EAST WEST UNIVERSITY

B. SC. ENGINEERING THESIS

Biomedical Image Segmentation using Deep Learning

Authors:
Nusrat Jahan DOLAMONY
(**2015-2-50-019**)
Md. Zahid REZA
(**2015-2-50-007**)
Nabila TABASSUM
(**2015-2-50-029**)

Supervisor:
Muhammad Suhail NAJEEB
Lecturer
ELECTRONICS AND COMMUNICATIONS ENGINEERING

A thesis submitted in fulfillment of the requirements for the degree of Bachelor of Science in Information and Communication Engineering

September 2019

Letter of Acceptance

The thesis entileted "Biomedical Image Segmentation using Deep Learning" submitted by Nabila Tabassum (ID: 2015-2-50-029), Md. Zahid Reza (ID: 2015-2-50-007) and Nusrat Jahan Dolamony (ID: 2015-2-50-019) to the Electronics and Communication Engineering Department, East West University, Dhaka-1212, Bangladesh is accepted as satisfactory for partial fulfilment of requirements for the degree of Bachelors of Science (B. Sc.) Information and Communications Engineering.

____

Dr. Mohammed Moseeur Rahman
Assistant Professor Chairperson
Department of Electronics and Communications Engineering
East West University

____

Muhammad Suhail Najeeb
Lecturer
Department of Electronics and Communications Engineering
East West University

Declaration of Authorship

We, Nabila Tabassum, Md. Zahid Reza and Nusrat Jahan Dolamony, declare that this thesis titled, “Biomedical Image Segmentation using Deep Learning” supervised by Muhammad Suhail Najeeb and the work presented in it are our own. We confirm that:

- This work was done wholly while in candidature for Bachelor of Science in Information and Communication Engineering degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where we have consulted the published work of others, this is always clearly attributed.
- Where we have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely our own work.
- We have acknowledged all main sources of help.
- Where the thesis is based on work done by ourselves jointly with others, we have made clear exactly what was done by others and what we have contributed ourselves.

____

Nabila Tabassum
2015-2-50-029

____

Md. Zahid Reza
2015-2-50-007

____

Nusrat Jahan Dolamony
2015-2-50-019

Abstract

The process of solving medical issues by evaluating images created in clinical workout is known as medical image analysis.The goal is to obtain data for enhanced clinical diagnosis in an effective way. In this article presents a systematic overview of the present advance stage of medical image analysis using deep convolutional networks and also aims to develop automated methods for hippocampus brain MRI segmentation to decrease the time-consuming workload done by radiologist. We compared our results with hand-labeled segmentation done by medical radiologist. Our automated segmentation agreed well with human raters using a leave-one-out approach and standard overlap and distance error metrics,any differences were comparable with differences between trained human raters. Our error metrics compare favorably with those previously reported for other automated segmentations of hippocampus, suggesting the effectiveness of the approach to large-scale studies.

Acknowledgements

We would like to express our most sincere gratitude to our supervisor Muhammad Suhail Najeeb for his patient guidance, continuous support and advice. He has always been very helpful and inspired us. We have been extremely lucky to have a supervisor who cared about our work, and who responded to our questions and queries so promptly. He was available for our every question and whenever we needed him. Without his help this thesis would remain unsuccessful. We are truly honoured and to have him as our supervisor.

Contents

List of Figures

Dedicated To Our Parents . . .

Chapter 1

Introduction

1.1 Motivation

In this advance medical technology time the progress in biomedical image segmentation has made medical image analysis one of the top areas of research and development. For this development the implementation of Artificial Intelligence(machine learning algorithm) to analyze medical images is one of the important factor. Neural network can learn attributes automatically by the effective use of Deep learning, a machine learning method. This compares with those techniques where hand-crafted characteristics are traditionally used.Selecting these characteristics and calculating them is a difficult job. Now-a-days for medical image analysis Deep convolutional networks is a must. This involves fields of implementation such as segmentation, detection of abnormalities, classification of diseases, diagnosis and recovery assisted by computer.

MRI scans were segmented before by radiologist/ oncologist.Since they had to do it manually it was time consuming and patient's health were at risk. Deep learning is faster, can be very useful for radiologist since it can to segment the whole image. Recently U-net is being used widely for medical image segmentation. It is build upon fully convolutional neural network and yields better segmentation in medical imaging. U-net is symmetrical and the links between the down-sampling path and the up sampling path apply a concatenation operator rather than a sum[]. Since medical data set sometimes can be small, it is easier to use u-net and train the machine learning model easily with even small dataset. Another advantage of the u-net is the resolution of the image remains almost same as input. For our task we have modified the u-net into a smaller one to apply this method for smaller resolution image.

1.2 Organization of thesis

For the task we had very small dataset of brain hippocampus. To extend our dataset and to train the machine more effectively we have augmented our dataset by using various augmentation techniques. Also to get more accurate data we have preprocessed the images. For our task we have modified the unet into "mini unet" to train the machine for smaller resolution image. After training the result had 0.67 dice coefficient, which means it has given us almost accurate result. We have organized the thesis by following manner:

Chapter 2: In this chapter we have discussed the literature review of this thesis. Then we have discussed about the history and development of CCN, FCN and U-net.

Chapter 3: This chapter describes how digital system are used recently in hospitals, what is biomedical imaging, processing and used metrices.

Chapter 4: This chapter contains necessary informations about image segmentation and masking.

Chapter 5: In this chapter we have discussed about the dataset we have used, the preprocessing process and the augmentation techniques.

Chapter 6: This chapert contains the the results and discussions about the result

Chapter 7: The results and findings of the thesis are summarized in this chapter. The future aspects of this thesis are also discussed.

Chapter 2

Literature Review

This review highlights the tremendous achievement of Deep convolutional network, machine learning algorithms applied to medical image analysis and clinical characteristics of this field. Deep convolutional network is a pattern recognition instrument, where a neural network can automatically learn characteristics rather than traditionally hand designed characteristics, and selecting these characteristics which was a difficult task.

2.1 Convolutional neural networks (CNN)

CNNs were used in Sahiner et al's job for medical image processing as long ago as 1996[1]. In this job, mammograms obtained ROIs representing either biopsy-proven masses or ordinary tissues.The CNN consisted of an input layer, two hidden layers, and an output layer. Training times in this pre-GPU age have been defined as "computationally intensive," but no times have been provided. In 1993[2], CNNs introduced a dual-match process and for lung nodule detection an artificial visual neural network technique .This neural network method is suitable for recognizing medical picture pattern in gray scale modeling, and the artificial neural network design is a modified network structure of human vision.This artificial neural network is rather than complete association with weighted multiplication as a two-dimensional local convolution. A CNN was used in 1995[3] to identify mammography micro-classification.This advancement of a computer program for the automated detection of clustered mammography microcalcifications and the use of a signal classifier based on CNNs, the classification accuracy of CNNs for ROIs (Regions of Interest) is calculated by the analysis of ROC (Receiver Operating Characteristics).

The typical CNN image processing architecture includes a sequence of convolution filter layers, classification with pooling layers.Convolution layer are implemented in human brain in low-level vision processing, convolution filters detect progressively appropriate picture characteristics, such as lines or circles that may represent straight edges (such as for organ identification) or circles (such as round objects such as colonic polyps), and then higher-level characteristics such as local and global form and texture.Typically, the CNN output is one or more labels. Convolution filters are learned from training data and reduce the time-consuming hand-crafting of characteristics that are needed to calculate computable characteristic. The other forms of network architecture, as like a profound recurring neural network known as long-term memory.

However, longevity was still incomplete, . Subsequently, other classifiers were created, including decision trees, vector machines boosting and supporting. Each of these was applied to medical image analysis, in particular to detect anomalies, but also in associated areas such as segmentation.As shown by such innovations, the norm was comparatively boosted false positive detection rates.

Many of the latest developments in computer vision are due to the effective implementation on graphics processing units (GPUs) of convolutional neural networks (CNN). GPU acceleration has greatly accelerated computations, enabling very profound and complicated models to be trained on big data sets.For examples, Krizhevsky et al used non-overlapping neurons and a very effective GPU application of the convolution process to accelerate training and decrease over fitting in the fully connected layers created a regularization technique called "dropout" and the output result was very efficient and entered a version of this model in the ILSVRC-2012[4] contest and reached a winning top-5 test error rate of 15.3 percent compared to 26.2 percent at the second-best entry level.Nearly have the error rate from one year to the next on the Image Net challenge dataset by training a profound CNN on over one million pictures on two GPU cards. CNNs are efficient because they are able to understand solely data-driven depictions of the hierarchical features of the picture.

Features that are useful for classification are learned from the pictures just provided a monitoring signal that specifies the required output for classification. Konstantinos Kamnitsas, in March 18, 2016[5] proposed a dual path, 11-layer profound, three-dimensional Convolutionary Neural Network for the difficult assignment of segmentation of brain lesions.The designed module is the result of a comprehensive analysis of the limitations proposed for similar applications by current networks. To overcome the computational burden of 3D medical scanning, this provides a dual path architecture that simultaneously processes multi-scale input pictures to integrate both local and global contextual data.For post-processing soft segmentation of the network, this uses of a completely linked 3D Conditional Random Field that efficiently removes false positives and widely evaluates the pipeline on three difficult lesion segmentation assignments in multi-channel MRI patient information with traumatic brain injuries, brain tumors, and ischemic stroke.This improves the state of the art for all three platforms, with top performance rankings on the 2015 and 2015 BRATS and ISLES government benchmarks. This so-called "monitored learning" has lately been applied to numerous scientific fields, including biomedical and radiological imaging.

A CNN typically contains of several layers of neural network layers that are convolutionary, pooled, and fully linked (or tightly linked). The convolutionary layers use spatial correlation in the input pictures by exchanging the weights of the filter kernel for each function map computation .[6]Pooling layers enable each input feature map to be reduced while maintaining the most appropriate feature explanations. Commonly used pooling involves max or average pooling . Max-pooling may also add some invariance to the objects ’ local shift in the picture input.Typically, the outputs of each CNN layer are provided to non-linear activation (often rectified linear units (ReLUs). Using non-linear activation functions enables us to model complicated mappings between the input picture and the required outputs .Figure 1 indicates a schematic overview of a typical CNN architecture that uses a soft max output for multi-class classification to produce a forecast per image. For many medical imaging functions, this form of architecture has been effectively implemented .Some examples in radiology include anatomy classification false positive decrease for computer-aided identification of lymph nodes or colonic polyps, detection of pulmonary nodules and pulmonary embolism detection. CNNs were also effectively used to classify endoscopic video sequences, e.g. in colonoscopy[or laparoscopic surgery][7].

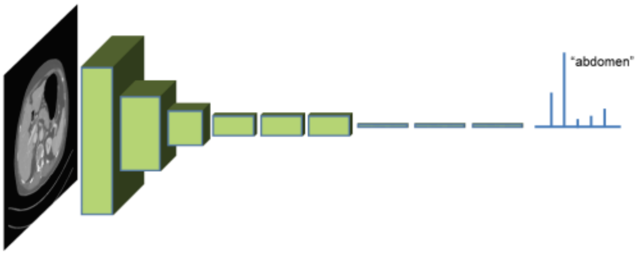


Fig. 2.1. Convolutional neural network (CNN).

2.2 Fully convolutional networks (FCN)

When the convolutionary features are inserted finally into the network’s fully connected layers One downside of CNNs is that the image’s spatial information is lost.Long et al[6]. suggested the Fully Convolutionary Network (FCNs) and showed that trained end-to-end, pixel-to-pixel semantic segmentation exceeded prior best outcomes without further equipment.The typical FCN setup is shown in Fig. 2.2 For semantic CT image slice segmentation[8].

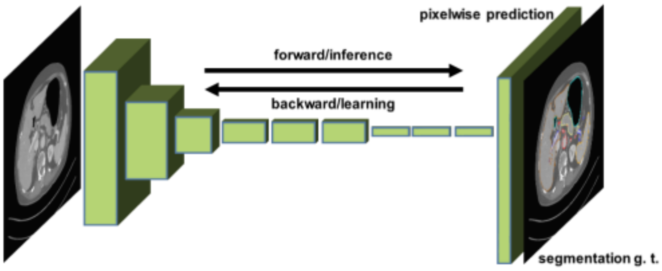


Fig. 2.2. Fully convolutional network (FCN)

2.3 3D Fully convolutional networks (3D U-Net)

The implementations of 3D convolution, growing GPU memory have made it successful to extend these process to 3D medical imaging and train networks on large amounts of annotated image volumes. One such example is the recently proposed 3D U-Net architecture[6]. Advanced implementation of 3D convolution and growing GPU memory has further enabled these methods to be extended to 3D medical imaging and networks to train large amounts of annotated image volumes. One such example is the 3D U-Net architecture recently proposed.The implemented network extends Ronneberger et al's previous u-net architecture by replacing all 2D operations with their 3D counterparts[9]. Implementation performs elastic deformations on - the-fly during training to efficiently increase data.It is trained from top to bottom, i.e. there is no need for a pre-trained network. This tests the performance of the proposed method on the Xenopus kidney, a complex, highly variable 3D structure, and achieves good results in both use cases. We describe how the 3D U-Net architecture can be applied to the multi-organ segmentation problem in CT. The typical FCN configuration is shown in below Fig For semantic CT image slice segmentation[10].

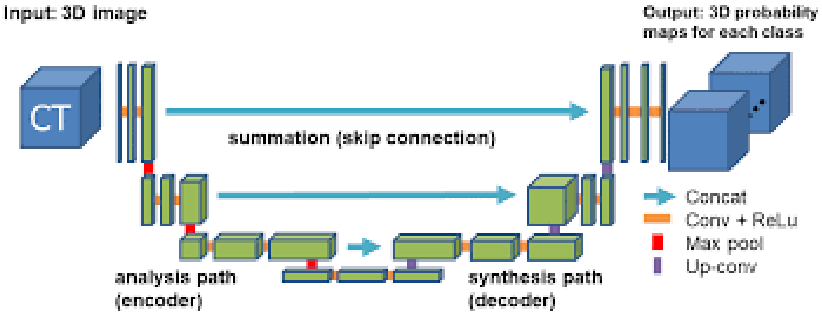


Fig. 2.3. Architecture of U-net for 3D CT image

Chapter 3

Biomedical Imaging

3.1 Biomedical Image retrieval

In this present time computers and digital information systems are use in hospitals at most of the time for biomedical image segmentation. A large number of medical images are produced by hospitals and radiology departments, ultimately resulting in huge repositories of medical images. Therefore, for better management of medical image information, the more effective methods of recovery are required[11].For retrieving medical images there are two ways, text-based and content-based methods. Text-based retrieval methods were initially proposed in the 1970s, where the images are retrieved using manually annotated text descriptions with a text-based image retrieval, and traditional image management techniques are used.In data-based image retrieval, the images are retrieved on the basis of features such as colour, texture, shape and so on that were derived from the images in a timely manner and can be accurately misleading with regard to the diagnosis and stage of a patient's disease.

