Network Architecture 32

6.2 $\tau = 0.4$ 33

6.3 $\tau = 0.5$ 33

6.4 $\tau = 0.6$ 33

List of Tables

4.1 Data Distribution among classes 20
4.2 Table showing the TPR(Sensitivity) , TNR(Specificity) and Accuracy for τ values 22

5.1 Dice Scores comparison of pathologies for different models 29

6.1 Table showing the TPR(Sensitivity) , TNR(Specificity) and Accuracy for τ values 34

Chapter 1

Introduction

Diabetic retinopathy, a chronic, progressive eye disease, has turned out to be one of the most common causes of vision impairment and blindness especially for working ages in the world today [5]. It results from prolonged diabetes. Blood vessels in the light-sensitive tissue (i.e. retina) are mainly affected in diabetic retinopathy. The non-proliferative diabetic retinopathy (NPDR) occurs when the blood vessels leak the blood in the retina. The Proliferative DR (PDR), which causes blindness in the patient, is the next stage to NPDR. The progress of DR can be categorized into four stages: mild, moderate, severe nonproliferative diabetic retinopathy, and the advanced stages of proliferative diabetic retinopathy. In mild NPDR, small areas in the blood vessels of the retina, called microaneurysms, swell like a balloon. In moderate NPDR, multiple microaneurysms, hemorrhages, and venous beading occur, causing the patients to lose their ability to transport blood to the retina. The third stage, called severe NPDR, results from the presence of new blood vessels, which is caused by the secretion of growth factor. The worst stage of DR is the proliferative diabetic retinopathy, in which fragile new blood vessels and scar tissue form on the surface of the retina, increasing the likelihood of blood leaking, leading to permanent vision loss. At present, retinopathy detection system is accomplished by involving a well-trained physician manually detecting vascular abnormalities and structural changes of retina in the retinal fundus images, which are then taken by dilating the retina using vasodilating agent. Due to the manual nature of DR screening methods, however, highly inconsistent results are found from different readers, so automated diabetic retinopathy diagnosis techniques are essential for solving these problems. Although DR can damage retina without showing any indication at the pre-liminary stage [14], successful early-stage detection of DR can minimize the risk of progression to more advanced stages of DR. The diagnosis is particularly difficult for early-stage detection because the process relies on discerning the presence of microaneurysms, retinal hemorrhages, among other features on the retinal fundus images. Furthermore, accurate detection and determination of the stages of DR can greatly improve the intervention, which ultimately reduces the risk of permanent vision loss. Earlier solutions of automated

diabetic retinopathy detection system were based on hand-crafted feature extraction and standard machine learning algorithm for prediction [18]. These approaches were greatly suffer due to the hand-crafted nature of DR features extraction since feature extraction in color fundus images are more challenging compared to the traditional images for object detection task. Moreover, these hand-crafted features are highly sensitive to the quality of the fundus images, focus angle, presence of artifacts, and noise. Thus, these limitations in traditional hand-crafted features make it important to develop an effective feature extraction algorithm to effectively analyze the subtle features related to the DR detection task. In recent times, most of the problems of computer vision have been solved with greater accuracy with the help modern deep learning algorithms, Convolutional Neural Networks (CNNs) being an example. CNNs have been proven to be revolutionary in different fields of computer vision such as object detection and tracking, image and medical disease classification and localization, pedestrian detection, action recognition, etc. The key attribute of the CNN is that it extracts features in task dependent and automated way.

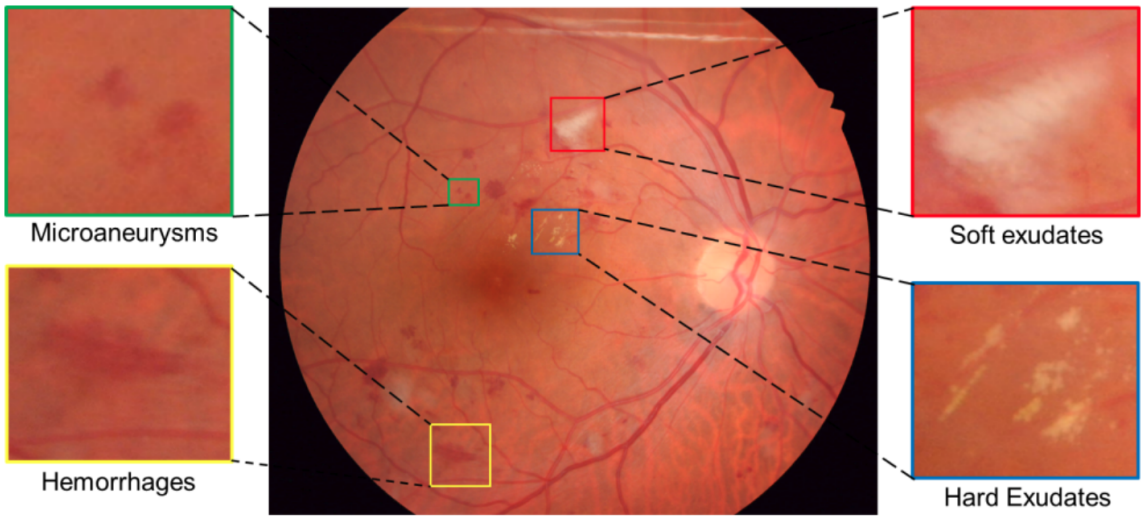

Figure 1.1: Fundus Image with pathological marking

Chapter 2

Related Work

The earlier works on automatic diabetic retinopathy detection were based on designing hand-crafted feature detectors to measure the blood vessels and optic disc, and on counting the presence of abnormalities such as microaneurysms, red lesions, haemorrhages, and hard exudates, etc. The detection was performed using these extracted features by employing various machine learning methods like support vector machines (SVM) and k-nearest neighbor (kNN) [18, 20]. In [2], Acharya et al. used features of blood vessel area, microaneurysms, exudes, and hemorrhages with an SVM, achieving an accuracy of 86%, specificity of 86%, and sensitivity of 82%. Roychowdhury et al. [16] developed a two-step hierarchical classification approach, where the non-lesions or false positives were rejected in the first step. For lesion classification in the second step, they used classifiers such as the Gaussian mixture model (GMM), kNN, and support vector machine (SVM). They achieved sensitivity of 100%, specificity of 53.16%, and AUC 0.904. However, these types of approaches have the disadvantage of utilizing limited number of features. Deep learning based algorithms have become popular in the last few years. For example, standard ImageNet architectures were used in [8, 22]. Furthermore, Kaggle [1] has recently launched a DR detection competition, where all the top ranked solutions were implemented employing CNN as the key algorithm. Pratta et al. [8] developed a CNN based model, which surpassed human experts in classifying advanced stages of DR. In [3], CNN based method was employed to detect microaneurysms a DR stage grading. Ensemble of CNN was employed to simultaneously detect DR and macular edema by Kori et al. [10]. They employed a variant of ResNet [6] and densely connected networks [7]. To make the model prediction more interpretable, a visual map was generated by Torre et al. [11] using CNN model, which can be used to detect lesion in the tested retinal fundus images. A similar approach was used in [23] along with generation of regression activation map (RAM). Some researches focused on exploring breakdown of classification task into subproblem prediction tasks. For example, Yang et al. [25] employed a two- stage deep convolutional neural network based methodology, where exudates, microaeurysms,and haemorrhage were first detected by local network and subsequent

severity grading was performed by global network. By introducing unbalanced weight map to emphasize leison detection, they achieved AUC of 0.9590. Authors of [4] implemented an architecture like VGG-16 [19] and Inception- 4 [21] network for DR classification.

Chapter 3

Background

3.1 Convolutional Neural Networks(CNN)

A convolutional neural network (CNN or ConvNet) is one of the most popular algorithms for deep learning, a type of machine learning in which a model learns to perform classification tasks directly from images, video, text, or sound. CNNs are particularly useful for finding patterns in images to recognize objects, faces, and scenes. They learn directly from image data, using patterns to classify images and eliminating the need for manual feature extraction. Applications that call for object recognition and computer vision — such as self-driving vehicles and face-recognition applications — rely heavily on CNNs.

Using CNNs for deep learning has become increasingly popular due to three important factors:

- CNNs eliminate the need for manual feature extraction—the features are learned directly by the CNN.
- CNNs produce state-of-the-art recognition results.
- CNNs can be retrained for new recognition tasks, enabling you to build on pre-existing networks.

