A COMPARATIVE STUDY OF DEEP LEARNING ALGORITHMS ON SKIN LESION SEGMENTATION

(DERİ LEZYONU SEGMENTASYONU ÜZERİNE DERİN ÖĞRENME ALGORİTMALARININ KARŞILAŞTIRMALI BİR ÇALIŞMASI)

by

FATİH ERGİN, B.S.

Thesis

Submitted in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE

in

COMPUTER ENGINEERING

in the

GRADUATE SCHOOL OF SCIENCE AND ENGINEERING

of

GALATASARAY UNIVERSITY

JUNE 2020

Approval of the thesis:

A COMPARATIVE STUDY OF DEEP LEARNING ALGORITHMS ON SKIN LESION SEGMENTATION

submitted by **FATİH ERGİN** in partial fulfillment of the requirements for the degree of **Master of Science in Computer Engineering Department, Galatasaray University** by,

Examining Committee Members:

Assist. Prof. Dr. İ. Burak Parlak
Supervisor, **Computer Engineering Department, GSU** ______________

Assist. Prof. Dr. JAMES DOE
Industrial Engineering Department, GSU ______________

Assist. Prof. Dr. JESSICA DOE
Industrial Engineering Department, University of Kentucky ______________

Date: ______________

ACKNOWLEDGMENTS

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Skin lesions are a severe disease globally. As with most diseases, it dramatically increase the survival rate that early detection of lesions on dermoscopy images (?). Skin diseases are difficult to recognize because of the similarity between lesions and low contrast between lesions and skin. Because of the increased ability of machine learning techniques to transform input data into high-level presentations, great emphasis has been placed on deep learning techniques used for image analysis in the last few years. For the accurate diagnosis, the medical field has an increasing interest in this technology, especially in the diagnosis of skin lesions. This paper presents a comparision between various state of the art deep learning based approaches namely Capsule Network, Generative Adversial Networks and U-Net: Convolutional Networks for Biomedical Image Segmentation to solve skin lesion analysis problems using a dermatoscopic image that contains a skin tumor. Used models are trained and evaluated on standard comparison datasets from the International Skin Imaging Collaboration (ISIC) 2017 Challenge (?).

Keywords : SKIN LESION, SEGMENTATION, DEEP LEARNING, UNET, CAPSULE NETWORK, GENERATIVE ADVERSIAL NETWORK

RÉSUMÉ

Les lésions cutanées sont une maladie grave à l'échelle mondiale. Comme avec la plupart des maladies, il augmente considérablement le taux de survie que la détection précoce des lésions sur les images dermoscopiques (**?**). Les maladies de la peau sont difficiles à reconnaître en raison de la similitude entre les lésions et du faible contraste entre les lésions et la peau. En raison de la capacité accrue des techniques d'apprentissage automatique à transformer les données d'entrée en présentations de haut niveau, une grande importance a été accordée aux techniques d'apprentissage approfondi utilisées pour l'analyse d'images au cours des dernières années. Pour le diagnostic précis, le domaine médical s'intéresse de plus en plus à cette technologie, notamment au diagnostic des lésions cutanées. Cet article présente une comparaison entre diverses approches basées sur l'apprentissage en profondeur à la pointe de la technologie, à savoir le réseau de capsules, les réseaux adverses génératifs et U-Net: réseaux convolutionnels pour la segmentation d'images biomédicales pour résoudre les problèmes d'analyse des lésions cutanées à l'aide d'une image dermatoscopique contenant une tumeur cutanée. Les modèles utilisés sont entraînés et évalués sur des ensembles de données de comparaison standard du Challenge International Collaboration Imagerie Skin (ISIC) 2017 (**?**).