Such systems have the disadvantage of being unable to perform well in unnoted image databases. Annotation of images is not just a subjective issue, but also a time-consuming process[**?**]. Texture, color and shape-based characteristics are used in CBIR methods to search and retrieve images from large data collections.

3.2 Biomedical Imaging

Medical imaging involves procedures that provide the human body with visual data. Biomedical imaging is intended to help radiologists and clinicians make the process of diagnosis and therapy more efficient[12]. Medical imaging is a predominant part of disease diagnosis and treatment and represents various modalities of imaging. These include X-ray, CT, magnetic resonance imaging (MRI), positron emission tomography (PET), and ultrasound to name a few, as well as hybrid modalities.These modalities play a vital role in the diagnosis and research of anatomical and functional information about different body organs[13].Figure 4 shows a typology of popular medical imaging methods used for various body components produced in radiology and laboratory settings. In modern healthcare systems, medical imaging is an essential aid[14].In CADx, machine learning plays a vital role in its applications in tumor segmentation, cancer detection, classification, imaging guided therapy, medical image annotation and recovery[15].

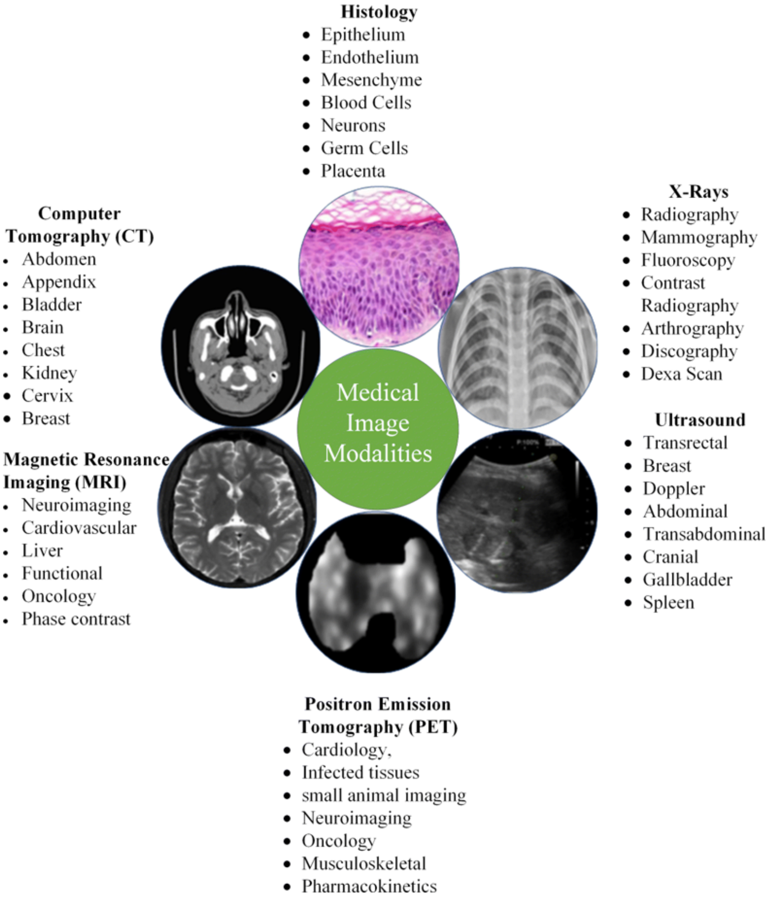


Fig. 3.1. Typology of medical imaging modalities

3.3 Biomedical Image Processing

Biomedical image processing is conceptually as like as multi-dimensional biomedical signal processing[12]. It involves the assessment, improvement and display of x-ray, ultrasound, MRI, nuclear medicine and optical image techniques captured pictures. Image reconstruction and modeling techniques make it possible to create 3D images instantly processing 2D signals[16]. It took literally hours to resulting one slice of image data when the original CT scanner was invented in 1972 and more than 24 hours to reconstruct that data into a single image[17]. This acquisition and reconstruction takes place in less than a second in present time. The image processing software helps to automatically identify and analysis which will not be apparent to the general human eye rather than simply eyeball an x-ray on a light-box. CNNs algorithms can provide effective way to detect tumor patterns and indicative characteristics and other disorders[18].

By the help of Biomedical Image Segmentation and the diagnosis being considered, image processing and analysis can be used to determine a tumor or organ's diameter, volume and vas culature ; flow parameters of blood or other fluids ; and microscopic changes that have yet to raise any otherwise discernible flags[19].

3.4 Evaluation metrics for medical image analysis system

Various main performance measures such as accuracy, F1-score, accuracy, recall, sensitivity, specificity, and dice coefficient are used to evaluate a typical medical image analysis scheme.

F1score = 2× (P recision × Recall) (P recision + Recall), (1)

where, P recision = T P (T P + FP), (2)

and

Recall = (T P) (T P + TN), (3)

Accuracy = (T P + TN) (T P + T N + FP + FN), (4)

Sensitivity = T P (T P + FN), (5)

Specificity = T N (T N + FP), (6)

DiceScore = 2 × |P ∩ GT | |P|+|GT | , (7)

Where True Positive (TP) represents the number of cases correctly recognized as defective, False Positive (FP) represents the number of cases incorrectly recognized as defective, True Negative (TN) represents the number of cases correctly recognized as undefected and False Negative (FN) represents the number of cases incorrectly recognized as undefected[20]. In Eq. 7, P refers to the prediction given by the system being evaluated for the test sample and GT refers to the ground truth of the test sample.

Chapter 4

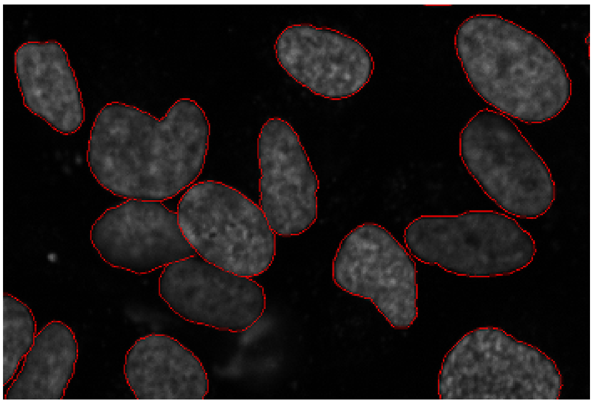
Segmentation Basics

4.1 Image segmentation

Image segmentation is a mechanism of digital image dividing into various sections (pixel sets, also known as ultra-pixels). The objective of segmentation is to simplify simply by making the representation of a picture more meaningful and easier to understand to evaluate[21].

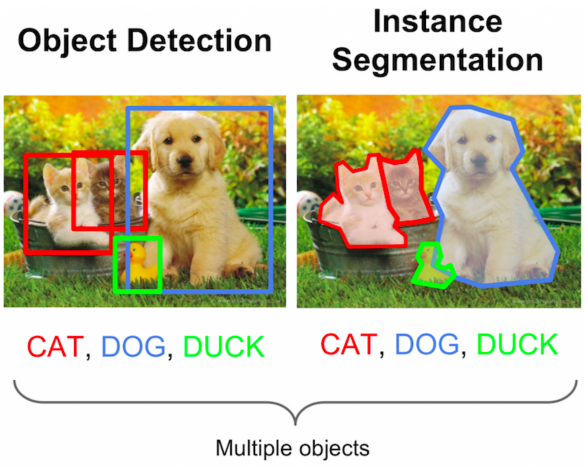
4.2 Need for Image Segmentation

Cancer has been a lethal disease for a long time. Cancer can be fatal even in today's age of technological advances if we don't recognize it early on. The earliest possible detection of cancer cells can possibly save millions of lives. Cancer cell shape plays a crucial role in determining cancer seriousness[21]. You might have put the pieces together – it won't be very useful here to detect objects. Only bounding boxes will be generated that will not assist us to identify the shape of cells. The methods of image segmentation have a MASSIVE effect here. They assist us get to grips with this issue in a more granular way and get more significant outcomes. A win - win in the healthcare industry for everyone.



4.3 Working procedure of image segmentation

We can split the picture into different components called segments or partition it. Processing the whole image at the same time as there will be regions in the image that do not contain any information is not a great idea . We can use the significant sections to process the picture by splitting the picture into sections[21]. That's how picture segmentation operates, in a nutshell. A picture is a multi-pixel collection or set. We group together the pixels that use picture segmentation to have comparable characteristics. Take a moment to go through the visual below (the picture segmentation will offer you a practical concept):



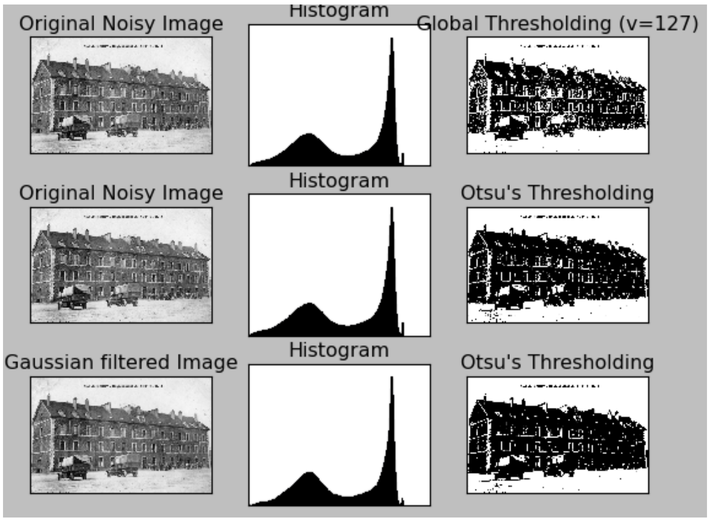
Object detection creates a bounding box in the picture that corresponds to each class. But it does not tell us anything about the object's form. We only get the coordinates set of bounding boxes. We want to get more data – for our purposes this is too vague. Image segmentation for each object in the image creates a pixel-wise mask. This method allows us to understand the object(s) in the picture much more granularly. Why are we going to have to go deep? Is it not possible to solve all image processing functions with easy bounding box coordinates? To answer this relevant question, let's take a real-world instance. There are three general approaches to segmentation, called thresholding, methods based on edge, and techniques based on region.

4.4 Threshold Segmentation

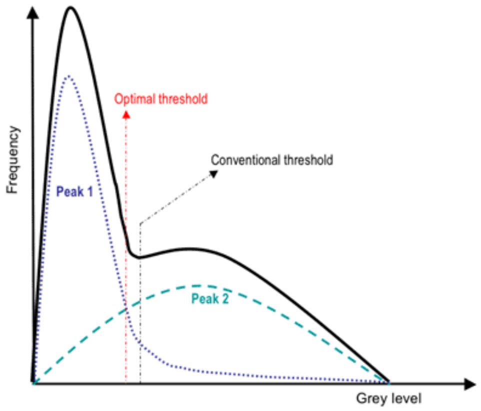
Threshold is the easiest method for non-contextual segmentation. It converts a gray scale or color picture into a binary picture as a binary area map with a single limit[22]. The binary map contains two possibly disjoint regions, one of them containing pixels with input data values smaller than a threshold and another relating to the input values that are at or above the threshold. The former and latter areas are generally labeled respectively with zero (0) and non-zero (1) labels. The segmentation relies on the threshold picture property and how the threshold is selected.

4.4.1 Simple thresholding

The most common threshold image property is the gray pixel level: g (x, y) =0 if the threshold is f (x, y) <T and g (x, y) =1 if the threshold is f (x, y) or T. A variety of gray levels linked to region 1 can be described using two thresholds: g (x, y) =0 if f (x, y) < T1 OR f (x, y) > T2 and g (x, y) = 1 if T1= f (x, y) = T2.



The main problems are whether it is possible and, if yes, how to choose an adequate threshold or a number of thresholds to separate one or more desired objects from their background[22]. In many practical cases the simple thresholding is unable to segment objects of interest, as shown in the above images. A general thresholding method is based on the premise that pictures are multimodal, i.e. different items of concern are associated with separate peaks (or modes) of the 1D signal histogram. Despite typical overlaps between the signal ranges corresponding to individual peaks, the thresholds must distinguish these peaks optimally. A threshold in the valley between two overlapping peaks distinguishes their primary bodies, but some pixels with intermediate signals are inevitably detected or mistakenly rejected. The ideal threshold that minimizes the anticipated number of false detections and rejections may not coincide between two overlapping peaks with the smallest point in the valley:

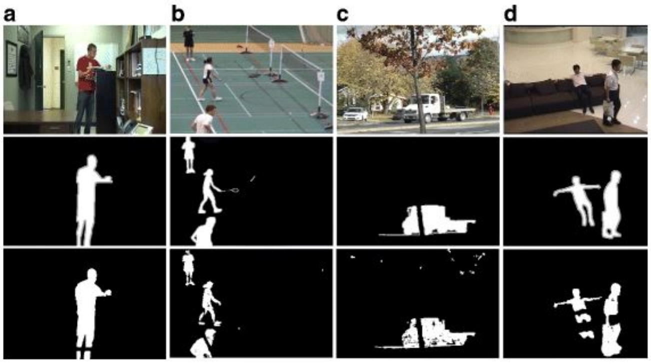


4.4.2 Adaptive thresholding

Because the threshold distinguishes the background from the object, the adaptive separation may take into account object (e.g. dark) and background (bright) pixel distributions of empirical probability[22]. Such a threshold must match two types of anticipated mistakes: assign a background pixel to the item and assign a background pixel to the item. More complicated adaptive thresholding methods use a spatially variable threshold to offset local spatial context effects (such a spatially variable threshold can be considered as background normalization). The classification of the object and background pixels is done at each iteration j by using the threshold Tj found at previous iteration. Thus, at iteration j, each grey level f (x, y) is assigned first to the object or background class (region) if f(x,y) Tj or f(x,y) > Tj, respectively. Then, the new threshold, Tj+1 = 0.5(j,ob + j,bg) where j,ob and j,bg denote the mean grey level at iteration j for the found object and background pixels, respectively:

4.4.3 Color thresholding

Color segmentation may be more precise due to more pixel-level data compared to gray-level pictures. The normal color depiction of Red-Green-Blue (RGB) has strong color components and a number of other color schemes (e.g. HSI Hue-Saturation-Intensity) have been designed in order to exclude redundancy, determine actual object / background colors irrespectively of illumination, and obtain more more stable segmentation. An example below [22] shows that color thresholding can focus on an object of interest much better than its greyscale analogue:



Segmentation of color images involve a partitioning of the color space, i.e. RGB or HSI space. One simple approach is based on some reference (or dominant) color (R0, G0, B0) and thresholding of Cartesian distances to it from every pixel color f(x,y) = (R(x,y),G(x,y),B(x,y)):

$$g(x,y)=\begin{cases}1 & \text{if } d(x,y)\le d_{\max}\\ 0 & \text{if } d(x,y)> d_{\max}\end{cases};\quad d(x,y)=\sqrt{(\mathrm{R}(x,y)-\mathrm{R}_0)^2+(\mathrm{G}(x,y)-\mathrm{G}_0)^2+(\mathrm{B}(x,y)-\mathrm{B}_0)^2}$$

where g(x,y) is the binary region map after thresholding. This thresholding rule defines a sphere in RGB space, centered on the reference color. All pixels inside or on the sphere belong to the region indexed with 1 and all other pixels are in the region 0. Also, there can be an ellipsoidal decision surface if independent distance thresholds are specified for the R, G, and B components. Generally, color segmentation, just as the grey scale one, may be based on the analysis of 3D color histograms or their more convenient 2D projections. A color histogram is built by partitioning of the color space onto a fixed number of bins such that the colors within each bin are considered as the same color. An example below of the partitioned $11 \times 11 \times 11$ RGB color space is from.