A convolutional neural network can have tens or hundreds of layers that each learn to detect different features of an image. Filters are applied to each training image at different resolutions, and the output of each convolved image is used as the input to the next layer. The filters can start as very simple features, such as brightness and edges, and increase in complexity to features that uniquely define the object.

Like other neural networks, a CNN is composed of an input layer, an output layer, and many hidden layers in between.

These layers perform operations that alter the data with the intent of learning features specific to the data. Three of the most common layers are: convolution, activation or ReLU, and pooling.

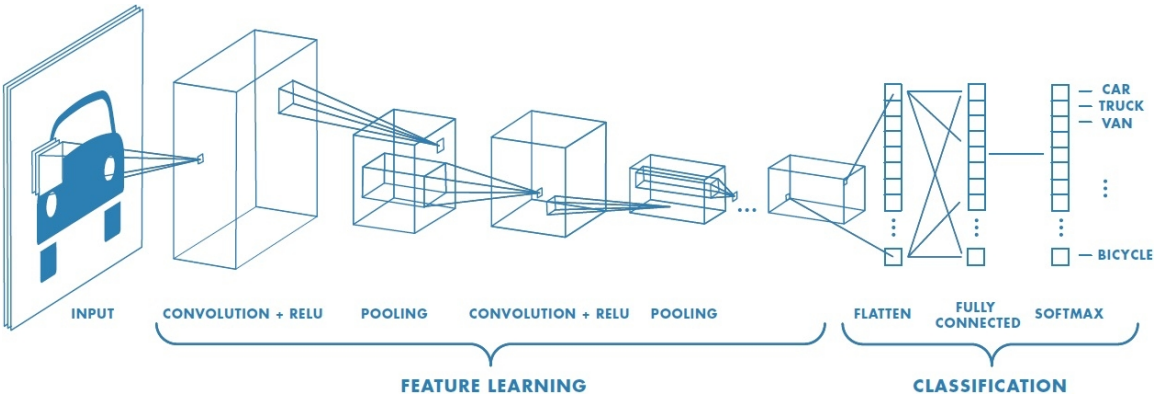

Figure 3.1: Example of a network with many convolutional layers. Filters are applied to each training image at different resolutions, and the output of each convolved image is used as the input to the next layer.

- Convolution puts the input images through a set of convolutional filters, each of which activates certain features from the images.

- Rectified linear unit (ReLU) allows for faster and more effective training by mapping negative values to zero and maintaining positive values. This is sometimes referred to as activation, because only the activated features are carried forward into the next layer.

- Pooling simplifies the output by performing nonlinear downsampling, reducing the number of parameters that the network needs to learn. These operations are repeated over tens or hundreds of layers, with each layer learning to identify different features.

After learning features in many layers, the architecture of a CNN shifts to classification.

The next-to-last layer is a fully connected layer that outputs a vector of K dimensions where K is the number of classes that the network will be able to predict. This vector contains the probabilities for each class of any image being classified.

The final layer of the CNN architecture uses a classification layer such as softmax to provide the classification output.

3.2 Semantic Segmentation

Semantic segmentation is understanding an image at pixel level i.e, we want to assign each pixel in the image an object class. It is one of the key problems in the field of computer vision. Looking at the big picture, semantic segmentation is one of the high-level task that paves the way towards complete scene understanding. The importance of scene understanding as a core computer vision problem is highlighted by the fact that an increasing number of applications nourish from inferring knowledge from imagery. Some of those applications include self-driving vehicles, human-computer interaction, virtual reality etc. With the popularity of deep learning in recent years, many semantic segmentation problems are being tackled using deep architectures, most often Convolutional Neural Nets, which surpass other approaches by a large margin in terms of accuracy and efficiency.

Semantic segmentation is a natural step in the progression from coarse to fine inference:

- The origin could be located at classification, which consists of making a prediction for a whole input.
- The next step is localization / detection, which provide not only the classes but also additional information regarding the spatial location of those classes.
- Finally, semantic segmentation achieves fine-grained inference by making dense predictions inferring labels for every pixel, so that each pixel is labeled with the class of its enclosing object ore region.

A general semantic segmentation architecture can be broadly thought of as an encoder network followed by a decoder network:

- The encoder is usually is a pre-trained classification network like VGG/ResNet followed by a decoder network.
- The task of the decoder is to semantically project the discriminative features (lower resolution) learnt by the encoder onto the pixel space (higher resolution) to get a dense classification.

  Unlike classification where the end result of the very deep network is the only important thing, semantic segmentation not only requires discrimination at pixel level but also a mechanism to project the discriminative features learnt at different stages of the encoder onto the pixel space. Different approaches employ different mechanisms as a part of the decoding mechanism

In 2014, Fully Convolutional Networks (FCN) by Long et al.[13], popularized CNN architectures for dense predictions without any fully connected layers. This allowed

segmentation maps to be generated for image of any size and was also much faster compared to the patch classification approach. Almost all the subsequent state of the art approaches on semantic segmentation adopted this paradigm.

Apart from fully connected layers, one of the main problems with using CNNs for segmentation is pooling layers. Pooling layers increase the field of view and are able to aggregate the context while discarding the 'where' information. However, semantic segmentation requires the exact alignment of class maps and thus, needs the 'where' information to be preserved. Two different classes of architectures evolved in the literature to tackle this issue. First one is encoder-decoder architecture. Encoder

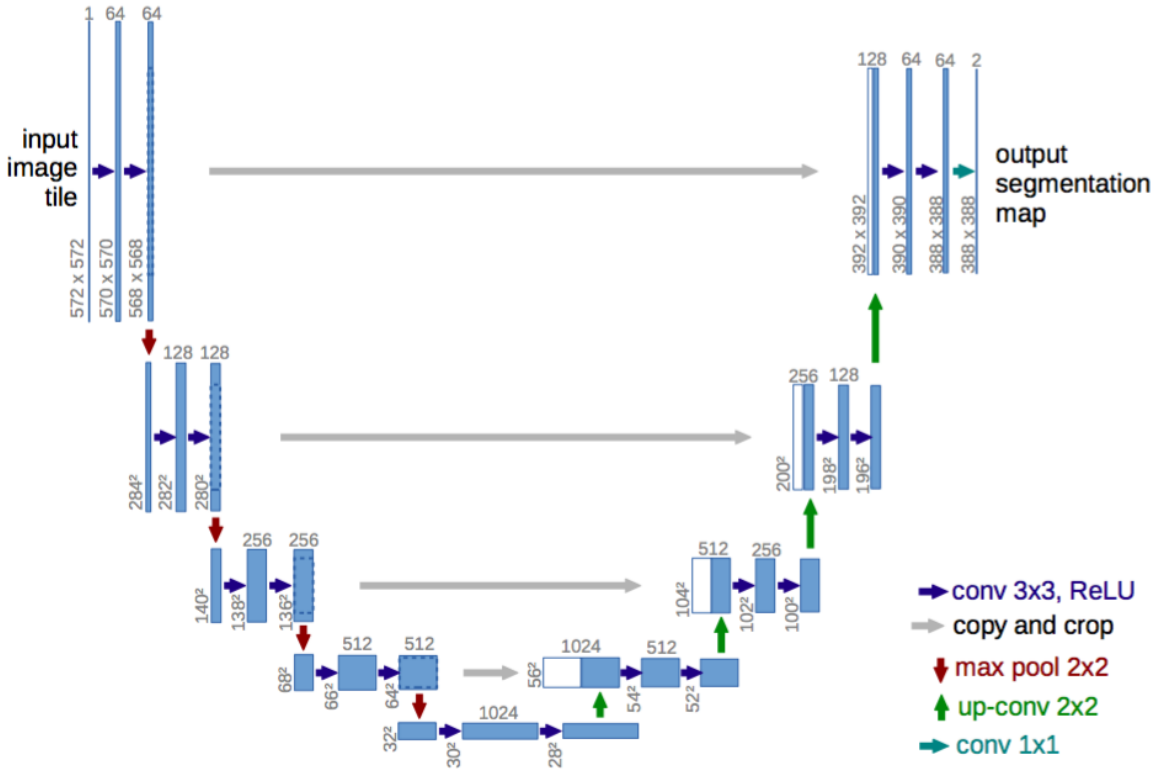

Figure 3.2: U-Net: An encoder-decoder architecture.

gradually reduces the spatial dimension with pooling layers and decoder gradually recovers the object details and spatial dimension. There are usually shortcut connections from encoder to decoder to help decoder recover the object details better. U-Net by Ronneberger et al. [15] is a popular architecture from this class.

