

04.06.2018

**Skin Disease Prediction Using Deep Neural Network**

Ahmet Görünücü, Emre Dindar, Zafer Emre Ocak

SUPERVISED BY ASSOS. PROF. DR. ALI FUAT ALKAYA

Contents

[1. Introduction 2](#_Toc515909953)

[1.1. Problem Description and Motivation 2](#_Toc515909954)

[1.2 Scope of the Project 3](#_Toc515909955)

[1.3 Definitions, Acronyms and Abbreviations 3](#_Toc515909956)

[2. Literature Survey 4](#_Toc515909957)

[3. Project Requirements 5](#_Toc515909958)

[3.1. Functional Requirements 5](#_Toc515909959)

[3.1.1 Authentication Screen 5](#_Toc515909960)

[3.1.1.1 E-mail 5](#_Toc515909961)

[3.1.1.2 Password 5](#_Toc515909962)

[3.1.1.3 SignUp Button 5](#_Toc515909963)

[3.1.1.4 LogIn Button 5](#_Toc515909964)

[3.1.2 Upload Screen 5](#_Toc515909965)

[3.1.2.1 Take Photo Button 5](#_Toc515909966)

[3.1.2.2 Upload Button 5](#_Toc515909967)

[3.1.3 Result Screen 5](#_Toc515909968)

[3.1.4 Database Side 5](#_Toc515909969)

[3.2 Nonfunctional Requirements 6](#_Toc515909970)

[4. System Design 7](#_Toc515909971)

[4.1 Use Case Diagram 7](#_Toc515909972)

[4.2 Database ER Diagram 8](#_Toc515909973)

[4.3 Test Plan 9](#_Toc515909974)

[5. Software Architecture 11](#_Toc515909975)

[5.1 Data Flow 11](#_Toc515909976)

[5.2 Control Flow 12](#_Toc515909977)

[6. Tasks Accomplished 13](#_Toc515909978)

[6.1 Current state of the project 13](#_Toc515909979)

[6.2 Task Log 13](#_Toc515909980)

[6.3 Task Plan With Milestones 14](#_Toc515909981)

[7. References 15](#_Toc515909982)

1. Introduction
   1. Problem Description and Motivation

Due to today's changing environmental conditions, the spectrum of dermatological diseases has increased greatly. A disease that is defined as redness in the past, can be defined in many ways such as cancer, measles, flower disease, melanoma in today's conditions. As a result of this diversity of diseases, the process of diagnosis is becoming increasingly difficult. The problem of solving this by raising number of doctors is a costly and time consuming process.

Early diagnosis plays a very important role in the treatment process. The late diagnosis of deadly diseases such as cancer can completely invalidate the treatment. The important factor which makes our project important, is to shorten the diagnosis process significantly. Thus, the possibility of treating many diseases with early diagnosis will increase considerably.

Our alternative solution is to create an information repository from the past diagnoses of experienced doctors, and it is an application that supports less-experienced doctors in the decision-making stage with the help of knowledge of diseases that are difficult to diagnose. In addition, another benefit of this continuously growing pool of information is; the decisions to be made are more up-to-date and correct.

The reason for us to make this project is to help our doctors to make decisions, to know that we will touch every life that will be saved, and to fulfill our duty as software developers.

1.2 Scope of the Project

Our project is aimed at creating a preliminary diagnosis of dermatological diseases in the human body by using image processing methods of photographs taken with the help of phone camera. To achieve this goal, we are going to use Convolution Neural Networks, then alter weights in convolution layers in the direction of our purpose, and finally we will try to reach 72% of the highest accuracy level[1] currently reached. In case of success, we would like to obtain a higher level of accuracy than current accuracy. We will design a user interface so that the result of the calculations that will be made by the network we create, will be clear to the doctor. We foresee our project phases as follows:

1. Obtaining the data
2. Implementation and training of Neural Network
3. Making Convolutional Neural Network structure suitable (adjusting weights)
4. Creating an Android application for easier understanding of results