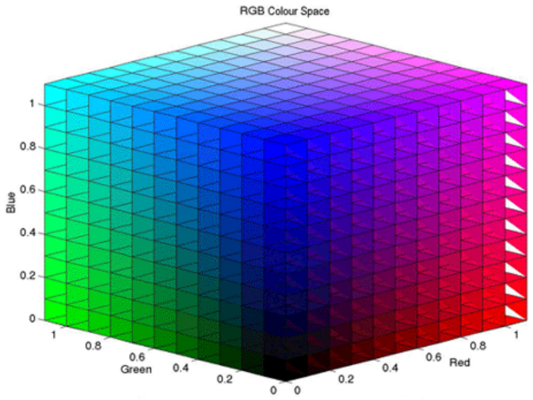


Fig. 4.1. RGB Color Space

4.5 Region-based Segmentation

Regional segmentation algorithms operate iteratively by grouping neighboring pixels with similar values and dividing pixel groups that differ in value. Growing seeded region is a semi-automatic merging type technique. We're going to clarify it as an instance. Figure: (a) displays a collection of radius 3 white seeds positioned inside all muscle fibers using a computer mouse-controlled on-screen cursor. Figure: (b) demonstrates again the edge filter output from Prewitt. The seeds and limits of a segmentation generated by a watershed algorithm type are superimposed on it in white.Figure: (c) also shows the limits superimposed on the initial muscle fiber picture[22].The watershed algorithm (we will explain the name later) works as follows

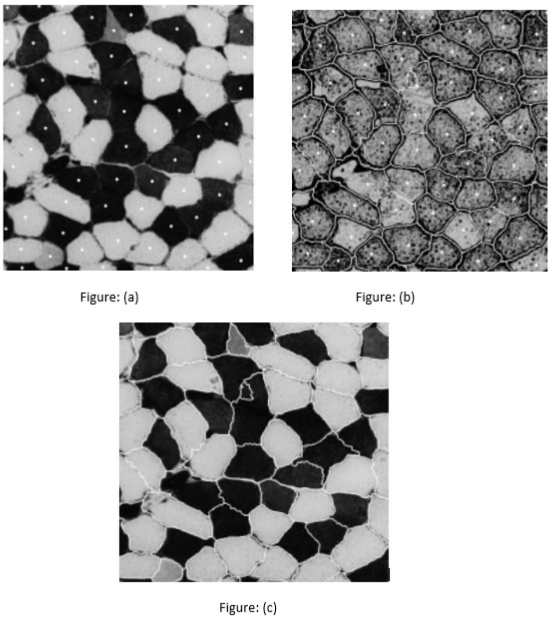


Figure: (a)

Figure: (b)

Figure: (c)

Fig. 4.2. Manual segmentation of muscle fibres image by use of watersheds algorithm: (a) manually positioned 'seeds' in centres of all fibres, (b) output from Prewitt's edge filter, together with watershed boundaries, (c) watershed boundaries superimposed on the image.

The normal use of the watershed algorithm is completely automatic. Again, by illustration, we will show this. Figure demonstrates the production of a Gaussian weights variance filter (§ 3.4.1) applied to the picture of muscle fibers after histogram-equalization.

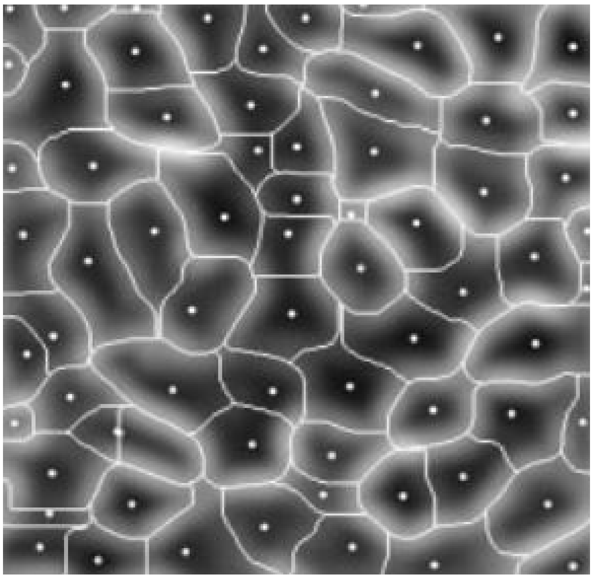


Fig. 4.3. Figure: Output of variance filter with Gaussian weights ($\sigma 2$ = 96) applied to muscle fibers image, together with seeds indicating all local minima and boundaries produced by watershed algorithm.

There are many other algorithms based on the region, but most of them are quite complex. We will consider only one more in this chapter, namely an elegant split-and-merge algorithm suggested by Horowitz and Pavlidis (1976). To segment the log-transformed SAR picture , we will present it in a mildly altered form, basing our segmentation choices on variances, whereas Horowitz and Pavlidis are based on their pixel range. The algorithm works in two phases and needs a maximum variance limit to be specified in pixel values in a region. The first phase is the one that splits. Initially, the entire image's variance is calculated.The picture is split into four quadrants if this variance exceeds the designated limit.Likewise, if the variance exceeds the threshold in any of these four quadrants, it is further subdivided into four. This goes on until the entire picture consists of a collection of squares of variable dimensions, all of which vary below the limit. (Note that this must be achieved by the algorithm because the variances are taken to be zero at the finest resolution of each square consisting of a single pixel.) The indicates the resulting white borders overlapping the log-transformed SAR picture with the variance limit set at 0.60. Note that:

- In non-uniform areas of the picture, squares are narrower.

- In non-uniform areas of the picture, squares are narrower.

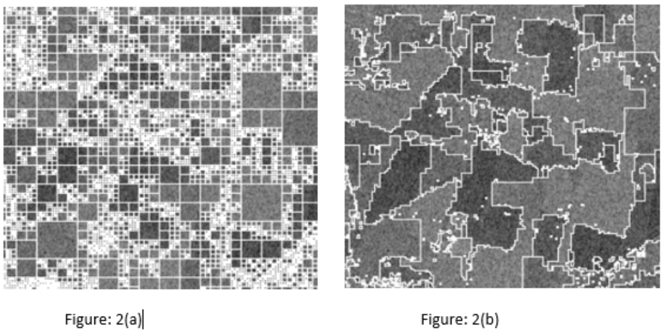


Fig. 4.4. Region-growing segmentation of log-transformed SAR image: (a) division of image into squares with variance less than 0.60, obtained as first step in algorithm, (b) final segmentation, after amalgamation of squares, subject to variance limit of 0.60.

- The strain limit was set at 0.60 instead of the speckle variance value at 0.41 (Hogan, 1994), as the resulting areas were very low in the latter case. The algorithm needs the size of the picture, n, to be a 2 force. The 250 x 250 SAR picture was therefore filled out to 256 x 256 by adding width 3 boundaries.

The second stage of the algorithm, the merging one, involves merging squares with a common edge, provided that this does not exceed the limit of the new region's variance. Upon completion of all amalgamations, the outcome is a segmentation in which each region has a variance below the specified threshold. However, although the outcome of the algorithm's first phase is unique, that from the second phase is not — it depends on the order of which squares are regarded. Fig: 2(b) shows the boundaries produced by the algorithm, superimposed on the SAR image. Dark and light fields appear to have been successfully distinguished between, although the boundaries are rough and retain some of the artefacts of the squares in Fig 2:(a). Pavlidis and Liow (1990) suggested by merging the outcomes with those from an edge-based segmentation to overcome the deficiencies in the limits generated by the Horowitz and Pavlidis algorthm. Many other concepts have been suggested for region-based segmentation (see, for instance, Haralick and Shapiro review, 1985), and it is still an active rese region.

4.6 Edge-based segmentation

Usually the findings of threshold-based segmentation are less than ideal, as we have seen[22]. A scientist will often have to change the automatic segmentation outcomes. One easy way to do this is to manage a screen cursor with a computer mouse and draw border lines between areas. Figure: 3(a) shows the boundaries obtained by thresholding the image of muscle fibers (as shown in Figure :3(a)), superimposed on the output from the edge filter of Prewitt (§ 3.4.2), with the contrast stretched so that values between 0 and 5 are displayed as shades of gray ranging from white to black and values above 5 are all displayed as black. This display can be used to help determine where to insert additional limits to fully segment all muscle fibers. Figure: 3(b) indicates the outcome after 71 straight lines have been added manually.

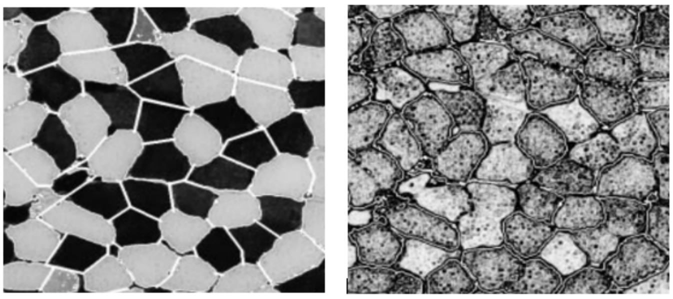


Figure : 3(a)

Figure: 3(b)

Fig. 4.5. 3(a) Boundaries acquired by thresholding the picture of muscle fibers, superimposed on the output of Prewitt's filter, with values between 0 and 5 displayed in gray colors gradually darker and values above 5 displayed as black. 3(b) Manual picture segmentation by adding additional lines to threshold limits, superimposed in white on the initial picture.

Algorithms are accessible for semi-automatic drawing of edges, smoothing and disturbing the rough lines of the scientist to maximize some criterion of matching with the picture (see, for instance, Samadani and Han, 1993). Alternatively, it is possible to make edge finding fully automatic, though not necessarily successful. The outcome of applying Prewitt's edge filter to the muscle fiber picture is shown in Figure (a). The filter output was thresholder at a value of 5 in this screen: all pixels above 5 are labeled as edge pixels and shown as black. Connected edge pixel chains split the picture into areas.

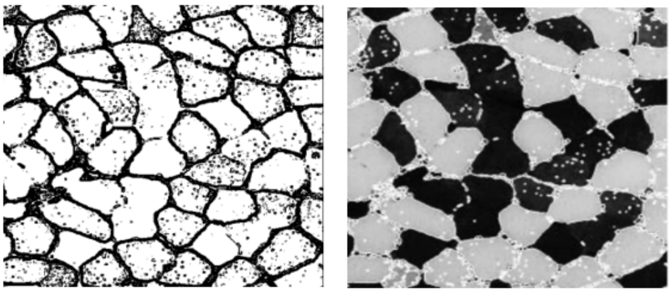


Figure: 4 (a)

Figure: 4 (b)

Fig. 4.6. (a) Thresholded output of Prewitt's edge filter applied to the picture of muscle fibers: values higher than 5 are shown as black, values less than or equal to 5 as white. (B) boundaries generated in (a) from linked areas, superimposed in white on the initial picture.

Segmentation can be accomplished by assigning all non-edge pixels not separated by an edge to a single category. Rosenfeld and For 4- and 8-connected areas, Pfalz (1966) provided an effective algorithm, called a connected component algorithm. This algorithm will be described in words, then mathematically.[23]

4.7 Segmentation using Image processing

Image processing is a method used to convert an image into a digital form and perform certain procedures on it in order to achieve an enhanced image or extract useful data. Its primary parts are importing, where an image is captured by scanning or digital photography ; the picture is analyzed and manipulated using multiple specific software applications ; and output (e.g. to a printer or screen)[24]. Image processing is widely used in many fields, including astronomy, medicine, industrial robotics, and satellite remote sensing. It is a sort of signal dispensation where input is image, such as video frame or image, and output may be image or image or characteristics related with that image. Image Processing System treats images as two-dimensional signals and applies them in already defined techniques of signal processing. Image processing goes through following three steps:

1. to import the image via image acquisition tools ;
2. to analyze and manipulate the picture ;
3. to modify the resulting picture or report based on image analysis[25].

Some techniques for image processing are[24]

- Image Editing, which basically implies using graphical software tools to alter images.
- Image Restoration, which relates to estimating the initial smooth image from the corrupt image taken to get back the lost data.
- Independent Component Analysis dividing a multivariate signal into additive subcomponents computationally.
- Anisotropic Diffusion, often referred to as Perona-Malik Diffusion, makes it possible to decrease image noise without removing important image components.
- Linear Filtering. Another digital image processing method is the processing of time-varying input signals and the generation of output signals subject to linearity constraints.
- Neural networks, which are computational models that are commonly used to solve different functions in machine learning.

- Pixelation, often referring to digitizing printed images (such as GIF).
- Principal Components Analysis, a method for digital image processing that can be used to extract features.
- Partial Differential Equations, which also deals with images to de-noise efficiently.
- Hidden Markov Models, used to analyze images which are in two dimension.
- Wavelets, which is a mathematical function used in the compression of images.
- Self-organizing maps, a method for image processing to classify images into several classes.