3.3 Diabetic Retinopathy

Diabetic retinopathy is a complication of diabetes and the leading cause of vision impairment and blindness among working-age adults. It occurs when diabetes damages the tiny blood vessels in the retina, which is the light-sensitive tissue at the back of the eye. Diabetic retinopathy may lead to diabetic macular edema (DME), which is a swelling in an area of the retina called the macula.Chronically high blood sugar from diabetes is associated with damage to the tiny blood vessels in the retina, leading to diabetic retinopathy. The retina detects light and converts it to signals sent through the optic nerve to the brain. Diabetic retinopathy can cause blood vessels in the retina to leak fluid. or hemorrhage (bleed), distorting vision. In its most advanced stage, new abnormal blood vessels proliferate (increase in number) on the surface of the retina, which can lead to scarring and cell loss in the retina. Diabetic retinopathy may progress through four stages:

- Mild nonproliferative retinopathy. Small areas of balloonlike swelling in the retina's tiny blood vessels, called microaneurysms, occur at this earliest stage of the disease. These microaneurysms may leak fluid into the retina.
- Moderate nonproliferative retinopathy. As the disease progresses, blood vessels that nourish the retina may swell and distort. They may also lose their ability to transport blood. Both conditions cause characteristic changes to the appearance of the retina and may contribute to DME.
- Severe nonproliferative retinopathy. Many more blood vessels are blocked, depriving blood supply to areas of the retina. These areas secrete growth factors that signal the retina to grow new blood vessels.
- Proliferative diabetic retinopathy (PDR). At this advanced stage, growth factors secreted by the retina trigger the proliferation of new blood vessels, which grow along the inside surface of the retina and into the vitreous gel, the fluid that fills the eye. The new blood vessels are fragile, which makes them more likely to leak and bleed. Accompanying scar tissue can contract and cause retinal detachment—the pulling away of the retina from underlying tissue, like wallpaper peeling away from a wall. Retinal detachment can lead to permanent vision loss.

3.4 Problem Statement

The main goal of this project is to classify a Fundus image according to it's Diabetic Retinopathic severity. To achieve this objective many researchers have used both Machine Learning and Deep Learning based techniques1. Our aim here is to develop a Deep Learning based fully automated system which classifies the given input Fundus image along with it also genrate a map indicating the presence of pathologies or lesions which are caused due to Diabetic Retinopathy.

3.5 Proposed Method

For grading the fundus image we have adopted two techniques:

- We have used an advanced ConvNet architecture incorporating *Visual Attention Mechanism* to classify the Fundus Images for Diabetic Retinopathic Severity.
- We have proposed a new model, where we Segment the Fundus images for Diabetic Retinopathic pathologies, and used the *Deep Pathological Features* learned in the process of *segmentation* to further improve the accuracy of the classification task.

Chapter 4

DR Severity grading through Attention based CNN

4.1 Introduction

4.1.1 Attention Mechanism

Attention mechanisms play an important role in modern Deep Learning Based architectures, especially in computer vision tasks. Many visual attention models have been introduced recently, and they have shown that attaching an attention mechanism to the existing model improves the accuracy in various tasks such as image classification, image caption generation, image generation, and visual question answering.

There are several motivations for incorporating attentive mechanisms in Deep CNN. One of them is an analogy to the perceptual process of a human being. The human visual system pays attention to a region of interest instead of processing an entire scene. Similarly, in a Deep CNN based Attention model, we can focus only on attended areas of the input image. This is beneficial in terms of computational cost; the number of hidden units may be reduced since the hidden activations only need to encode the region with attention. Also, focusing only on the relevant regions of the image, makes the model less succeptible to noise and improves the models sensititvity and specificity.

Our aim is to generate *Attention maps* to detect the visual information used by CNNs for their classification outcome.

This approach is based on the hypothesis that there is an advantage in identifying salient regions in the image and boosting their influence, while attenuating the irrelevant and confusing information in other regions of the image. We have developed a trainable attention map generator and integrated it into standard CNN architecture build for the task of DR severity grading.

4.1.2 Problem Statement

Given colour Fundus Images our task is to classify the images according to it's Diabetic Retinopathic severity.

4.1.3 Proposed Approach

To grade DR from fundus images we have developed an *Attention* based deep CNN model. Employing attention mechanisms in the CNN architecture allows the network to focus on relevant regions on the input fundus image(regions affected with the disease) which in turn improves the performance of the ConvNet while reducing the computational burden.

We have also reformulated the classification problem as a regression problem, in which our model ouptuts a DR severity score in the range {0...4}. This has allowed us to better evaluate the model's performance.

4.2 Attention Map Generation

An attention map is a scalar matrix which represents the relative importance of layer activations at different 2D spatial locations with respect to the target task. This notion of a nonuniform spatial distribution of relevant features being used to form a task-specific representation, and the explicit scalar representation of their relative relevance, is known as *Attention*. Previous works have shown that for a CNN trained using image-level annotations alone, extracting the attention map provides a straightforward way of determining the location of the object of interest. In this method *Attention* is represented as a probabilistic map over the input image, and these maps are learned via an end-to-end deep learning framework.

Fig. 4.1 shows the schematic of the *attention* architecture. Here the term *'local features'* refers to features extracted by some intermediate layer of the CNN whose receptive fields are, respectively, contiguous proper subsets of the image ('local') and the term *'gloabal features'* refers to the entire image ('global'). By computing a *compatibility measure* between local and global features, we force the CNN architecture to classify the input image using only a weighted combination of local features, with the weights represented here by the attention map. The network therefore learns an attention map relevant to solving the given classification task.

Let L^s be the set of feature vectors extracted at a layer s. $L^s = \{l_1^s, l_2^s, \ldots, l_n^s\}$, be the n feature vectors at a scale s. Here each ${l_i}^s$ is a vector of output activations at the spatial location i of n total spatial locations in the layer. The global feature vector $\mathbf{g}$ has all the global information of the image, having only to pass through the final fully connected layers to produce the class score for that input Now we want to

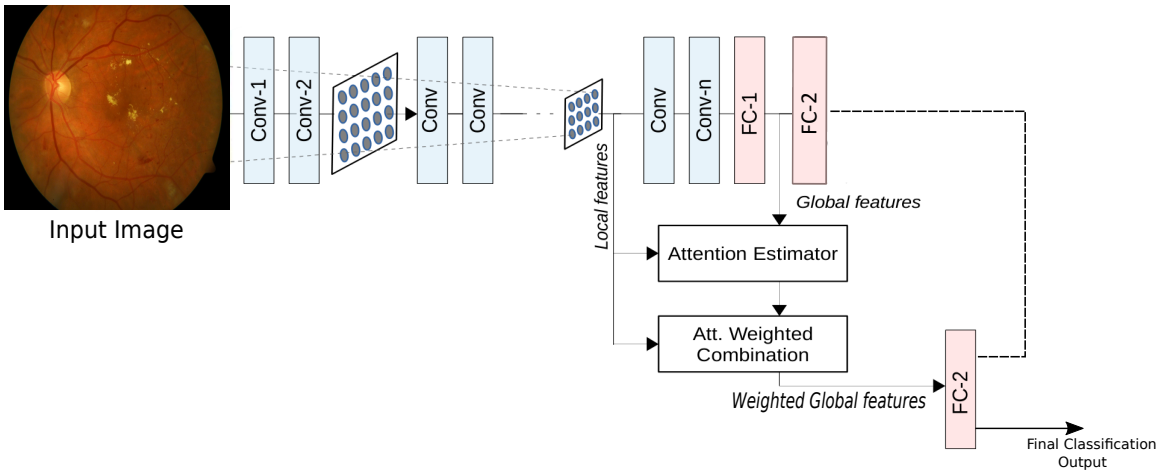

Figure 4.1: Schematic showing Attention Mechanism

generate attention maps for the layer s. To do that we compute a compatibility score of the n feature vectors with the global feature vector $\mathbf{g}$. The feature vectors are first transformed via linear mapping of l_i^s to the dimensionality of $\mathbf{g}$. Then we compute the compatibility of each of n feature vectors with $\mathbf{g}$ by computing the dot product between them. Finally the scores are normalized to lie between (0,1) by passing the scores through a sigmoid activation function. And hence we get the attention map at a scale s. After the attention map is obtained, we compute the weighted sum of the feature vectors at the scale s, yielding a vector of length equal to the C^s where C^s is the number of channels at scale s. We get the attended feature vector G_a^s at scale s as:

$$G_a^s = \sum_{i=1}^{n} \alpha_i l_i^s \tag{4.1}$$

where α_i's are the attentional scores.