## 1.3 Definitions, Acronyms and Abbreviations

|  |  |
| --- | --- |
|  | Definitions, Acronyms and Abbreviations |
| NN | Neural Network |
| ANN | Artificial Neural Network |
| CNN | Convolutional Neural Network |
| API | Application Program Interface |
| BOF | Bag of Features |
| ISIC | International Skin Imaging Collaboration |
| ILSVRC | Image Large Scale Visual Recognition Challenge |

## **Literature Survey**

Along with the increase in the processing power of the machines and the data sets available, companies have begun to work on this problem in recent years. However, since it has an aspect of business dimension, it is not available for us to reach the contents. Some of the work done in the same field with us is as follows:

1. **Codella:** They are trying to diagnose Melanoma, the deadliest kind of skin cancer, by taking advantage of image processing. They make inferences using a pre-trained NN. They use the ILSVRC 2012 as the dataset. The dataset contains 900 training images and 379 test images. They keep working on ImageNet dataset which we will use later. They state that they have achieved an accuracy rate of 70.5% in their last published articles. [2]
2. **Barata:** The approach proposed by Barata et al. [3] utilizes two different methods for the detection of melanoma in dermoscopy images based on global and local features. The global method uses segmentation and wavelets, Laplacian pyramids or linear filters followed by a gradient histogram are used to extract features such as texture, shape, and colour from the entire lesion. After that, a binary classifier is trained from the data. The second method of local features uses a BOF classifier for image processing tasks (i.e. object recognition).
3. **Kawahara:** A previously trained convolution neural network was used. They have practiced on ISIC 2017 dataset. There are 2000 labelled images on 10 dermatological diseases in this dataset. It also contains 150 test images. Likewise, we are going to do as well, they carried out performance tests on AlexNet, VGG16, ResNet50, and ResNet101 frameworks. [4]