4.8 The Segmentation Mask

Interpretability of the neural network is a important aspect for systems across a variety of domains. One way to explain neural networks is through a information mask to show which input information is accountable for the choice. In this work, we present a method for producing pre-trained network explanation masks. Our masks define the most significant components of the input to predict accurately. Masks are developed by a secondary network whose objective is to produce an explanation as low as possible while maintaining predictive precision. We show the applicability of our technique for CNN image classification, RNN sentiment analysis and combined CNN / RNN prediction of physical properties. it is shown in figure 4.23

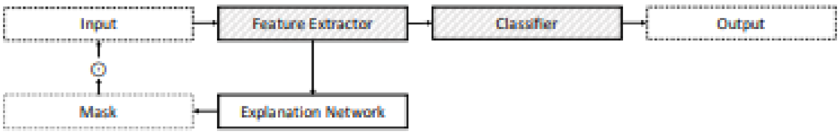


Fig. 4.7. Architecture of the explanation mask. The pre-trained (top) network is divided into a component to extract and classify features. The function extractor output feeds into an explanation network, generating a mask multiplied with the input element-wise. The masked input is supplied to the final output through the initial network. Shading shows weights that are frozen.

The explanation network structure can be any type that matches the issue assignment, leaving the technique quite flexible. Figure 1 shows a schematic of the general structure. First, the pre-trained network is divided into a component to extract and classify features. Then the output of this extraction function stage is fed into the network of explanations. Given the characteristics of the input

4.9 Experiments

The picture classification dataset CIFAR10 (Krizhevsky Hinton, 2009) to evaluate the visual information capability of our model. Here ResNet164 v2 (He et al., 2016a) as a base model, which is a residual network of 164 layers. The model achieves 94.04 percent test precision. The 256-length vector ResNet generated due to a final soft ax selection as input to the explanation network. The vector is transformed into a matrix of 16 or 16 and passed through 4 convs. blocks. There are 4 2D convs in each block. ELU enabled layers and 64 filters. Padding ensures that the image's size remains continuous. The first and final conv are summed by a residual link (He et al., 2016b). Activations of layers. The mask is up sampled to 32 after the first block, the image size. After the final block, a single 2D convolution with filter size 1 and sigmoid activation finishes with the explanation network. The mask is increased and applied similarly across all color channels by the initial input. Sparsity is provided by regularization of L1 and L2. We provide representative examples in Figure 2:

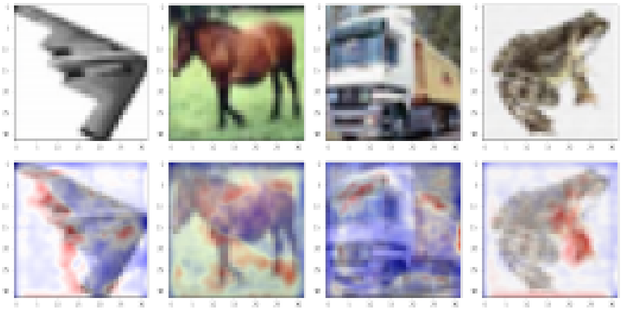
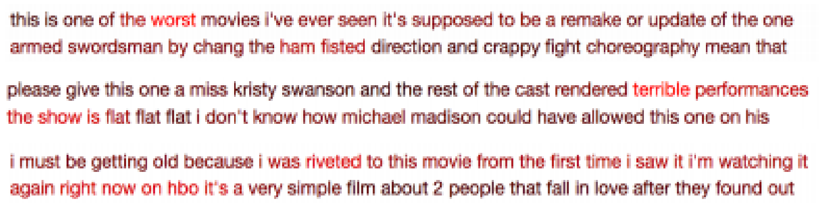


Fig. 4.8. CIFAR10 images (top) along with their learned explanation masks (bottom).

Sentiment Analysis

With the IMDB review sentiment project, we also test our method to text classification (Maas et al., 2011). With a vocabulary size of 20,000 and a maximum length of 500 words, we preprocess the input. Our basic model utilizes pre-trained 100-dimensional GloVe embedding with 40 percent-word dropout to avoid overfitting. A bidirectional GRU layer of length 256 follows this. This layer's output is averaged over time steps and passed to a fully connected layer of 128 dimensions with activation of ReLU and a final dense softmax. We are modifying Adam's binary crossentropy. We have achieved a test efficiency of 84.7 percent after 10 epochs. The complete output of the bidirectional GRU layer is used as input by our explanation network. We use a 100-width bidirectional 2-layer GRU. The output of each time step is passed to a dense layer of size 256 with ReLU activation before a sigmoid layer generates a single value representing the mask of the word.



this is one of the worst movies i've ever seen it's supposed to be a remake or update of the one armed swordsman by chang the ham fisted direction and crappy fight choreography mean that

please give this one a miss kristy swanson and the rest of the cast rendered terrible performances the show is flat flat flat i don't know how michael madison could have allowed this one on his

i must be getting old because i was riveted to this movie from the first time i saw it i'm watching it again right now on hbo it's a very simple film about 2 people that fall in love after they found out

Fig. 4.9. IMDB reviews with words highlighted based on explanation mask weights).

weight. We regularize by penalizing mask entropy. After 10 epochs we achieve a test accuracy of 81.8%. Figure 3 presents example sentences with words highlighted by mask weights. In Table 1 we also present the most attended words from positive and negative reviews. weight. By penalizing entropy of the mask, we regularize. We achieve a test accuracy of 81.8% after 10 epochs. Figure 3 presents sample phrases with mask weights highlighted words. The most attended expressions from beneficial and negative reviews are also presented in Table 1.

4.10 Using Deep Learning

Deep learning is a form of artificial intelligence which simulates the functioning of the human brain in information processing and generating patterns for decision making use. Deep learning is a subset of artificial intelligence (AI) machine learning it has networks capable of understanding from unorganized or unidentified information without supervision.Deep learning might have developed hand in hand with the computer age, resulting in an increase of information in all kinds that from all areas of the universe. Along with other sources like social media, internet search engines, e-commerce platforms, and digital films, this information, known actually as big data. This massive amount of information is easily available [26].

How deep learning generates of this kind outstanding results: in a phrase, precision. Deep learning generates identification effectiveness at higher levels than before. This allows consumer electronics to satisfy user expectations and seems to be essential for safety-critical applications like driverless vehicles. Recent developments in deep learning have risen to the extent that certain functions, like cl, have been implemented. such as classifying objects in pictures, deep learning outperforms people. While deep learning was first theorized in the 1980s, there are two main reasons it has only recently become useful: 1. Deep learning needs big quantities of information marked. For example, it takes millions of images and thousands of hours of video to develop driverless cars. 2. Deep learning needs considerable electricity in computing. High-performance GPUs have an effective parallel architecture for profound learning. This allows development teams to decrease training time for a deep learning network from weeks to hours or less when coupled with clusters or cloud computing. Examples of Deep Learning at Work: Automated Driving:Automobiles research teams use deep learning to automatically understand goods such as stop signs and traffic lights. To addition, deep learning is used to identify pedestrians, which helps reduce accidents. Aerospace and Defense: Deep learning is being used to classify areas of research for satellites and to detect safe or unsafe areas for soldiers. Medical Research:Deep learning is also used by cancer researchers to classify cancer cells instantly. UCLA teams have developed a unique microscope that gives a high-dimensional collection of data used it to train a deep learning application for specific cancer cell detection. Industrial Automation: Deep learning serves to ensure the safety of workers around heavy equipment by automatically recognizing people or products within an unsafe range of devices. Electronics: Home support tools that relate to our voice and recognize our preferences are driven.Deep learning is used in automatic hearing and speech translation.

4.10.1 Neural Networks

Deep learning becomes part of a bigger family of machine learning methods based on artificial neural networks. Artificial neural networks were inspired by the processing of information and the production of communication nodes in biological systems. Neural networks, a magnificent programming framework that encourages a computer to learn from experimental data Deep learning, a solid set of neural network learning techniques[27].

Neurons

Neuron: A element in a neural network comprising of a prematurely quantity of neurons. A neural network has various layers. A neuron is a function (x1, x2,..., xn), a femoral function that uses f as input and delivers a binary output and a weight factor multiplied by the operate of sigmoid and calculates how often this neuron is considered for the output of the layer.

Neurons are the basic unit of a neural network. Basically, the neurons involve multiple axons (inputs), a cell nucleus (processor) and an axon (output). When the neuron activates, all of its incoming inputs are gathered, and a message passes through axon when it exceeds a certain threshold. sort of. The key thing for neurons is that they can learn. Artificial neurons look more like this:

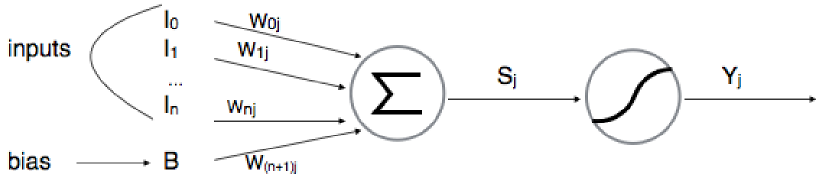


Fig. 4.10. Neuron Neural Network

There is a weight for each input as you can see that they have multiple outputs. When the artificial neuron activates, it computes its condition by adding all incoming inputs multiplied by their linked attachment weight. Neurons, however, always have one extra input, the distortion, which is always 1 and is associated with its own weight.This ensures that there would be activation in the neuron even though all inputs are none (all 0s). Where guan is all inputs (such as bias) After measuring its situation, the neuron passes it through its activation function, which normalizes the outcome (usually between 0-1).

Activations

Functions for activation are statistical equations that evaluate the output of a neural network. The function is attached to each neuron of the network and determines whether it should be activated ("fired") or not depending on whether each neuron's input is relevant to the model's prediction. Activation functions often attempt to normalize each neuron's output to a range of 1 to 0 or -1 to 1. Another element of the activation features is they are computationally demanding because thousands or even millions of neurons are calculated for each information sample the model, which places the activation function and its derivative function with an enhanced computational strain.

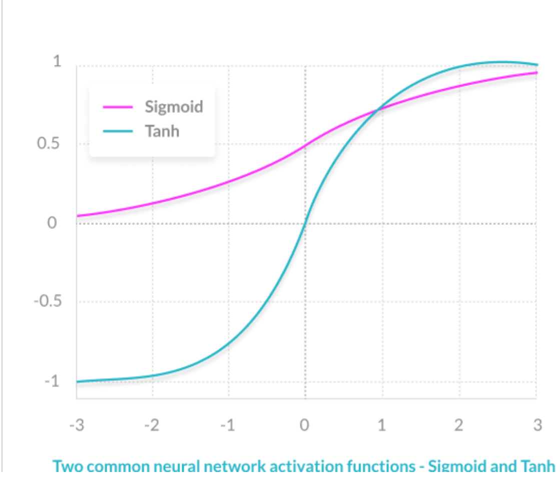


Fig. 4.11. Activation

Elements of a Neural Network: -

Layer of input:-This layer recognizes characteristics of input. All of this offers the network with data from the outside world, no computation is done on this layer, networks here just pass the information(features) on to the hidden layer. Hidden Layer :- Nodes of this layer are still not presented in the outside world, they are function of any neural network's abstraction. Hidden layer calculates all kinds of characteristics entered through the input layer and transfers the outcome to the output layer. Output Layer :- This layer introduces for the outside world the data the network has gained.

What is an activation function and why to use them?

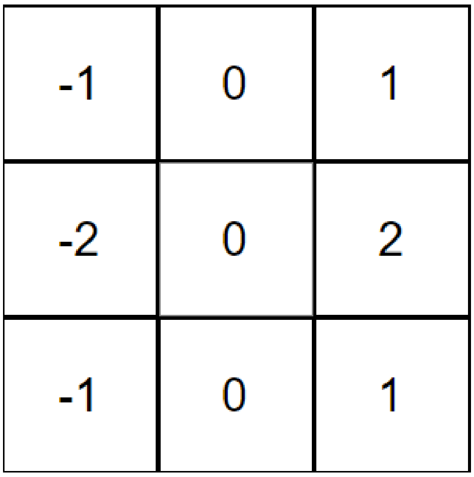
Definition of activation function: - Activation function chooses whether or not to activate a neuron by calculating weighted sum and adding bias with it further. The activation function's goal is to bring nonlinearity into a neuron's output.

Explanation :- We understand that the neural network has neurons that operate in weight, prejudice and their corresponding activation function accordingly. In a neural network, based on the output mistake, we would update the weights and biases of the neurons. This method is called back-propagation. Activation functions make the back-propagation possible since the gradients are supplied along with the error to update the weights and biases.

4.10.2 Convolutional Neural Networks

Convolution

They are basically just neural networks using Convolutionary layers, a.k.a. Conv layers, depending on convolution's mathematical operation. Conv layers are a collection of filters that you can only think of as 2d numbers matrices. Here's an example 3x3 filter:



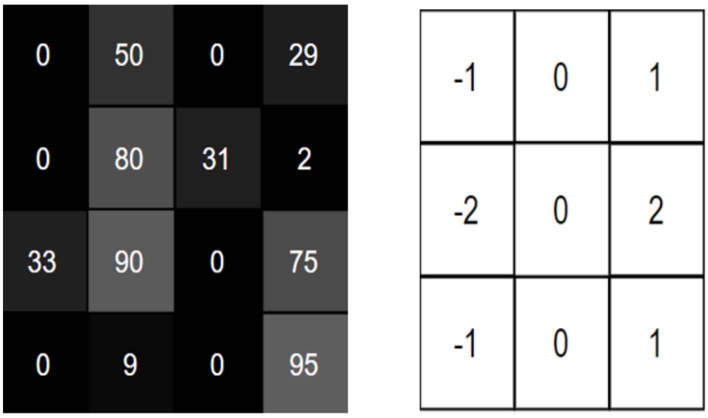
-1	0	1
-2	0	2
-1	0	1

Fig. 4.12. 3x3 filter

Fig: 3x3 filter use for an input image and a filter to produce an output image by convolving the filter with the input image.