We extract these attended feature vector at multiple scale, and aggregate them and pass it through the final classifaction layer. This mechanisms constrains the network to learn local features relevant to the global information which is fed through the final classification layer. This approach ensures that the network learns relevant attributes of the classes at each scale, and hence the learning process is more stable.

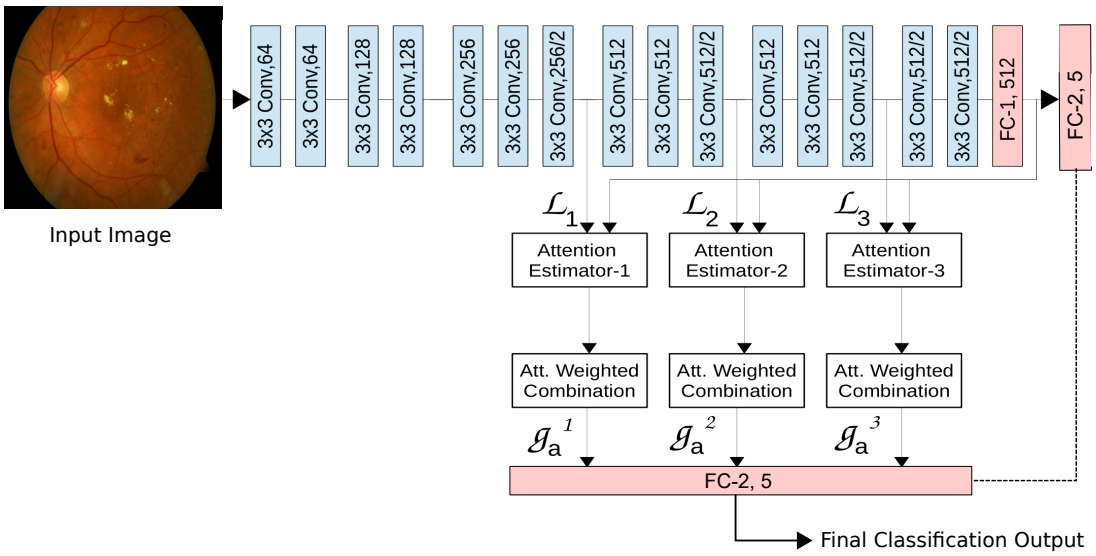

Figure 4.2: Deep Attention Architecture

4.3 Deep Attention Architecture

Our Attention based ConvNet consists of 17 layers, of which first 15 are convolutional blocks and the last 2 are fully connected layers. Attention mechanism is implemented by taking the output activations of convolutional layers 7, 10, and 13 as *local features* $\mathbf{L_1}$, $\mathbf{L_2}$, $\mathbf{L_3}$. The output activations of layer16 (fc) define our global feature vector $\mathbf{g}$. Our network has 5 logistic output nodes, each node corresponds to one of the DR classes.

4.3.1 Ordinal Regression

Figure 4.3: Stages of DR

While training the network with a standard cross entropy loss function, we have noticed that the network is often *confused* to classify the image to it's correct class among it's *neighboring classes*. In order to avoid such confusion while training or

testing our model, we have reformulated the DR severity classification problem as an *ordinal class regression problem*. DR severity classes have a natural ordering among them. The classes can be ordered based on the magnitude of the pathologies present in the fundus images. In figure 4.3, we can see that all the five fundus images look almost similar, the only difference being the amount of the pathologies present in them. So, it's beneficial for us if the model could ouput a score (a value in the range of $\{0, 4\}$, rather than trying to output discrete values in $[0, 4]$. To make the network output a countinous value in the range $\{0, 4\}$, we used the following mechanism. Since our network outputs five logistic unit whose value lie in the range $(0, 1)$. We attach a fully connected layer with only one output node, on top of the five logistic output of our network. We make the weights of this layer learnable. And we optimize the *Mean Squared Error* loss function to train the network.

The loss function is as follows:

$$L = \sum_{i=1}^{n} (\hat{Y}_i - Y_i)^2 \tag{4.2}$$

where $\hat{Y} \in [0, 4]$ and $Y \in \{0, 4\}$

Now as our model outputs a score **s** in the range $\{0, 4\}$, we classify each of the input images of the network, based on a threshold τ . If the network's ouptut score s, lies in the range $[k, k + 1\}$ where $k \in [0, 3]$, we classify the image as follows:

$$\begin{aligned} if \;\; s - k <= \tau \;\; then \;\; i \in class \;\; k \\ else \;\; i \in class \;\; k + 1 \end{aligned}$$

4.4 Experimental Details

Stage	***Severity***	***No. of Images***	***Percentage***
0	Normal	25810	73.48%
1	Mild	2443	6.96%
2	Moderate	5292	15.07%
3	Severe	873	2.48%
4	Proliferative	708	2.01%

Table 4.1: Data Distribution among classes

For training the above network we have used the publicly available EyePacs dataset. The dataset consists of 35,126 color fundus images. Table 4.1 shows the distribution of classes among the dataset. The dataset is splitted into training, valdidation and test set, in the ratio of 70:30:10. The validation set and the test set are made balanced. The network is trained on a single Nvidia P100 GPU for 50 epochs. The training mechanism adopted by us is as follows:

- We have used the minibatch gradient descent algorithm with a batch size of 30. As, the dataset has imbalanced classes(the number of training samples are different among classes), we used a balanced mini-batch gradient descent algorithm to train our network.
- Initially we train the network without the ordinal regression layer, by optimizing the categorical cross-entropy loss function.
- Finally we add the ordinal regression layer, and optimize the Mean Squared Error loss.

4.5 Results

Figure 4.4 shows the Attention maps generated by the model at multiple scale. The maps are first interpolated via bilinear intrepolation to match the dimension of the input image, and then they are overlayed over the input image. From the figure we can see that these attention maps are most active at regions where there is a higher density of pathologies.

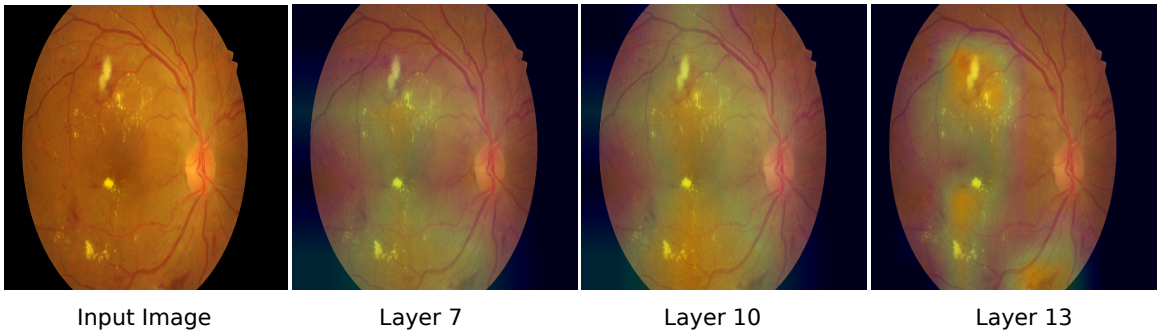

Figure 4.4: Attention Maps from multiple layers overlayed on a Fundus Image

The confusion matrices for multiple values of threshold τ are shown below. The test set consists of 350 Fundus images with 70 images belonging to each class. We have evaluated our model by computing the Quadratic Weighted Kappa metric, class wise sensitivity or the true positive rate, class wise specificity or the true negative rate and balanced overall accuracy, The evaluated metrics are shown in table 4.2. The *Quadratic Weighted Kappa* Obtained is **0.85** for our best model.