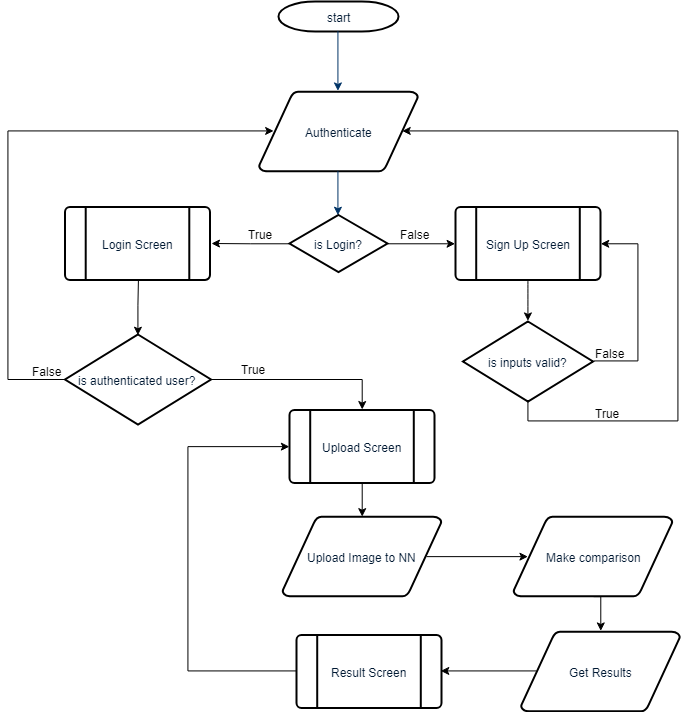
# 

# **Project Requirements**

# Functional Requirements

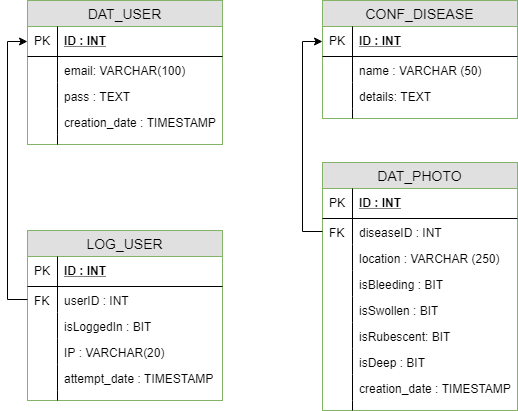
* + 1. Authentication Screen**:** There need to be a screen that will allow the system to create records for new users. On this screen, the following information is taken from the user.
       1. E-mail: Username must be the user's e-mail address.
       2. Password: Password information must consist of at least 8 characters and must consist of the one number, one upper case letter and one lower case letter.
       3. SignUp Button: When the button is clicked, the information entered should be sent to the server.
       4. LogIn Button: If the entered information matches the information we have on our server, the user logs in. Otherwise, the warning text appears on the screen.
    2. Upload Screen**:** Users who are logged in must encounter this screen. On this screen, users upload the following information to the system. The system informs user about the skin disease.
       1. Take Photo Button: When the button is clicked, the camera of the phone must be opened. So, the user takes a picture of the diseased area.
       2. Upload Button: If there is still no image taken, the button is not active. After the photo is taken, clicking on the button will upload the photo to the system.
    3. Result Screen: The system must compare the captured image to the trained NN. As a result of the comparison made, top 5 skin diseases with the highest similarity and their probabilities must be reported to the user.
    4. Database Side
       1. User information must be held.
       2. Authentication processes must be controlled.
       3. Every attempt must be logged.
  1. Nonfunctional Requirements
     1. Usability
        1. User can register to the system only once with the same e-mail.
        2. The password field in the database must be encrypted with various methods.
        3. NN must be trained when the application is first installed.
     2. Efficiency(Performance)
        1. An efficient search will be performed because the database will be indexed to the user ID’s
        2. An efficient find will be performed because the Neural Network will be on local system.
     3. Portability
        1. The system will support Android platforms.
        2. User’s android systems must be more than API 26 (Android 8.0)
     4. Privacy Requirements
        1. Verification of the data will be provided by hash functions.
     5. Security Requirements
        1. 3-tier system architecture will ensure the security of the database against attack from the outside.
        2. As a result of an attack from outside, regular backups will be done on the server side against deletion of information in the system.

1. System Design
   1. Use Case Diagram



* 1. Database ER Diagram

We will use database to hold user information and dataset information. All user login attempts will be logged in the LOG\_USER table. Also, dataset information will be held on the database, because it will has some extra attributes like isBleeding, isSwollen etc.



4.3 Test Plan

This section contains a list of tests we will perform and the results we expect to achieve. The tests relate to the general use of the system.

|  |  |
| --- | --- |
|  | Test Case 1 |
| Test Name : | User Registering. |
| Procedure : | It will be tried to create a user who does not exist in the database. |
| Expected Result: | SUCCESS! User is registered. |

|  |  |
| --- | --- |
|  | Test Case 2 |
| Test Name : | User creation with same E-mail. |
| Procedure : | We will try to create a user with an e-mail same as in the database. |
| Expected Result: | ERROR! User has already registered. |

|  |  |
| --- | --- |
|  | Test Case 3 |
| Test Name : | User Login |
| Procedure : | Trying to login with a user e-mail existing in the database |
| Expected Result: | SUCCESS! Login Successful |

|  |  |
| --- | --- |
|  | Test Case 4 |
| Test Name : | Unregistered User |
| Procedure : | Trying to login with a user e-mail that does not exist in the database. |
| Expected Result: | ERROR! Login Unsuccessful |

|  |  |
| --- | --- |
|  | Test Case 5 |
| Test Name : | Password Features |
| Procedure : | Setting a password with at least one uppercase, lowercase, and number. |
| Expected Result: | SUCCESS! |
|  |  |

|  |  |
| --- | --- |
|  | Test Case 6 |
| Test Name : | Incorrect Password |
| Procedure : | Entering at least one input that does not meet the specified criteria. |
| Expected Result: | ERROR! |

|  |  |
| --- | --- |
|  | Test Case 7 |
| Test Name : | Taking Photo |
| Procedure : | Opening the camera by pressing the take photo button and taking the photo successfully |
| Expected Result: | SUCCESS! Getting high quality photo. |