1. Overlay the filter at some place on top of the picture. 2. Multiplication of the element-wise values in the filter and the respective values in the picture. 3. Summing up all the products that are element-wise. This amount is the output value in the output picture for the target pixel. 4. Repeat for all sites.[**?**]

That description of 4 steps was a little abstract, so let's take an example. Consider this small image of 4x4 grayscale and this filter of 3x3:

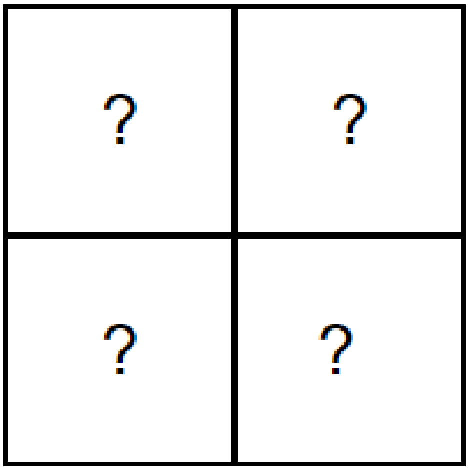


0	50	0	29
0	80	31	2
33	90	0	75
0	9	0	95

-1	0	1
-2	0	2
-1	0	1

Fig. 4.13. 4x4 image (left) and 3x3 filter (right).

The picture figures depict the levels of intensity of pixels, where 0 is black and 255 is white. To generate a 2x2 output image, we will convert the input image and the filter:



?	?
?	?

Fig. 4.14. 2x2 filter.

output image

To start, lets overlay our filter in the top left corner of the image:

Fig. 4.15. Step 1: Overlay the filter (right) on top of the image (left)

Next we multiply element-wise between the overlapping values of the picture and the values of the filter. Here are the findings, from the top left corner to the right, then down:

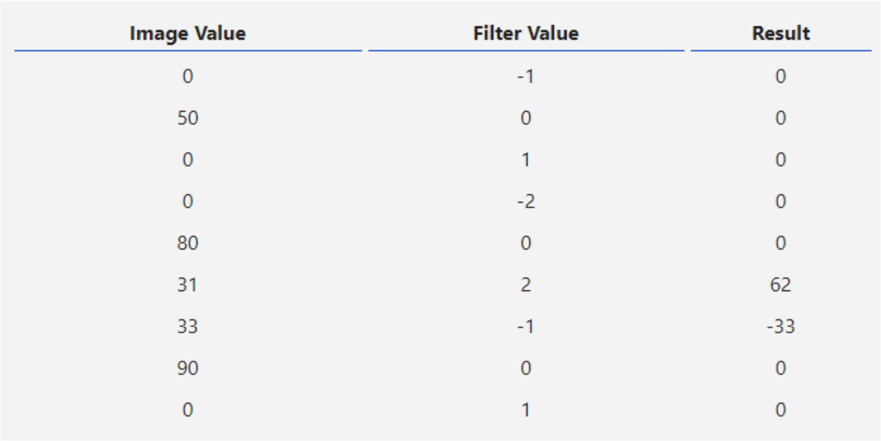
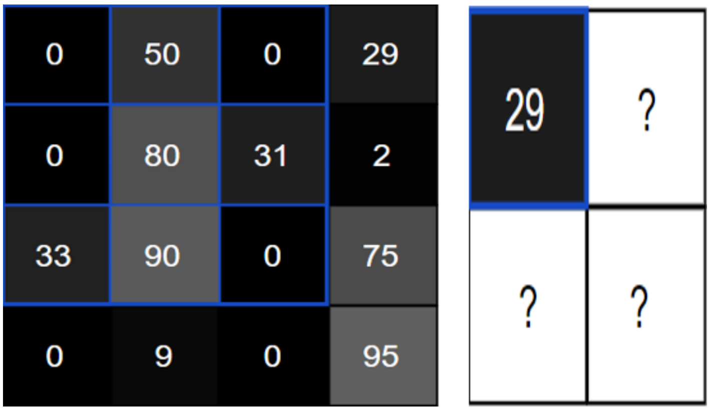


Image Value	Filter Value	Result
0	-1	0
50	0	0
0	1	0
0	-2	0
80	0	0
31	2	62
33	-1	-33
90	0	0
0	1	0

Fig. 4.16. Step 2: Performing element-wise multiplication.

Next, we sum up all the results. That's easy enough: 62 - 33 = $\boxed{29}$6233=29

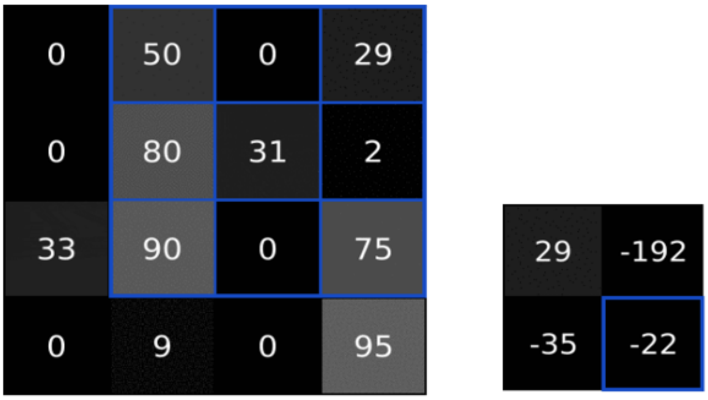
Finally, we position our outcome in our output image's destination pixel. As our filter is overlaid in the top left corner of the input picture, the top left pixel of the output picture is our target pixel:



0	50	0	29
0	80	31	2
33	90	0	75
0	9	0	95

29	?
?	?

Fig. 4.17

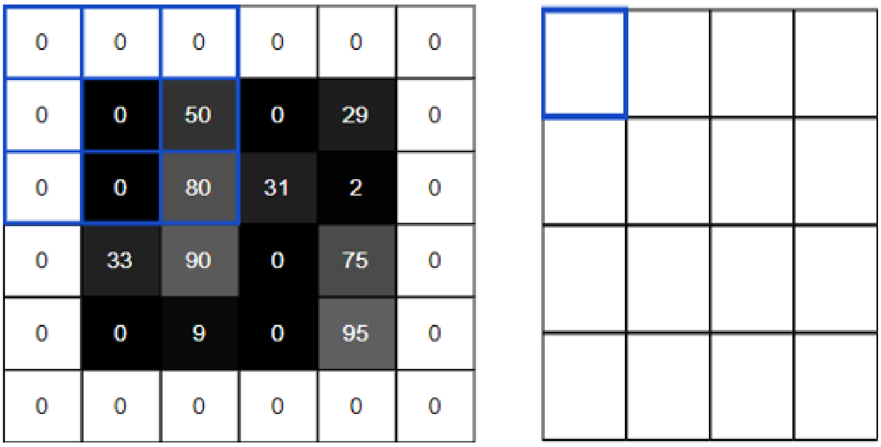


0	50	0	29
0	80	31	2
33	90	0	75
0	9	0	95

29	-192
-35	-22

Fig. 4.18. Here do the same thing to generate the rest of the output image.

Padding: Remember to convert a 4x4 input picture to generate a 2x2 output picture with a 3x3 filter previously? We will often prefer the output picture to be the same size as the input picture. To do this, we are adding zeros around the picture so that in more locations we can overlay the filter. 3x3 filter needs 1 pixel padding:



0	0	0	0	0	0
0	0	50	0	29	0
0	0	80	31	2	0
0	33	90	0	75	0
0	0	9	0	95	0
0	0	0	0	0	0

Fig. 4.19. 4x4 input convolved with 3x3 filter to produce a 4x4 output using same padding.

This is called "same" padding because the sizes of input and output are the same. It is sometimes referred to as "valid" padding not to use any padding, which is what we have done and will continue to do for this article.

Conv Layers: Since know how much the transformation of images works and why it is useful, let's see how it is effectively used in CNNs. As mentioned previously, CNNs include converting layers using a set of filters to convert input images into output images. A conv layer's primary parameter is the quantity of filters it has. For here MNIST CNN, will use as the previous layer in our network a relatively small conv layer with 8 filters. This implies that the 28x28 input picture will become a volume of output 26x26x8.

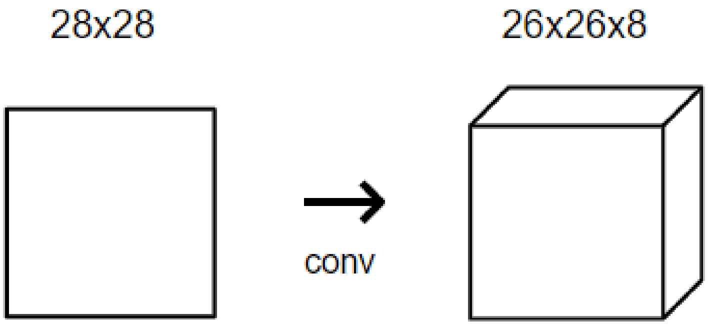


Fig. 4.20. Conv Layers.

Each of the 8 filters in the conv layer produces a 26x26 output, so stacked together they make up a 26x26x8 volume. All of this happens because of 3

4.10.3 Pooling

General pooling

The pooling technologies may also behave other tasks as well as max pooling, such as average pooling or even L2-norm pooling. Historically, average pooling is often used, but it has recently fallen by the wayside relative to the max pooling method, which has been shown to work better.

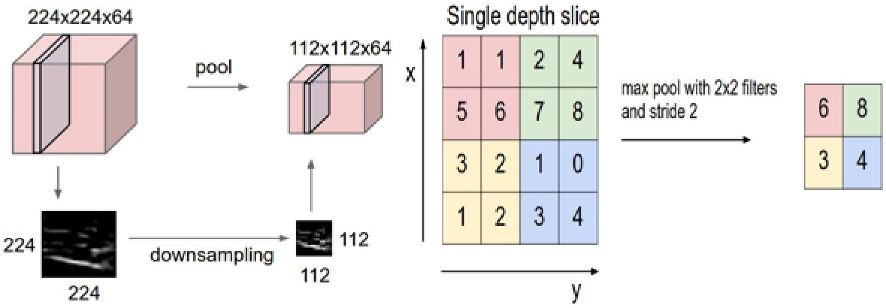


Fig. 4.21. Pooling

Pooling layer down samples the number in space, differently in each feedback volume shape slice. Left: In this example, the length input volume [224x224x64] is pooled into the size output volume [112x112x64] with filter size 2, step 2. Note that the depth of the quantity is maintained. Right:The most widespread down sampling procedure is max, resulting in max pooling, with a step of 2 shown here. That is, 4 numbers (little 2x2 square) are drawn over each max.

4.10.4 U-NET Architecture

U-NET: U-Net is regarded one of the regular CNN architectures for image identification tasks where it is necessary mostly to define this whole image by class, as well as to segment the image area by class, i.e. to create a mask that separates the image into several classes. The architecture comprised of a hiring route for context capturing and a symmetrical extension approach for exact localiztion.

The network is instructed in end-to-end operation from very few images and reaches the past perfect method on the ISBI challenge of segmenting neuronal structures in electron microscopic stacks (a convolutionary sliding window network).That use the same network hired on transmitted light microscopy pictures (stage contrast and DIC), U-Net gained a big margin in these categories in just the 2015 ISBI cell tracking challenge. The network is also quick. Segmentation of a image of 512 range512 on a contemporary GPU requires less than a second.

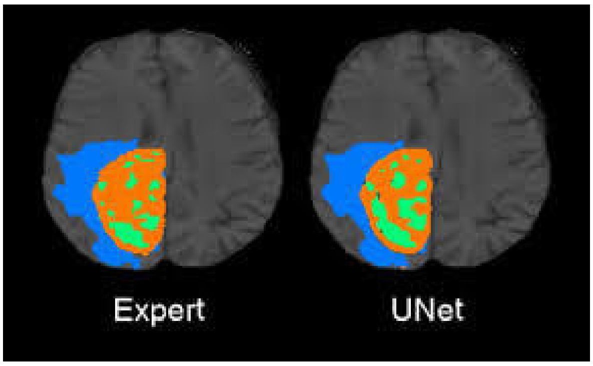


Fig. 4.22. Brain Segmentation

The U-net Architecture

Developed on the Fully Convolutional Network, the U-Net architecture is modified to improve segmentation in medical imaging. Compared to FCN-8, the two key differences are:

1. U-net is symmetric
2. The links between the down sampling route and the up sampling route apply a concatenation operator rather than a sum.

The aim of these skip connections is to provide local information when sampling global information. Because of its simplicity, the network has a large number of feature maps in the up sampling path, which allows information to be transmitted. By comparison, the basic FCN architecture contained only a number of class feature maps in its sampling path.[27]. The U-Net owes its name to its symmetric shape, which is different from other FCN variants. U-Net architecture is separated in 3 parts:

- The contracting/down sampling path
- Bottleneck
- The expanding/up sampling path

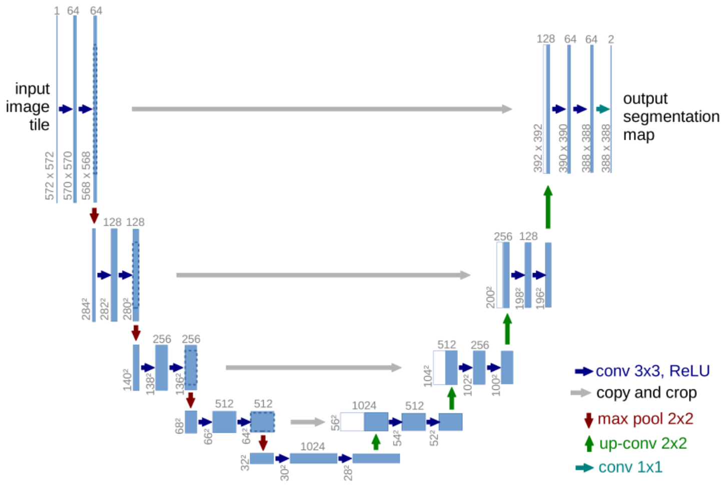


Fig. 4.23. U-net architecture (display at smallest resolution for 32×32 pixels). Each blue box connects to a function map of different channels. The set of channels on top of the box is stated. Thex-y-size is displayed at the box's upper left edge. White boxes are duplicate maps of features. The arrows represent the various actions.

Figure 1 explains the network architecture. It contains a agreement route (left side) and an extension path (right side). The standard network architecture follows the contracting path. It includes repeated application of two 33 covenants (unpadded covenants), each accompanied by a linear rectified unit (ReLU) and a 2×2 max pooling procedure with step 2 down sampling. New app channels are matched to each step of down sampling. Each stage in the evolving route comprises of a sampling of the function border followed by an up-convolution of 2×2 that halves the number of function segments, a concatenation with the correspondingly cropped function map from the contracting route, and two convolutions of 3×3, merged by a ReLU. The shading is necessary in each and every convolution due to the current loss of boundary pixels.