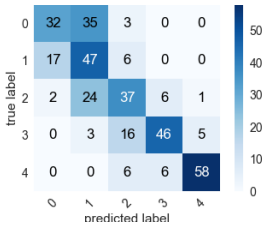

Figure 4.5: $\tau = 0.4$

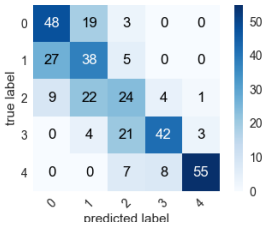

Figure 4.6: $\tau = 0.5$

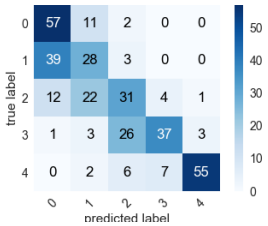

Figure 4.7: $\tau = 0.6$

τ	Class_0		Class_1		Class_2		Class_3		Class_4		Accuracy
	TPR	TNR	TPR	TNR	TPR	TNR	TPR	TNR	TPR	TNR	
0.4	0.45	0.86	**0.67**	0.83	0.52	0.87	0.65	0.95	**0.82**	**0.98**	**0.62**
0.5	0.68	**0.93**	0.52	0.77	0.4	**0.89**	0.6	0.95	0.78	0.97	0.60
0.6	**0.81**	0.81	0.4	**0.86**	0.44	0.86	0.52	**0.96**	0.78	**0.98**	0.59

Table 4.2: Table showing the TPR(Sensitivity) , TNR(Specificity) and Accuracy for τ values

Chapter 5

Attention based DR Pathology Segmentation

5.1 Introduction

When pathologists grade a Fundus image for Diabetic Retinopathy, they look for the presence of typical pathologies associated with DR in the Fundus Image. The presence and intensity of the pathologies is indicative of the severity of Diabetic Retinopathy for the subject. Here we have build a complete DR diagnostic pipeline, where we not only grade the Fundus Image for DR severity but also locate the pathologies responsible for it, in the Fundus image. In order to build such a system, we have first *Segmented* the Fundus Image for DR specific pathologies, then based on these pathological evidence extracted from the image we further grade the image.

5.1.1 Problem Statement

Given a color Fundus Image our task is to *Segment* the regions corresponding to *Microaneurysms*, *Haemorrhages*, *Soft Exudates*, *Hard Exudates* and *Optic Disc*. Formally, given an Image I of size $M \times N \times 3$ the problem of segmentation can be formulated as a pixel-wise classification task where each pixel is assigned a label $L \in \{l_1, l_2, ..., l_C\}$ such that an output image O of size $M \times N \times C$ is generated. Each channel of O corresponds to one of the classes.

5.1.2 Challenges

Segmenting a Fundus Image for DR specific pathologies is a challenging task for the following reasons :

- Pathologies such as *Microaneurysm* and *Haemorrhages* are usually scarce and

their dimensions are negligible compared to the image dimension. Standard segmentation architectures like UNET and FCNN does not yield satisfactory results, mainly because these architectures progressively uses *max pooling* operation in their encoding path to achieve *translational invariance* over small spatial shifts in the input image meanwhile loosing important spatial information (boundary details) of these tiny pathologies. Therefore it necessitate the construction of Segmentation Architectures which can preserve the spatial information of these pathologies at every stage of the network.

5.1.3 Proposed Solution

To address the problems associated with standard UNET architecture in segmenting DR pathologies, we have integrated attention mechanism into a standard UNET[15] architecture, by implementing *attention gates*. Additionally we have also used a *Capsule Network* based segmentation architecture to further improve the segmentation performance.

5.2 Multi Scale Attention UNET with Deep Supervision

5.2.1 Attention Mechanism in Deep Segmentation Architectures

In a standard CNN architecture the feature maps are gradually downsampled to capture a sufficiently large receptive field in the input image.In this way, features on the coarse spatial grid level model location. However, it is very difficult to reduce false-positive predictions for small objects such as *Microaneurysms* and *Haemmorhages* that show large shape variability. To alleviate this issue, we have integrate *Attention Gates (AG)* into the UNET architecture. Attention Gates act as a filter to the responses from the encoding stage of the UNET that are being concatenated to the decoding stage via skip connections. These gates are scalar matrices whose values lies in the range $[0, 1]$ and represent probabilities of salient objects relevant for the Segmentation task, and when these maps are multiplied by the input response before concatenating it with the decoding stage, helps to atteneutae feature responses that are not relevant for the task. These maps are generated during the test time, and thus these maps can be learned end-to-end via a deep learning framework.

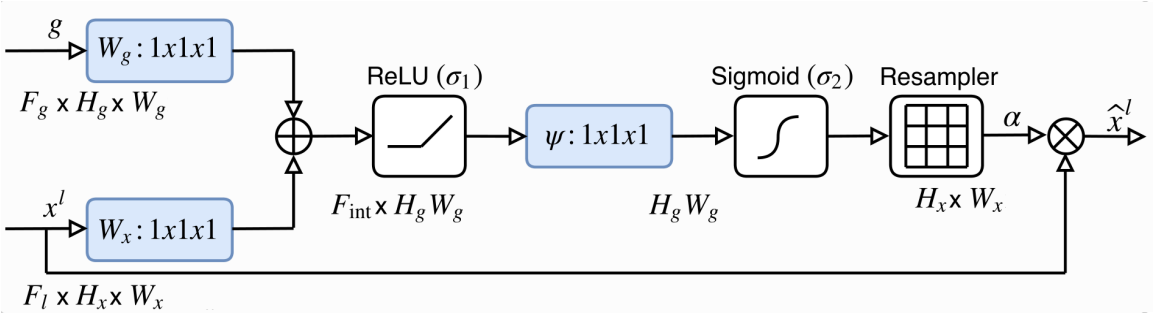

Figure 5.1: Attention Gate

5.2.2 Attention Gates

As explained in the previous section that Attention Gates helps to preserve feature responses relevant only to the task, these gates are implemented via the mechanism as described in Fig 5.1

Input features x^l are scaled with attention coefficients α modelled via Attention gates(AG). Relevant spatial regions are passed by analysing the activations of the gating signal g which contains rich spatial information and the contextual information provided by the input signal x^l

5.2.3 Network Architecture

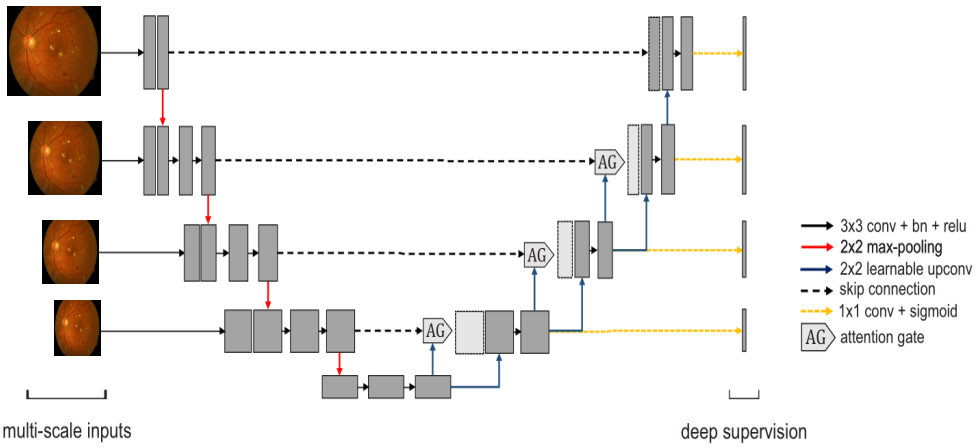

Figure 5.2: Multi Scale Attention UNET with deep supervision

Fig 5.2 shows the schematic of the *Multi Scale Attention UNET* architecture. The architecture is similar to the standard UNET structure, with *Attention Gates* integrated into it. The network is trained via *Deep Supervision technique* [5], which

forces intermediate layers to be semantically discriminative at every scale. After each max-pooling stage, the input image is downscaled and concatenated with the feature maps. We have found that this technique has been very advantegeous in improving the segmentation accuracy of small features like *Microaneurysms* and *Haemorrhage*.