Mots Clés : LÉSION CUTANÉE, SEGMENTATION, APPRENTISSAGE PROFOND, UNET, RÉSEAU CAPSULE, RÉSEAU ADVERSARIAL GÉNÉRATEUR

ÖZET

Deri lezyonları dünya çapında ciddi bir hastalıktır. Çoğu hastalıkta olduğu gibi, dermoskopi görüntülerinde lezyonların erken saptanması, sağkalım oranını önemli ölçüde artırır (?). Lezyonlar ve cilt arasındaki düşük kontrast, çeşitli lezyonlar arasındaki benzerlikler hastalığın tanınmasını zorlaştırır. Son birkaç yılda, makine öğrenmesi yöntemlerinin veri dönüştürmedeki yeteneklerinin artmasına bağlı olarak resim analizinde derin öğrenme yöntemlerine büyük önem verilmiştir. Verilerin yararlı kullanımını artırmak; doğru teşhis yapabilmek adına, diğer tıbbi alanlarda olduğu gibi deri lezyonu teşhisinde de derin öğrenmeye olan ilgi artmıştır. Bu çalışma, bir cilt tümörü içeren dermatoskopik bir görüntü kullanarak cilt lezyonu analiz problemlerini çözmek için Kapsül Ağları, GAN ve U-Net: Biyomedikal Görüntü Segmentasyonu için Konvolüsyonel Ağlar gibi son teknoloji derin öğrenme temelli yaklaşımlar arasında bir karşılaştırma sunmaktadır. Kullanılan modeller, International Skin Imaging Collaboration (ISIC) 2017 Challenge'daki (?) standart karşılaştırma veri kümelerinde eğitilir ve değerlendirilir.

Anahtar Kelimeler : CİLT LEZYONU, SEGMNENTASYON, DERİN ÖĞRENME, UNET, KAPSÜL AĞLARI, GAN

1 INTRODUCTION

1.1 Motivation

1.2 Work Plan

2 BACKGROUND

2.1 Medical Image Segmentation

2.2 Artificial Neural Networks

2.3 Convolutional Neural Networks

2.4 Image Segmentation Architectures

2.5 Related Works

3 METHODOLOGY

3.1 Networks

3.1.1 U-Net

3.1.2 SegAN

3.1.3 SegCaps

3.2 Dataset

The first part of ISBI Challenge 2017 (?) - Skin Lesion Analysis Towards Melanoma Detection : Lesion Segmentation dataset is used in this project. This dataset has train, validation and test data separately. The training dataset consist of 2000 dermoscopic .jpg images and the related masks with .png format. The dataset include various type of lesions namely malignant melanoma, nevus and seborrhoeic keratosis .
There are also validation and test datasets which contain 150 and 600 images respectively which is provided by the organizers. The results are based on several common image similarity metrics which are given related section.

The images are of various dimensions and the all used neural networks can't handle relatively big images because of their different internal architectures and memory

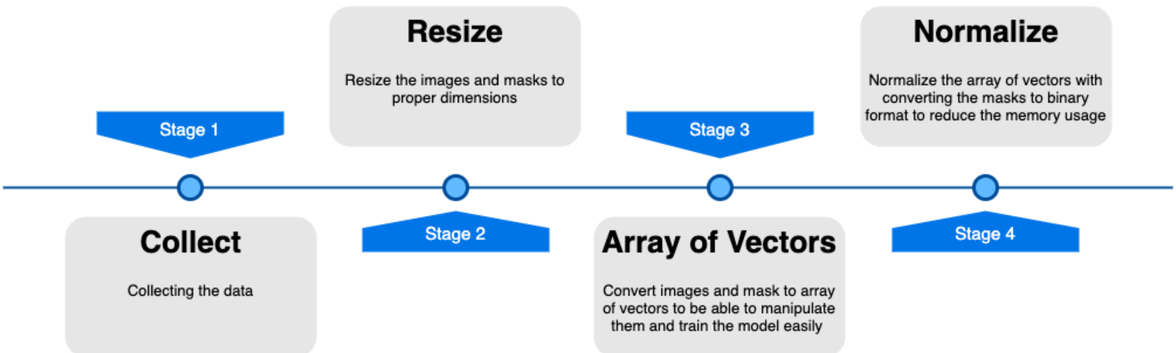

Figure 3.1: Data preparation process

constraints. We also had to resize all images into same dimension to reduce the memory consumption and increase the accuracy as a preprocessing stage. As it can be stated at Figure 3.1, arrays of mask files converted to uint8 to reduce the size of the masks.

3.3 Experiments

3.3.1 Experiment 1 : ...

3.3.2 Experiment 2 : ...

3.3.3 Experiment 3 : ...

3.4 Evaluation

Several evaluation metrics were used to determine the quality of the models. Dice coefficient, Jaccard index, Accuracy, Sensitivity and Specificity were used to compare the target and predicted segmentation mask. The true positives (TP) determine pixels (or voxels) correctly classified as being part of the segmentation, a false positive (FP) is a pixel incorrectly classified as being part of the segmentation, and a false negative (FN) is a pixel which should have been part of the segmentation but was not.