Advantages

1. High consistency of adequate training, appropriate data sets and time for training.
2. No heavy layer, then we can use images of various sizes as input (as the only dimensions to learn on convolution layers are the kernel, as well as the kernel size is independent of the size of the input image).
3.The use of huge information gains is significant in fields such as biomedical segmentation, as the percentage of samples annotated is generally restricted

Chapter 5

Data Preperation

5.1 Dataset

For our "Mini Unet" we have selected a data set of brain hippocampus. The hippocampus is a very small, curved brain formation that plays a momentous part in the limbic system. The hippocampus is connected in the creation of fresh memories, as well as learning and emotions.

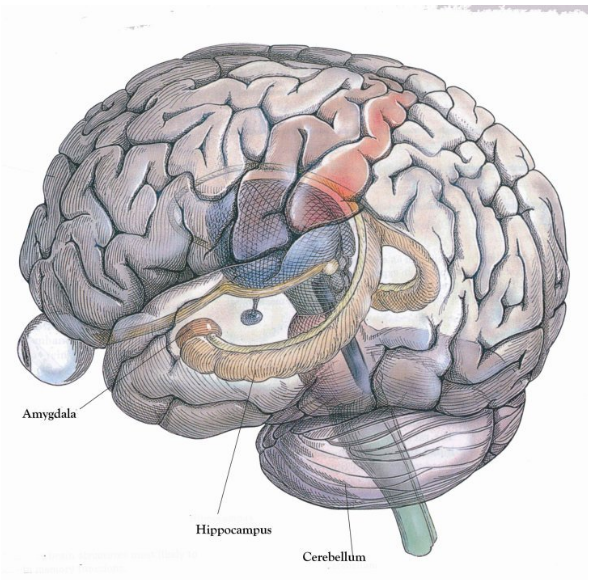


Fig. 5.1. Brain image showing the hippocampus

Our target was to segment hipocampus head and body. The dataset of the brain hippocamus contains mono modal MRI images. The data set has 394 images. Among them 263 are training and 131 are testing images. Each of the images are in 3D and provided in "nii" files. Resolution of the images are 40x48. The following figures shows some MRI scans of hippocampus and their labels.

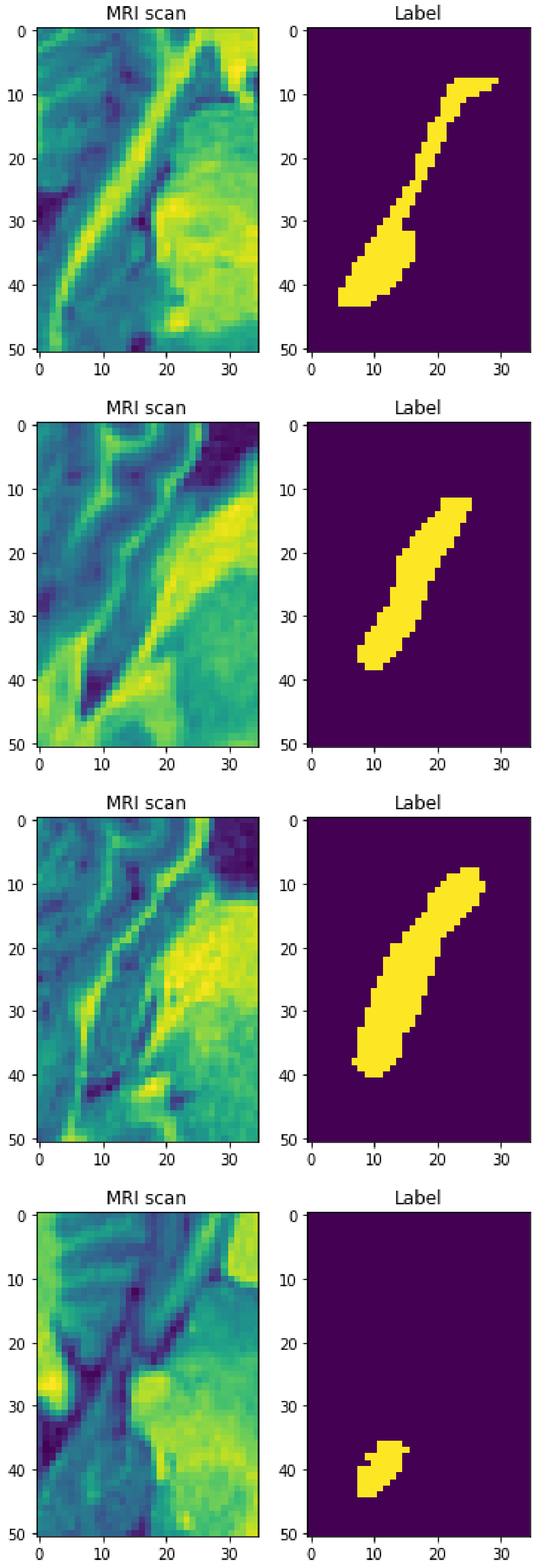


Fig. 5.2. MRI scans and labels

5.2 Preprocessing

Preprocessing relates to all the raw information conversions before they are fed into the machine learning algorithm or deep learning. It can imply any filtering applied to images, such as recognition and classification, to perform computer vision tasks. Training a convolutional neural network on raw pictures, will likely lead to poor performance in classification. Also preprocessing is essential to speed up training. The aim of this step is to make the data ready for the ML model to make it simpler to computationally analyze and process it, as with images[23].

Usually, the data we obtained are chaotic and come from various sources. They need to be standardized and cleaned up to feed them into the ML model (or neural network).Preprocessing is used more often than not to conduct steps that reduce complexity and increase the accuracy of the algorithm applied. For each of the conditions in which an image was taken, we can not write a distinctive algorithm, so when we obtain an image, we tend to transform it into a form that can be solved by a general algorithm.Some will claim that pre-processing images is not a good idea because they distort or alter the real nature of raw data. Smart use of pre-processing images, however, can provide advantages and solve issues that eventually lead to better identification of local and global features [28].

Since our dataset images are in nii files it is hard to extract images from these type of files. Preprocessing made it easier for us to extract the images that we needed

5.2.1 3D to 2D

Deep learning methods usually take several approaches to handle 3D information. This involves converting 3D data into 2D slices and combining 2D and multi-view features to take advantage of contextual information[29]. MRI scans are 3D image by default and segmentation methods works on 2D image. So we have taken slices of the 3D image, cut it by two axis to converted it in 2D.

5.2.2 Thresholding

Image thresholding is an effective and simple way of dividing an picture into the foreground and background.This technique is a sort of image segmentation technique that isolates objects by converting grayscale images into binary images. Image thresholding in high contrast images is most efficient[30]. The simplest thresholding methods replace each pixel in a black pixel image if the image intensity is lower than a fixed constant or a white pixel if the image intensity is greater.

It is helpful in many vision applications to be able to distinguish the image regions corresponding to the objects we are interested in from the image regions corresponding to the background.Usually this is performed to separate "object" or foreground pixels from background pixels to assist in the processing of images.Based on the distinct intensities or colors in the foreground and background regions of an image, thresholding often offers a simple and convenient way to execute this segmentation. In addition, it is helpful to identify which regions of an image consisting of pixels whose values are within a defined range or band of intensities (or colors).

Thresholding methods[31]

1. Using an iterative method to calculate a threshold value automatically.
2. Approximate the image histogram as a bimodal distribution and select the threshold level as the midpoint value.
3. Adaptive thresholding. Threshold is adaptive. Use alternating rows to evaluate the limit based on the last 8 pixels in each row. When used as part of the ImageEdgeDetection procedure, this technique is not endorsed.
4. Fuzzy thresholding with entropy measures the "fuzziness".
5. Fuzzy thresholding utilizes a method to minimize a "fuzziness" metric involving the mean gray level of the object and background.
6. Determines an ideal threshold by histogramming the data and representing the image as an iteratively decreased set of clusters until two clusters remain. Then the threshold value is set to the smaller cluster's lowest rate.
7. Determines the ideal threshold value by maximizing the total "object" and "background" variance.
8. Default method where we must use the /T flag to specify a threshold value.

MRI pixel value range is usually in between 0 to 4000. For machine error the value could exceed. To keep the pixel values in range we have used thereshold function in processing of the image to have the convenience to process the image.

5.2.3 Normalizing

Data normalization is a significant step that ensures that each input parameter (in this case pixel) has a comparable distribution of data. Basically, this process changes the range of pixel intensity values. While training the network, this makes convergence quicker. Normalization is sometimes referred to as stretching contrast or stretching histograms. It is also referred as dynamic range expansion in more particular areas of data processing, such as digital signal processing. When we normalize data we make the data have zero mean and unit variance with a formula such as[32]:

$$X_{norm} = (X - X_{min})/(X_{max} - X_{min})$$

With normalization, we get two advantages. First, if data is not normalized, characteristics with higher numerical values dominate features with lower numerical values and therefore we won't receive contributions from features with lower values. Second, with normalized data, many learning algorithms act well. This manifests for normalized data in greater test precision than non-normalized data[32].

Many machine learning algorithms try to discover data trends by comparing data point features. However, when the features are on drastically distinct scales, there is a problem. Consider, for instance, a housing data set. The amount of spaces in the house and the age of the house in years could be two potential features. An algorithm for machine learning might attempt to predict which building is the best. When the algorithm compares data points, the other is entirely dominated by the feature with the bigger scale. The objective of normalization is to ensure that each data point has the same scale so that each feature is equally essential[33].

Min Max Normalization

A common ways in which data can be normalized is min-max normalization. For every feature, the minimum value of that feature transforms into 0 and the maximum value transforms into 1, and every other value transforms into a decimal between 0 and 1. For example, if the minimum value of a feature is 20, and the maximum value is 40, then 30 will be transformed to about 0.5 since it is halfway between 20 and 40. The formula for min max normalization as follows:

$$\frac{value-min}{max-min}$$

Min-max normalization has one problem. It does not manage outliers very well. For example, if we have 99 values between 0 and 40, and one value is 100, then the 99 values will all be transformed to a value between 0 and 0.4. That data is squished and we can not get the proper value[33].

Z-Score Normalization

Z-score normalization can avoids the outlier issue of min max normalization. The formula for Z-score normalization:

$$\frac{value-\mu}{\sigma}$$

Here, μ is the feature's mean value, and σ is the feature's normal deviation. If a value is precisely the mean of all the function values, it will be normalized to 0. It is a negative number if it's below the mean, and if it's above the mean it is a positive number. The magnitude of these negative and positive numbers is determined by the initial feature's standard deviation. If there is a large standard deviation in the unnormalized data, the normalized values will be closer to 0. But the problem with Z-score normalization is it handles outliers, but does not produce normalized data with the exact same scale[33].

5.3 Augmentation

Data augmentation technique significantly increases the diversity of data available for training models, without actually collecting new data. Data augmentation is a way of creating new 'data' from prior information with distinct orientations. It is done by applying domain-specific techniques to examples from the training data that create new and different training examples. Image augmentation includes more variety to the training dataset and in the event that it is done right, reflects the variety within the genuine data and thus makes a difference in the model to generalize better[34].

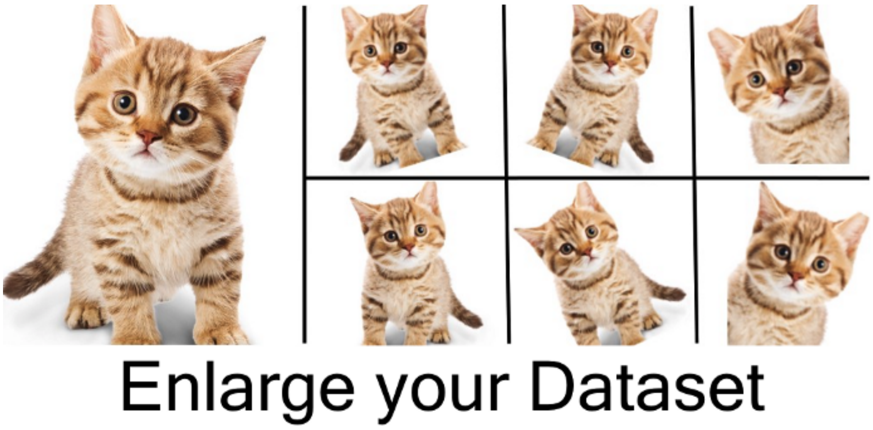


Fig. 5.3. Result of Augmentation of an image that enlarges the dataset

5.3.1 The role of augmentation techniques in Machine Learning

Training a model of machine learning implies it adjusts its parameters so that it can map a specific input (in this case picture) to some output (label). Here our objective is to ensure that the loss of the model is low, which occurs when parameters are correctly tuned if the machine has many parameters, a proportional number of examples should be shown to the machine learning model in order to achieve good performance. The amount of parameters is also proportional to the complexity of the task to be performed by the model. In our dataset, we only have 394 images. To get a better result this amount of data is insufficient. So we need the help of data augmentation to increase the training dataset. Training deep learning neural network models on more information can increase the model's skill, and augmentation methods can generate image variants that can enhance models capacity to generalize what they have learned to new images[34].

If each image in the training set is perfectly centered within the frame, traditional feed-forward models will be confused if the test image is slightly shifted from the background to the right. For example, if the portrait of each image of a cat is taken, the model will not recognize the cat when it turns its face to other angles. Modern deep learning algorithms such as the convolution neural network or CNN can learn features invariant to their location of image. In this transforming invariant learning approach, however, augmentation can further help the model in learning features that are also invariant to transformations such as left-to-right to top-to-bottom ordering, light levels in photographs and more[34].

5.3.2 Augmentation Techniques used:

Shift

Objects in the images may not be centered within the outline. They may be off-center in a variety of different ways. A shift to an image implies moving all pixels of the image in one course. It could be horizontally or vertically, and keeps the image dimensions the same. This means that some of the pixels are clipped off the image and new pixel values must be specified in a region of the image. There are two types of shift; horizontal and vertical. Horizontal shift shifts the image to a certain direction along the horizontal axis (left or right). In vertical shift, it shifts the image along the vertical axis (up or down). The arguments for the ImageDataGenerator width_shift_range and height_shift_range regulate the amount of horizontal and vertical shift respectively. These arguments can indicate a floating-point value indicating the proportion (between 0 and 1) of the shifting image of width or height. Alternatively, we can specify the number of pixels to shift the image. In particular, for each image and the shift performed, e.g.[0, value], a value in the range between no shift and the percentage or pixel value will be sampled. Alternatively, we can specify a min and max variety tuple or array from which to sample the change; e.g.:[-100, 100] or[-0.5, 0.5][34].