5.2.4 Experimental Details

The dataset used for training the Segmentation network is from the IDRiD Diabetic Retinopathy Segmentation and Grading Challenge. The dataset consists of 81 color Fundus Images, with segmented ground truth mask for four pathologies such as *Microaneurysms*, *Haemorrhage*, *Hard Exudates* and *Soft Exudates*, and a segmented mask for *Optic Disc*. The color fundus images are of resolution 4288x2848. The images are of a very high resolution, to be trained on a single GPU. Also the number of images in the dataset is too less to train the network. So, we extract around 250 patches of resolution 256x256 from each image, and use those patches to train the Segmentation Network. We have used the Binary Cross Entropy loss as the objective function and is optimized by mini-batch gradient descent with batch size of 8, using an Adam Optimizer with learning rate 0.001 for 100 epochs.

5.3 Capsule Network based Segmentation Architecture

5.3.1 Introduction

Capsule Network is a recently published architecture by Hinton et al. [17]. The major advantage of capsule network lies in their ability to preserve more information about input by replacing max-pooling layers with dynamic routing algorithm. We have used Capsule Networks for the task of Segmenting DR specific pathologies from Fundus images. Our architecture follows the conventional encoder-decoder architecture like the UNET. The encoder is composed of multiple sequential capsule layers, whose responses are routed with a novel *Vectorized Convolution Operation*. Here the max-pooling operation in the UNET architecture is replaced by *locally-constrained dynamic routing*[12]. The decoder is composed of multiple transposed Convolutional layers, each followed by *Vectorized Convolution Operation*.

5.3.2 Capsule Networks

A *Capsule* is a group of neurons whose output represent different properties of a single entity. These capsules encapsulates a large number of pose information(e.g. position,

orientation, scaling and skewness) together with other instantiation parameters such as color and texture for different parts and fragments of that entity.

5.3.3 Drawbacks of the original Dynamic Routing Algorithm

The Dynamic Routing Algorithm, proposed by Sabour et al. [17] is used to encode the part to whole relationships between features, and this is sufficient to extract meaningful features for simple dataset like MNIST. But for complex dataset, using only this routing algorithm is insufficient. This algorithm alone fails to extract complex features, and thus Capsule network performance on complex dataset has been unsatisfactory. So we have to came up with a novel operation on Capsule layers known as the **Vectorized Convolution Operation**, analogous to the convolutional operation in a deep CNN. Recectly Bing et al. [24] shows that merely increasing the number of Capsule layers in the network and using Dynamic Routing procedure between them is not sufficient for producing satisafctory result on complex datasets. Moreover the Dynamic routing algorithm is computationally expensive if used in it's original form, so we used a modified version of this algorithm known as **Locally Constrained Dynamic Routing**[12].

5.3.4 Vectorized Convolutional Operation

Intuition:

In order to extract semantically complex features from a set of simple features, we define a new operation that can operate on the capsule layers representing simple features and used it to extract more complex features from them. As we know each capsule outputs a vector representing the pose parameters of an entity, consider two entities A and B, represented by their pose vectors $\vec{v_a}$ and $\vec{v_b}$. We need a parameterized functional operation on $\vec{v_a}$ and $\vec{v_b}$, to be able to extract a complex entity C represented by the pose vector $\vec{v_c}$. Formally, $\vec{v_c} = f(\vec{v_a}, \vec{v_b}; \theta)$. The nature of f must be nonlinear, in order to extract features which are semantically more complex than features extracted by the previous layers. Here θ is the learnable parameter.

The operation:

Consider a input capsule volume V^l of dimension (h^l, w^l, d^l, n^l). Where h^l and w^l are the spatial resolution of the input capsule volume, d^l is the number of dimensions of each capsule and n^l is the number of capsule types. Consider a square vector kernel $\vec{K_t}$ of dimension (f, f, d^l, n^l), where f is the filter's resolution, and $\vec{K_t}$ is the kernel of type t producing a capsule of type t for the next layer. see fig. 5.3

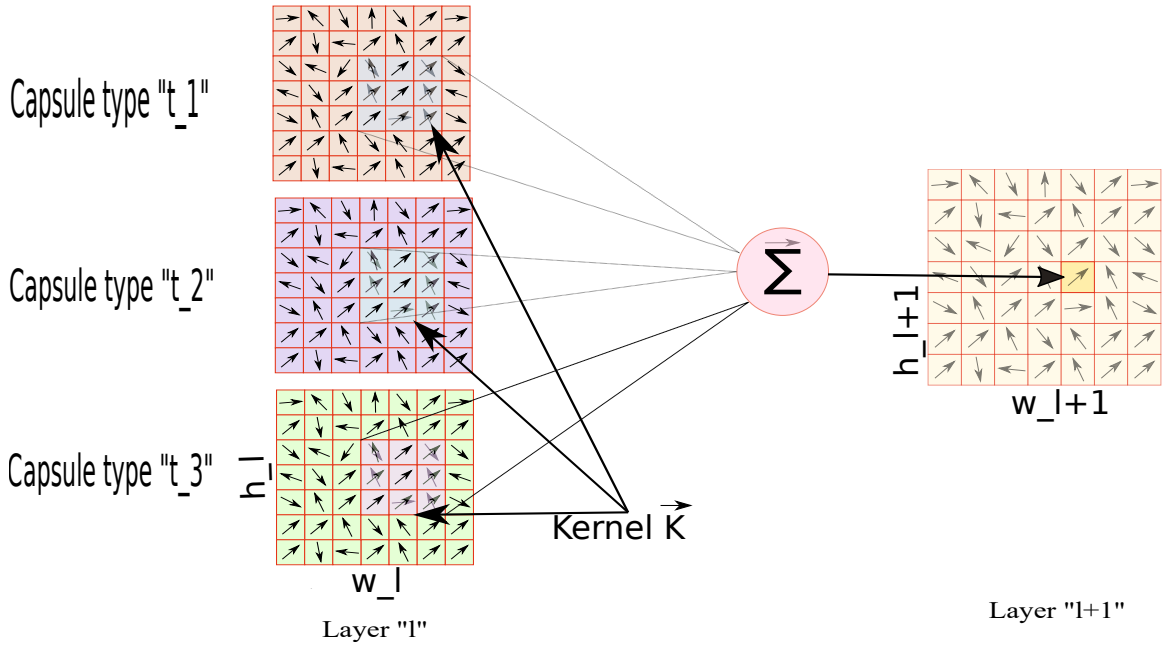

Figure 5.3: Vectorized Convolution Operation

Let the capsule volume $\mathbf{V}$ with element $V_{i,j,k,l}$ is convolved with the kernel $\vec{K}_t$ to produce an output volume $\mathbf{Z}$. Then

$$(Z_d^{l+1})_t = \sum_{i,j,t_1,t_2} V_{i-t_1,j-t_2,d,t_3} * K_{t_1,t_2,d,t_3}$$

5.4 Network Architecture

Fig. 5.4 shows the architecture of CapsUnet. The input to the network is a 256x256 patch extracted from the fundus image. The first 2 layers are the Convolution layer. Then the output feature from the initial Convolutional layers are then reshaped to get the first primary capsule layer. Then these capsule layers is acted upon by subsequent Vectorized Convolutional Operation and the capsules are routed to the next layer capsules through Locally Constrained Dynamic Routing. Finally at the decoding stage we upsample the capsule layers using transposed convolution followed by Vectorized Convolutional Operation and finally to generate the segmentation map we used a 1x1 Convolution filter as a output layer.