3.4.1 Metrics

3.4.1.1 Dice coefficient

Dice coefficient is computed by comparing the pixel-wise agreement between the ground-truth and its corresponding predicted segmentation. Specially, this metric is just used to evaluation the segmentation model performance.

$$Dice(A, B) = \frac{2|A \cdot B|}{|A| + |B|} \tag{3.1}$$

3.4.1.2 Jaccard index

Jaccard index, also known as the Jaccard similarity coefficient, compares predictions (A) with the groundtruth (B) to see which samples are shared and which are distinct. The higher the index, the more similar the two subsets.

$$Jaccard(A,B) = \frac{|A \cdot B|}{|max(A,B)|} = \frac{|A \cdot B|}{|A| + |B| - |A \cdot B|} \quad (3.2)$$

3.4.1.3 Accuracy

Accuracy measures the proportion of true positives and true negatives whose are correctly segmented instances to the total number of instances. It is derived from sensitivity and specificity which are given below.

$$Accuracy(A,B) = \frac{|A \cdot B| + |\overline{A} \cdot \overline{B}|}{|All|} \quad (3.3)$$

3.4.1.4 Sensitivity

Sensitivity aims to measure correctly segmented instance ratio.

$$Sensitivity = \frac{TP}{TP + FN} \quad (3.4)$$

3.4.1.5 Specificity

Sensitivity measures incorrectly segmented instance ratio.

$$Specificity = \frac{TN}{TN + FP} \quad (3.5)$$

3.5 Tools and Frameworks

This section presents tools used for development and testing during the project. Python is selected as the main programming language for this project.

3.5.1 TensorFlow

TensorFlow is an open source library for performing numerical computations. Although it can be used for computations in general, it is most commonly used as a tool for machine learning research. TensorFlow can be interfaced using Python and is then translated to a computational graph. The computational graph can be fed with the tensors by launching a TensorFlow session which are generalization of N-dimensional arrays. The graph performs a series of mathematical operations on the data. Weight matrices and biases are trainable variables in the TensorFlow graph during a session. Loss functions and optimization algorithms for backpropagation exist in TensorFlow (**?**). That makes training a model is as simple as specifying an objective function to optimize for, as well as running the optimizer with a batch of data inside a session.

3.5.2 Keras

Keras is a neural networks API for Python (**?**). It runs on top of TensorFlow or Theano (**?**) which is used as the main neural network framework. Keras is user-friendly and allows for complex models to be created with relatively few lines of code. Keras consists of many commonly used building blocks of neural networks. These are parts as layers, objectives, activation functions and optimizers. The components include parts for convolutional and recurrent neural networks as convolutions, pooling, dropout and batch normalization.

3.5.3 PyTorch

Pytorch is a machine learning framework developed by Facebook which has relatively advantages over TensorFlow in terms of simplicity and usability. It implements dynamic computational graphs which makes dynamic changes on the networks possible with a little effort. Debugging is relatively easy with Pytorch.

3.5.4 Numpy

NumPy (Numerical Python) is a scientific computing library for the Python that allows us to perform scientific calculations quickly (**?**). Numpy arrays form the basis of Numpy. Numpy arrays are similar to python lists, but are more useful in terms of speed and functionality than python lists.

3.5.5 Scipy

SciPy is a package for scientific computing which includes functionality several clustering algorithms, Fourier transforms, linear algebra, interpolation, regression, image and signal processing for the Python programming language (**?**).

3.5.6 Python Imaging Library (PIL)

The Python Imaging Library (PIL) is a free Python library which supports several widely-used image manipulation procedures like per-pixed manipulating, image filtering, image enhancing, masking etc (**?**).

3.5.7 Jupyter Notebook

Jupyter Notebook is an open source web application that allows editing and running code which can be used with over 40 different programming languages (**?**). It is a Json based document that has ordered cells which can be live code, equations, visualizations or narrative text.

3.5.8 Google Colab

A Google product Colaboratory which is also known as Google Colab that requires no setup and runs entirely in the cloud is used in this paper. It is a free Jupyter Notebook environment that aims to encourage Machine Learning and Artificial Intelligence research where often the barrier to learning and success is the requirement of tremendous computational power. It is possible to develop deep learning applications with Google Colaboratory on the free Tesla K80 GPU using common Deep Learning frameworks and tools like Keras, TensorFlow and PyTorch. Google Colab runs on a connected Google Drive accounts. It also allows the to the users to use and share Jupyter Notebooks with others without having to download, install, or run anything on their own computer other than a browser. All models were trained and tested using Tesla K80 GPU which has 25 GB of video memory on a Ubuntu 18.04.

4 RESULTS

5 DISCUSSION AND FUTURE WORKS

6 CONCLUSION

APPENDIX A PROOF OF SOME THEOREM

BIOGRAPHICAL SKETCH

Write your curriculum vitae here.

PUBLICATIONS

— If you have publications you must write there.