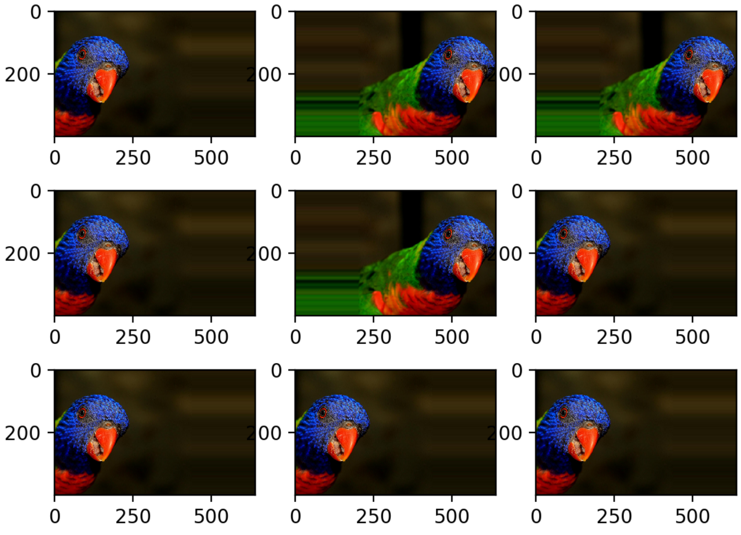


Fig. 5.4. Horizontal shift of an image

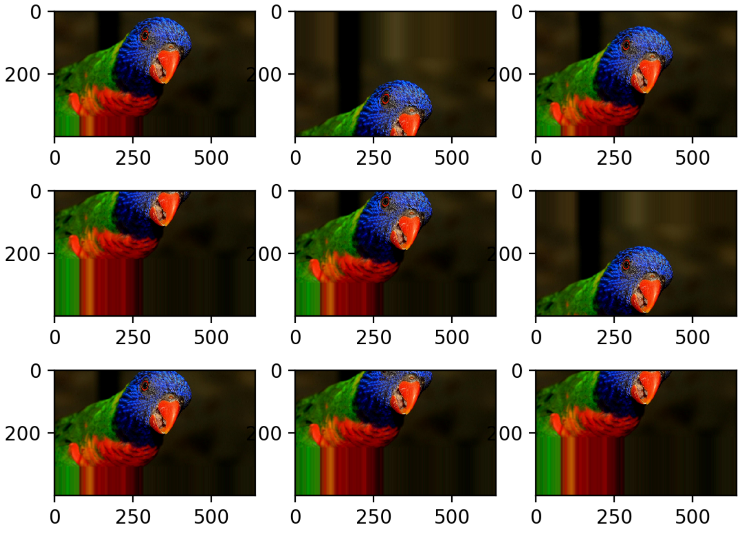


Fig. 5.5. Horizontal shift of an image

Rotation

A rotation augmentation rotates the picture randomly from 0 to 360 degrees in the clockwise direction. Image dimensions after rotation might not be maintained. The rotation will rotate pixels out of the picture outline and take-off zones of the frame with no pixel data that must be filled in. If the image is square, the image size will be preserved by rotating it at right angles. If it is a rectangle, it would maintain the size by rotating it by 180 degrees. The final image size will also be changed by rotating the image with finer angles[34].

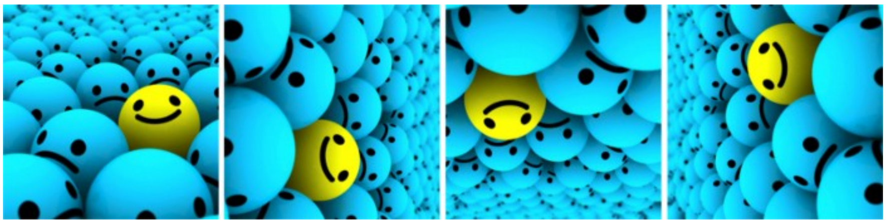


Fig. 5.6. Example of rotation

Flip

Flipping is critical for CNN to evacuate certain features of the object is accessible in as if it were a specific side. An image flip implies switching the lines or columns of pixels within the case of a vertical or horizontal flip respectively. The flip augmentation is defined in the ImageDataGenerator class constructor by a boolean horizontal flip or vertical flip argument.

Horizontal flips may create a sense for pictures such as the bird picture, but vertical flips would not. For other sorts of pictures, such as aerial photos, cosmology photos, and microscopic photos, vertical flips make sense. Flipping numbers is not helpful as they will always have the right-left and right orientation, but this may be helpful for issues with object pictures in a scene with a diverse orientation. Although in natural pictures only horizontal flips are used, we think that vertical flips capture a distinctive property of medical images, i.e. invariance to vertical reflection. Usually, only horizontal flips of the original images are used for natural pictures, as vertical flips often do not represent natural pictures (i.e. an upside-down cat would not usually make a model more discriminatory during training). A vertical flip of a mass, however, would still result in a realistic mass[34].

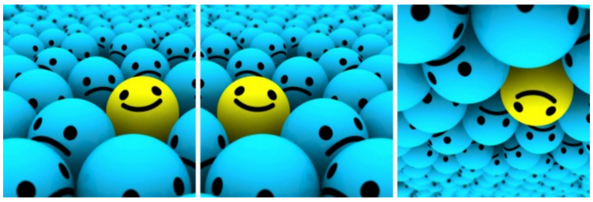


Fig. 5.7. Example of horizontal and upside down flip

Zoom

A zoom augmentation zooms pictures randomly, either adding fresh pixels around the picture or interpolating pixels respectively. For each dimension (width, height) the zoom quantity is sampled evenly randomly independently from the zoom region. Zooming is a strong augmentation capable of making a network robust to (small) object size modifications. The zoom_range parameter controls the zooming factor. The following figure shows the example of the zoomed image, showing a random zoom in which the width and height dimensions differ, and the aspect ratio of the object in the image changes randomly[34].

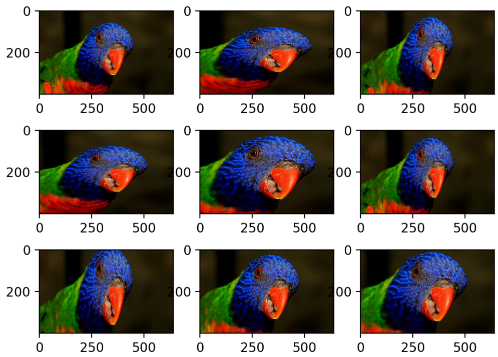


Fig. 5.8. Example of zoom augmentation by different dimentions

Shear

A shear map is a linear map in plane geometry, which displaces each point in a fixed direction, equal to its signed distance from a line parallel to that direction. This sort of mapping is also referred to as shear conversion, transvection, or just shearing. The parameter that regulates the displacement rate is called shear_range and corresponds to the deviation angle (in radians) in the initial picture between a horizontal line and the picture (in the mathematical context) of this line in the transformed image. It is possible to apply a shear transformation as a horizontal shear, a vertical shear, or both.

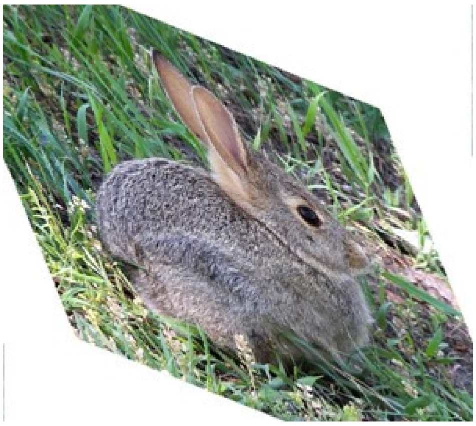


Fig. 5.9. Example of shearing an image

Elastic Transformation

Image distortions are the act of applying filters to a picture to obtain a different picture. Because of the distortion, there will be some kind of shift in the image. Elastic deformation extends and squeezes the image at certain points to simulate uncontrolled hand muscle oscillations. It is mainly performed by calculating each pixel of the image. This technique divides the image into grids and shits the pixel towards a random direction by random strength. here the center of the image is distorted without affecting the size or aspect ratio of the image[35].

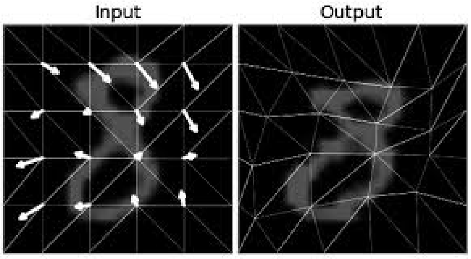


Fig. 5.10. Transforming elastisity pixel wise of an image

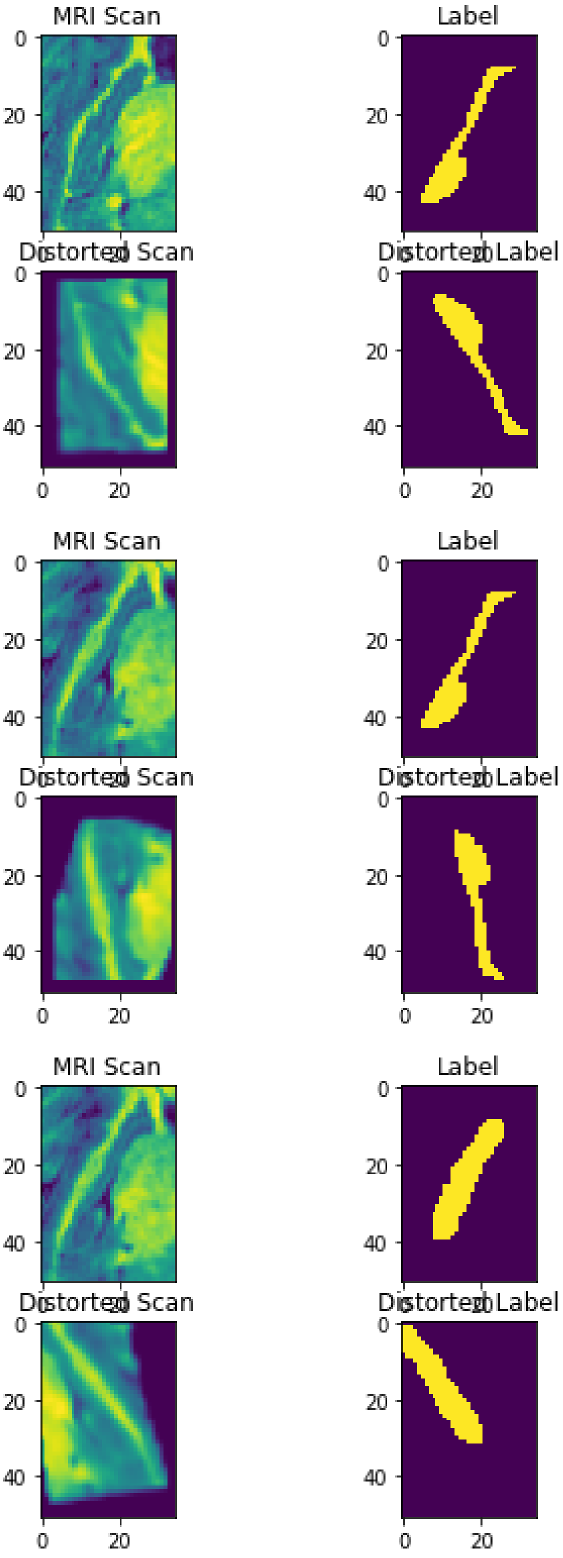


Fig. 5.11. 4-fold augmentation result of MRI scans

Chapter 6

Results & Discussion

6.1 Modified UNET architecture

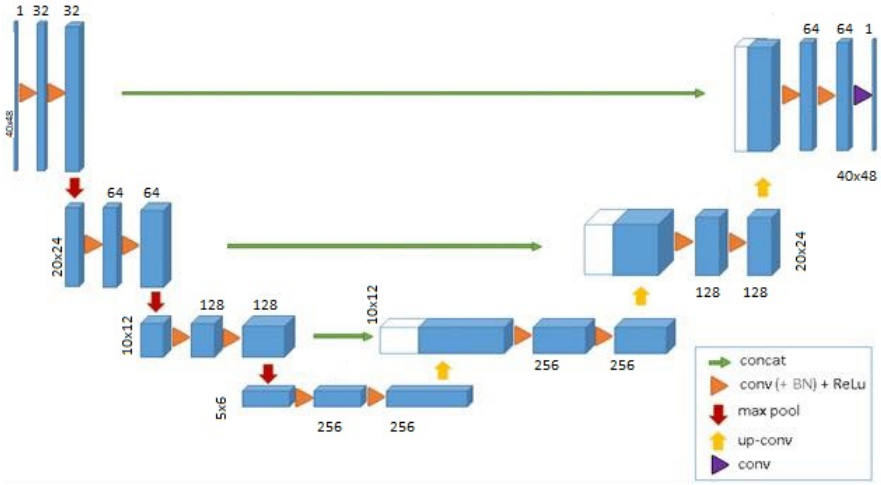


Fig. 6.1. Mini U-net developed for this task

We have developed a "Mini U-net" being inspired from the actual U-net. This mini U-net is helpful for images with lower resolution. Medical image can not be always in higher resolution due to the nature of the scan. Our mini u-net can be great help for that. The key difference between unet and our proposed architecture is in size and scale. Since our target images has a smaller resolution of 40×40, while the original unet architecture was designed for images of size 256×256, it was necessary to scale the size of the network. For this the number of contracting/expanding blocks for mini u-net has reduced to four compared to six contracting/expanding blocks found in the main unet. The filter sizes of convolution and upsampling are also adjusted accordingly. Bottle neck collects all the features all together. It is in between contracting and expanding path.

6.2 Training Setup

For the training the unet 9270 2D samples were generated from the MRI scans. Every image went through 4-fold augmentation after processing that increased the size of sample to 46350 images of resolution 40x48. The train test split ratio is 0.2. Due to the problem of GPU we had to train the data on Google Collaboratory, device was Nvidia Tesla k80. To train the network, algorithm that we have used is named "Adam". Learning Rate is 1 x 10^-5, loss function for this network is negative dice loss. Metrice that we have used is dice co-effiecient. Total number of epochs is 800. We could not train the data more than 800 epochs dew to the limitation of GPU. And finally it took about 8 hours to train the model.