5.4.1 Experimental Details

We have implemented the entire architecture with the PyTorch Deep Learning Framework. We trained the entire network with SGD optimizer with a learing rate of 0.01. The number of routing iterations is kept constant at 2 for all the layers. We used

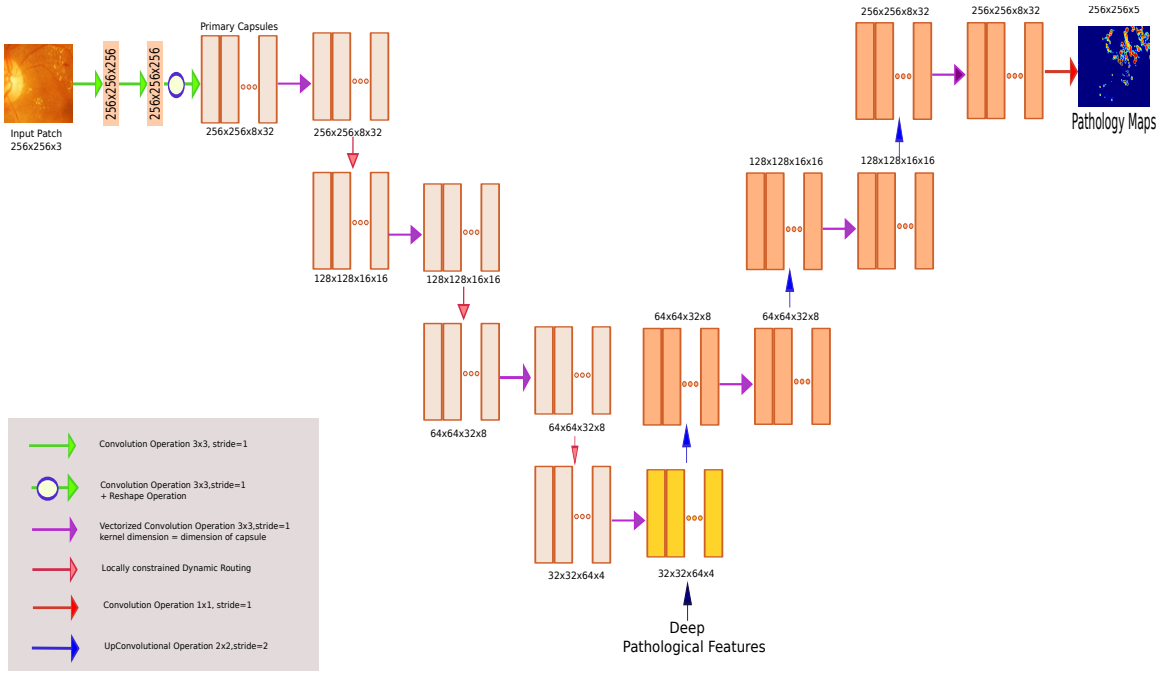

Figure 5.4: CapsUnet Architecture

the IDRiD dataset for training and validation purpose. The dataset consists of 81 colour fundus image with segmented map for each of the pathologies and optic disc. Each image are of dimension (2848x4288). We have extracted patches of dimension (256x256) and then used these patches as to train our model. The network is trained on NVidia Tesla P100 GPU.

5.5 Results

In figure 5.5 we show two samples of our Segmentation output along with the ground truth and the original image.

The performance comparison of our model with other standard architecture is shown in table 5.1. The dice score is evaluated for each of the pathologies.

Model	**Exudates**	**Haemmorhage**	**MicroAneurysms**	**SoftExudates**
UNET(baseline)	0.463	0.214	0.184	0.04
AttUNET	0.924	0.364	0.535	0.31
AttUNET with DS	0.945	0.754	0.716	0.47
CapsUNET	**0.975**	**0.821**	**0.732**	**0.52**

Table 5.1: Dice Scores comparison of pathologies for different models

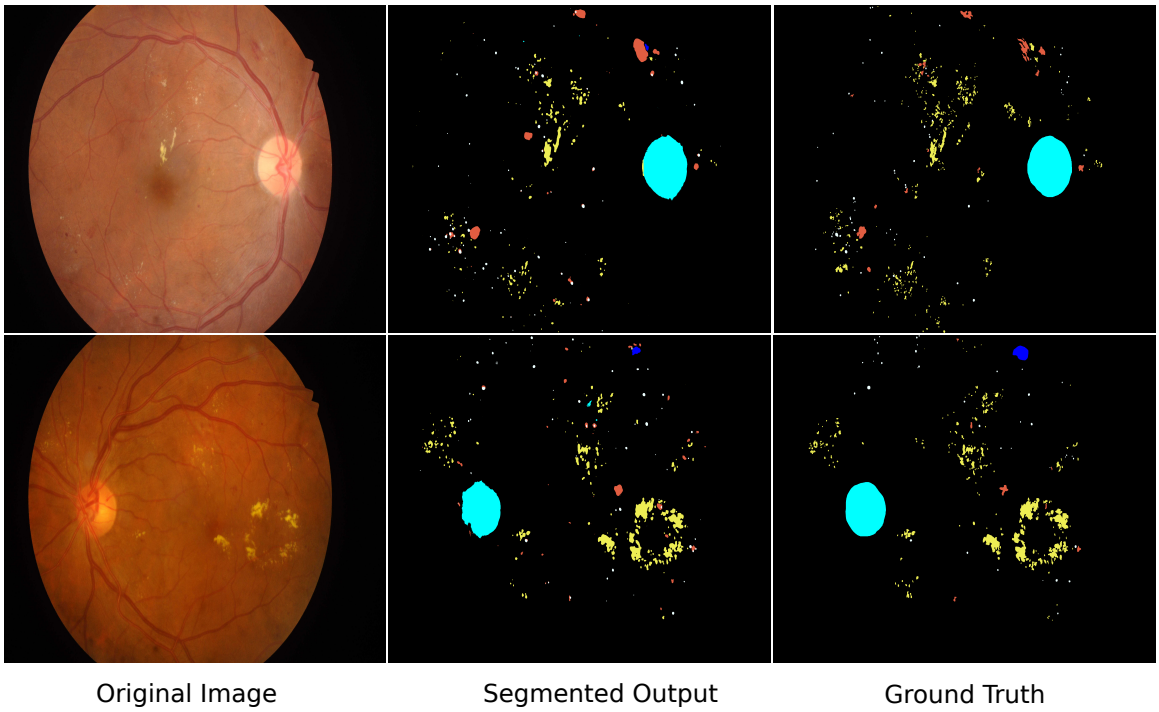

Figure 5.5: Segmentation Result

Chapter 6

DR classification through Deep Pathological Feature

6.1 Introduction

Since our main goal of this project was to not only grade the Fundus images for DR, but also to know the reason for it, for which we have already developed an automated pathology segmentation method (see chapter: 5). Here, we have developed a DR grading algorithm which uses the knowledge(pathology specific features) extracted from the fundus image, to improve the classification accuracy. This system mimics the approach adopted by a patholgists to grade a fundus image.

6.2 Deep Pathological Features

While trying to segment the fundus images for DR pathologies, we have adopted an advanced version of UNET with attention as shown in **??**, and also developed a Capsule based Segmentation architecure CapsUnet as shown in 5.4. These networks have been trained with ground truth containing pathology segmentation mask. Therefore the bottleneck layer for both the architecture must contain relevant pathological information. We therefore use these pathological features as an input to a classification network, and fine-tune only the classification layers while keeping the weights of the encoder part fixed.

6.3 Network Architecture

The network is adapted from the Segmentation networks used for pathology segmentation in Chapter 4. We have used here only the encoding part of the networks to generate the *Deep Pathological Features*, then using the obtained encoded features, we fine-tune a classifier network consisting of Fully connected layers. The final architecture is shown in fig 6.1.

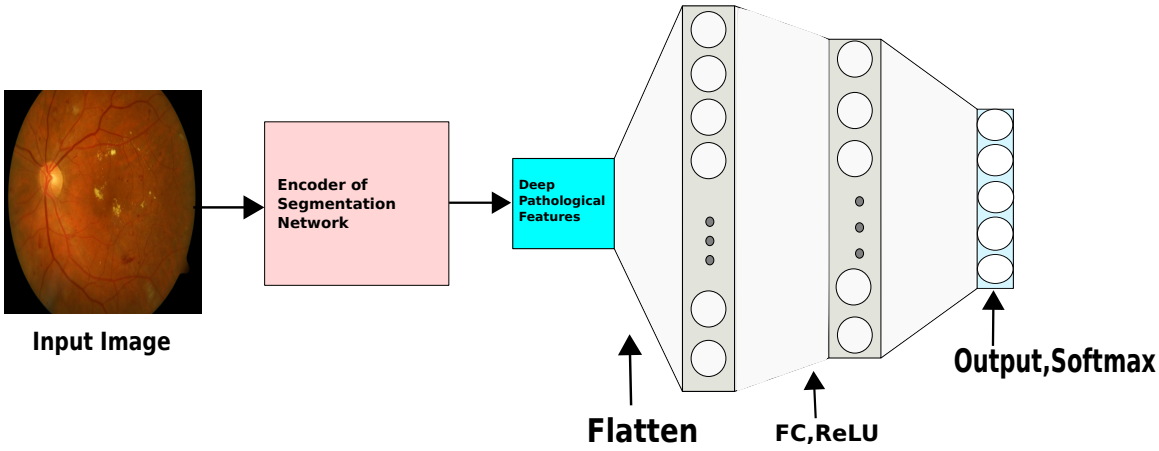

Figure 6.1: Classification Network Architecture

6.4 Experimental Details

For training the above network we have used the publicly available EyePacs dataset. The dataset consists of 35,126 color fundus images. Table 4.1 shows the distribution of classes among the dataset. The dataset is splitted into training, valdidation and test set, in the ratio of 70:30:10. The validation set and the test set are made balanced. The network is trained on a single Nvidia P100 GPU for 100 epochs. The training mechanism adopted by us is as follows:

- We have used the minibatch gradient descent algorithm with a batch size of 30. As, the dataset has imbalanced classes(the number of training samples are different among classes), we used a balanced mini-batch gradient descent algorithm to train our network.
- Initially we train the network without the ordinal regression layer, by optimizing the categorical cross-entropy loss function.
- Finally we add the ordinal regression layer, and optimize the Mean Squared Error loss.

6.5 Results

Figure 6.2, 6.3, 6.4, shows the confusion matrices for τ value 0.4,0.5,0.6.

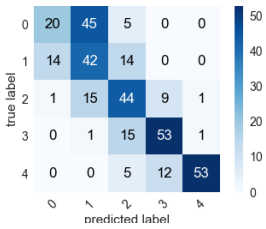

Figure 6.2: $\tau = 0.4$

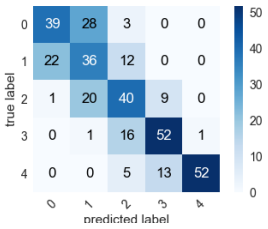

Figure 6.3: $\tau = 0.5$

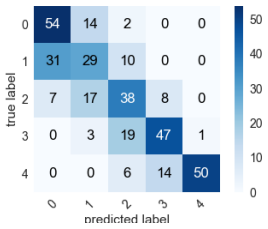

Figure 6.4: $\tau = 0.6$

The test set consists of 350 Fundus images with 70 images belonging to each class. We have evaluated our model by computing the Quadratic weighted Kappa metric, class wise sensitivity or the true positive rate, class wise specificity or the true negative rate and balanced overall accuracy, The evaluated metrics are shown in table 6.1. The *Quadratic weighted Kappa* obtained is 0.85 for our best model.

τ	**Class_0**		**Class_1**		**Class_2**		**Class_3**		**Class_4**		**Accuracy**
	TPR	TNR	TPR	TNR	TPR	TNR	TPR	TNR	TPR	TNR	
0.4	0.28	**0.96**	**0.6**	0.78	**0.62**	0.86	**0.75**	0.92	**0.75**	0.99	60.57
0.5	0.55	0.91	0.51	0.82	0.57	**0.87**	0.57	0.92	0.74	0.99	**62.57**
0.6	**0.77**	0.86	0.41	**0.87**	0.54	0.86	0.67	0.92	0.71	0.99	62.28

Table 6.1: Table showing the TPR(Sensitivity) , TNR(Specificity) and Accuracy for τ values

Table 6.1 shows the comparison of models with different values of τ. From the confusion matrix we can see that the model performs slightly better than the Attention model shown in chapter 4.1. Table 6.1 shows the True Postive Rate(Sensitivity) and True Negative Rate(Specificity) and the overall accuracy of the model for τ value 0.4, 0.5, 0.6.

Chapter 7

Conclusion and Future Work

In the project we have achieved a balanced test data set accuracy of around 63%. Along with grading the fundus image our model has the ability to locate the lesions responsible for the Retinopathic condition. We have used *Visual Attention mechanism* to achieve the above results. In future we aim to further extend this work to achieve a better performance mainly in terms of Sensitivity of the Stage 1 Diabetic Retinopathy, as it is in this stage the detection of the disease is crucial for it's prevention.

Bibliography

[1] Kaggle: Diabetic retinopathy detection. ((2015)), https://www.kaggle.com/c/diabetic-retinopathy-detection

[2] Acharya, U.R., Lim, C.M., Ng, E.Y.K., Chee, C., Tamura, T.: Computer-based detection of diabetes retinopathy stages using digital fundus images. Proceedings of the institution of mechanical engineers, part H: journal of engineering in medicine 223(5), 545–553 (2009)

[3] Antal, B., Hajdu, A.: An ensemble-based system for microaneurysm detection and diabetic retinopathy grading. IEEE transactions on biomedical engineering 59(6), 1720–1726 (2012)

[4] Bravo, M.A., Arbeláez, P.A.: Automatic diabetic retinopathy classification. In: 13th International Conference on Medical Information Processing and Analysis. vol. 10572, p. 105721E. International Society for Optics and Photonics (2017)

[5] Congdon, N.G., Friedman, D.S., Lietman, T.: Important causes of visual impairment in the world today. Jama 290(15), 2057–2060 (2003)

[6] He, K., Zhang, X., Ren, S., Sun, J.: Identity mappings in deep residual networks. In: European conference on computer vision. pp. 630–645. Springer (2016)

[7] Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4700–4708 (2017)

[8] Islam, S.M.S., Hasan, M.M., Abdullah, S.: Deep learning based early detection and grading of diabetic retinopathy using retinal fundus images. arXiv preprint arXiv:1812.10595 (2018)

[9] Jaderberg, M., Simonyan, K., Zisserman, A., et al.: Spatial transformer networks. In: Advances in neural information processing systems. pp. 2017–2025 (2015)

[10] Kori, A., Chennamsetty, S.S., Alex, V., et al.: Ensemble of convolutional neural networks for automatic grading of diabetic retinopathy and macular edema. arXiv preprint arXiv:1809.04228 (2018)

[11] de La Torre, J., Valls, A., Puig, D.: A deep learning interpretable classifier for diabetic retinopathy disease grading. Neurocomputing (2019)

[12] LaLonde, R., Bagci, U.: Capsules for object segmentation. arXiv preprint arXiv:1804.04241 (2018)

[13] Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3431–3440 (2015)

[14] Melville, A., Richardson, R., Mason, J., McIntosh, A., O'keeffe, C., Peters, J., Hutchinson, A.: Complications of diabetes: screening for retinopathy and management of foot ulcers. BMJ Quality & Safety 9(2), 137–141 (2000)

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

[16] Roychowdhury, S., Koozekanani, D.D., Parhi, K.K.: Dream: Diabetic retinopathy analysis using machine learning. IEEE Journal of Biomedical and Health Informatics 18(5), 1717–1728 (Sep 2014)

[17] Sabour, S., Frosst, N., Hinton, G.E.: Dynamic routing between capsules. In: Advances in neural information processing systems. pp. 3856–3866 (2017)

[18] Silberman, N., Ahrlich, K., Fergus, R., Subramanian, L.: Case for automated detection of diabetic retinopathy (2010)

[19] Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)

[20] Sopharak, A., Uyyanonvara, B., Barman, S.: Automatic exudate detection from non-dilated diabetic retinopathy retinal images using fuzzy c-means clustering. sensors 9(3), 2148–2161 (2009)

[21] Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.A.: Inception-v4, inception-resnet and the impact of residual connections on learning. In: Thirty-First AAAI Conference on Artificial Intelligence (2017)

[22] Wang, S., Yin, Y., Cao, G., Wei, B., Zheng, Y., Yang, G.: Hierarchical retinal blood vessel segmentation based on feature and ensemble learning. Neurocomputing 149, 708–717 (2015)

[23] Wang, Z., Yang, J.: Diabetic retinopathy detection via deep convolutional networks for discriminative localization and visual explanation. arXiv preprint arXiv:1703.10757 (2017)

[24] Xi, E., Bing, S., Jin, Y.: Capsule network performance on complex data. arXiv preprint arXiv:1712.03480 (2017)

[25] Yang, Y., Li, T., Li, W., Wu, H., Fan, W., Zhang, W.: Lesion detection and grading of diabetic retinopathy via two-stages deep convolutional neural networks. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 533–540. Springer (2017)