6.3 Training Curves

We have taken two functions to determine whether the model is working swiftly or not. Dice co efficient indicates the similarity between train and test data. Result of training dice co efficient is 69.23 and validation dice co efficient is 67.86. We got the best result for epochs 791.

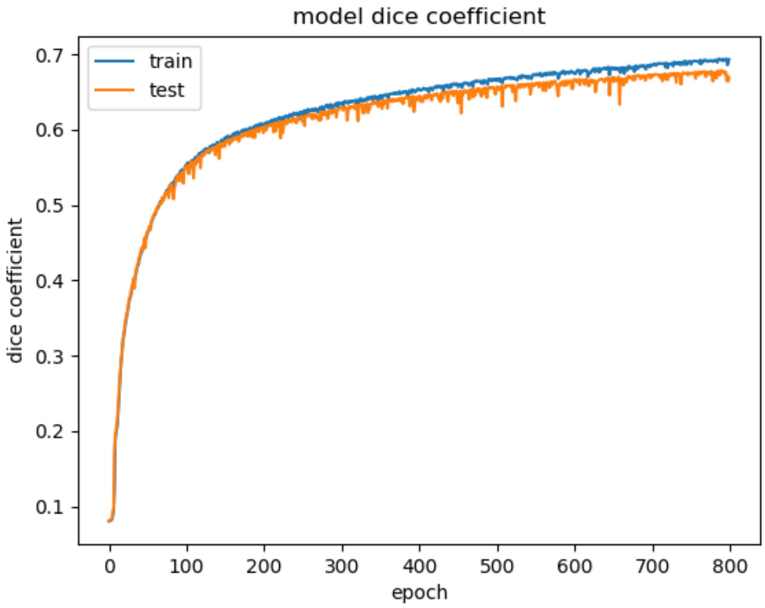


Fig. 6.2. Training Curve of Dice Coefficient

From the graph of dice co efficient and dice loss we can see that the dice co efficient gets higher by time and the dice loss minimizes.

Epochs	Training Dice Coefficient	Validation Dice Coefficient
791	69.23	67.86

Table 6.1. Training Accuracy Metrics

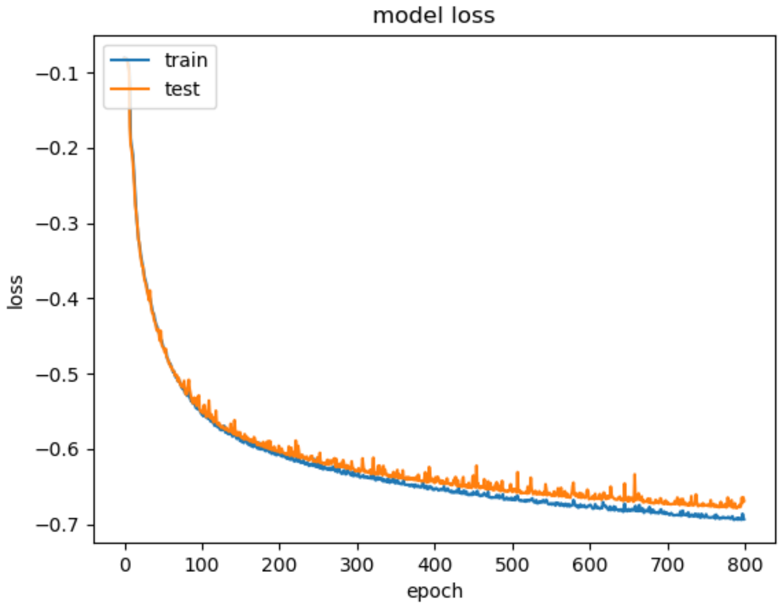


Fig. 6.3. Training Curve of Dice Loss

6.4 Results

After training the model we got the results that shows that the predictions of our model gives pretty much accurate result of hippocampus of the brain. Results are shown into three blocks. First block indicates the main MRI scan from the dataset, second block indicates the true result of hippocampus and third block indicates our models prediction of the result.

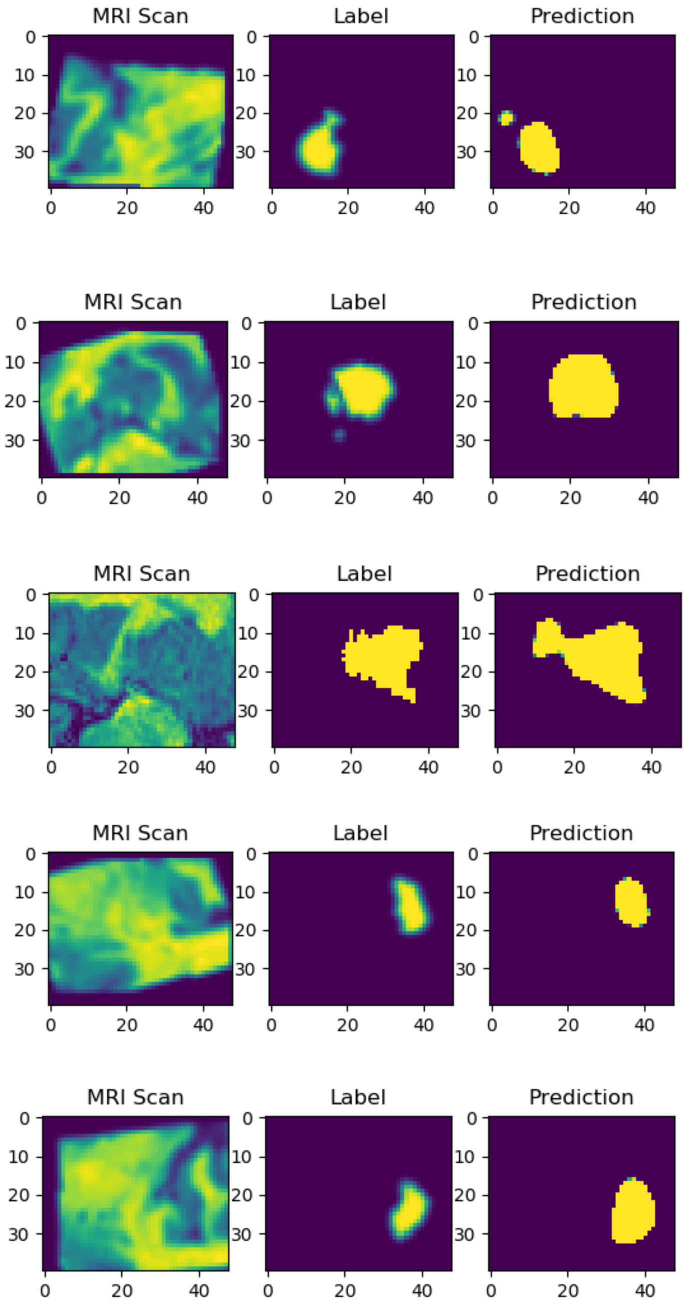


Fig. 6.4. Some examples of results that we obtained. MRI scan indicates the actual scan, label indicates the true result, and prediction block indicates our model's prediction

Chapter 7

Conclusion and Future Scope

We studied several image processing technique for biomedical image data and created a new segmentation model. This model can give a very good result even for small dataset and low resolution images. The more we train the model the more efficient result we can get. Since the dataset consisted only 394 images this problem was eliminated by using augmentation techniques. The model is fast and light weight comparing to others. So it is easy to implement the model. The model is also scalable, so it can be implemented for bigger resolution images. Model can also be used for other biomedical image segmentation data set. For example: brain, liver, lung, kidney etc. We strongly believe that this model could be a great help for radiologist and could save thousands of lives.

Bibliography

[1] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[2] H.-P. Chan, S.-C. B. Lo, B. Sahiner, K. L. Lam, and M. A. Helvie, “Computer-aided detection of mammographic microcalcifications: Pattern recognition with an artificial neural network,” *Medical Physics*, vol. 22, no. 10, pp. 1555–1567, 1995.

[3] S.-C. B. Lo, J.-S. Lin, M. T. Freedman, and S. K. Mun, “Computer-assisted diagnosis of lung nodule detection using artificial convoultion neural network,” in *Medical Imaging 1993: Image Processing*, vol. 1898. International Society for Optics and Photonics, 1993, pp. 859–869.

[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[5] K. Kamnitsas, C. Ledig, V. F. Newcombe, J. P. Simpson, A. D. Kane, D. K. Menon, D. Rueckert, and B. Glocker, “Efficient multi-scale 3d cnn with fully connected crf for accurate brain lesion segmentation,” *Medical image analysis*, vol. 36, pp. 61–78, 2017.

[6] H.-C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers, “Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning,” *IEEE transactions on medical imaging*, vol. 35, no. 5, pp. 1285–1298, 2016.

[7] H. R. Roth, L. Lu, A. Farag, A. Sohn, and R. M. Summers, “Spatial aggregation of holistically-nested networks for automated pancreas segmentation,” in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2016, pp. 451–459.

[8] Y. Zhou, L. Xie, W. Shen, Y. Wang, E. K. Fishman, and A. L. Yuille, “A fixed-point model for pancreas segmentation in abdominal ct scans,” in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2017, pp. 693–701.

[9] Z. Yan, Y. Zhan, Z. Peng, S. Liao, Y. Shinagawa, S. Zhang, D. N. Metaxas, and X. S. Zhou, "Multi-instance deep learning: Discover discriminative local anatomies for bodypart recognition," *IEEE transactions on medical imaging*, vol. 35, no. 5, pp. 1332–1343, 2016.

[10] N. Tajbakhsh, J. Y. Shin, S. R. Gurudu, R. T. Hurst, C. B. Kendall, M. B. Gotway, and J. Liang, "Convolutional neural networks for medical image analysis: Full training or fine tuning?" *IEEE transactions on medical imaging*, vol. 35, no. 5, pp. 1299–1312, 2016.

[11] H. R. Roth, L. Lu, J. Liu, J. Yao, A. Seff, K. Cherry, L. Kim, and R. M. Summers, "Improving computer-aided detection using convolutional neural networks and random view aggregation," *IEEE transactions on medical imaging*, vol. 35, no. 5, pp. 1170–1181, 2015.

[12] A. A. A. Setio, F. Ciompi, G. Litjens, P. Gerke, C. Jacobs, S. J. Van Riel, M. M. W. Wille, M. Naqibullah, C. I. Sánchez, and B. van Ginneken, "Pulmonary nodule detection in ct images: false positive reduction using multi-view convolutional networks," *IEEE transactions on medical imaging*, vol. 35, no. 5, pp. 1160–1169, 2016.

[13] M. R. Al Asif, S. Roy, A. Abdullah, M. Raihan, R. Akter, and M. Z. Hossain, "Role and impact of biomedical engineering discipline for developing country perspective," 2018.

[14] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *2016 Fourth International Conference on 3D Vision (3DV)*. IEEE, 2016, pp. 565–571.

[15] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, "3d u-net: learning dense volumetric segmentation from sparse annotation," in *International conference on medical image computing and computer-assisted intervention*. Springer, 2016, pp. 424–432.

[16] X. Zhou, T. Ito, R. Takayama, S. Wang, T. Hara, and H. Fujita, "Three-dimensional ct image segmentation by combining 2d fully convolutional network with 3d majority voting," in *Deep Learning and Data Labeling for Medical Applications*. Springer, 2016, pp. 111–120.

[17] K. H. Hwang, H. Lee, and D. Choi, "Medical image retrieval: past and present," *Healthcare informatics research*, vol. 18, no. 1, pp. 3–9, 2012.

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

[19] A. P. Twinanda, S. Shehata, D. Mutter, J. Marescaux, M. De Mathelin, and N. Padoy, "Endonet: a deep architecture for recognition tasks on laparoscopic videos," *IEEE transactions on medical imaging*, vol. 36, no. 1, pp. 86–97, 2016.

[20] S. M. Anwar, S. Yousaf, and M. Majid, "Brain tumor segmentation on multimodal mri scans using emap algorithm," in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2018, pp. 550–553.

[21] S. Van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart, and T. Yu, "scikit-image: image processing in python," *PeerJ*, vol. 2, p. e453, 2014.

[22] B. Demirtaş *et al.*, "Atatürk döneminde eğitim alanında yaşanan gelişmeler," *Gazi Akademik Bakış*, no. 02, pp. 155–176, 2008.

[23] C. K. Ingold *et al.*, *Structure and mechanism in organic chemistry*. Cornell University Press Ithaca, NY, 1969, vol. 1.

[24] G. A. Baxes, *Digital image processing: principles and applications*. Wiley New York, 1994.

[25] S. Hijazi, R. Kumar, and C. Rowen, "Using convolutional neural networks for image recognition," *Cadence Design Systems Inc.: San Jose, CA, USA*, 2015.

[26] Y. Ma, D. Xiang, S. Zheng, D. Tian, and X. Liu, "Moving deep learning into web browser: How far can we go?" in *The World Wide Web Conference*. ACM, 2019, pp. 1234–1244.

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

[28] S. Van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart, and T. Yu, "scikit-image: image processing in python," *PeerJ*, vol. 2, p. e453, 2014.

[29] S. M. Anwar, M. Majid, A. Qayyum, M. Awais, M. Alnowami, and M. K. Khan, "Medical image analysis using convolutional neural networks: a review," *Journal of medical systems*, vol. 42, no. 11, p. 226, 2018.

[30] M. Sezgin and B. Sankur, "Survey over image thresholding techniques and quantitative performance evaluation," *Journal of Electronic imaging*, vol. 13, no. 1, pp. 146–166, 2004.

[31] J. Brugués, J. Gomis, and K. Kamimura, “Newton-hooke algebras, nonrelativistic branes, and generalized pp-wave metrics,” *Physical Review D*, vol. 73, no. 8, p. 085011, 2006.

[32] M. P. Pound, J. A. Atkinson, D. M. Wells, T. P. Pridmore, and A. P. French, “Deep learning for multi-task plant phenotyping,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2055–2063.

[33] W. P. Wolfensberger, B. Nirje, S. Olshansky, R. Perske, and P. Roos, “The principle of normalization in human services,” 1972.

[34] F. Chollet, “Building powerful image classification models using very little data,” *Keras Blog*, 2016.

[35] P. Y. Simard, D. Steinkraus, J. C. Platt *et al.*, “Best practices for convolutional neural networks applied to visual document analysis.” in *Icdar*, vol. 3, no. 2003, 2003.