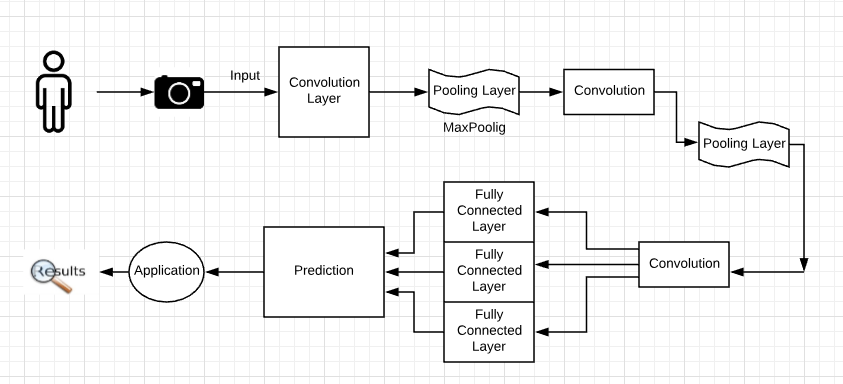
|  |  |
| --- | --- |
|  | Test Case 8 |
| Test Name : | Uploading Image |
| Procedure : | Upload image to the system by pressing the upload image button |
| Expected Result: | SUCCESS! Image uploaded successfully |

|  |  |
| --- | --- |
|  | Test Case 9 |
| Test Name : | Upload without Image |
| Procedure : | Click upload button without taking photo. |
| Expected Result: | ERROR! |

|  |  |
| --- | --- |
|  | Test Case 10 |
| Test Name : | Evaluate the results |
| Procedure : | Evaluation of the results of the program by a expert doctor. |
| Expected Result: | SUCCESS! |

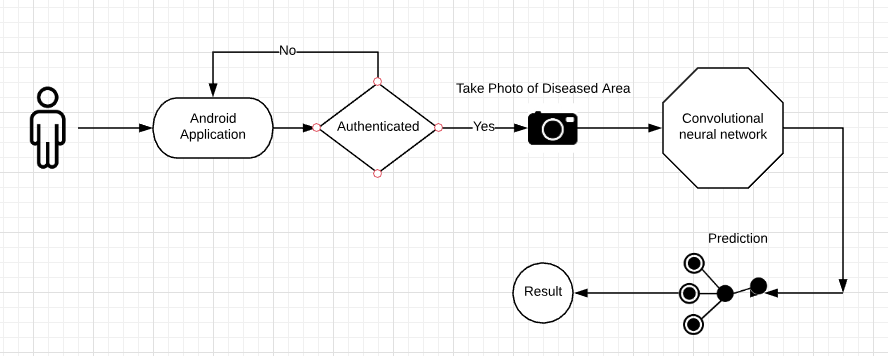
1. Software Architecture
   1. Data Flow

The core component of our system is artificial neural network. The purpose of artificial neural networks is to develop new models in order to make computers faster than human mind in a variety of ways, to develop new models and to use the processing power efficiently. At this step, we are using Convolution Neural Network. This system consists of multiple convolution layers. Each layer in the system includes an abstraction of the input data at a higher level. The output of one layer is the input of another layer associated with it. Up to 1000 convolutions with a minimum 5 can be found between layers. Inputs to the layers are calculated by the activation functions and transferred to the other layer.



# **Control Flow**

The user must first log in to the android application. After the user has passed the authentication step, he will take the photo of the diseased area and upload it to the application. The photo will be input for convolutional neural network. There will be some calculations on this input image. As a result of these calculations, the disease will be estimated and the results will be shown clearly to the user.



1. Tasks Accomplished
   1. Current state of the project

* Neural network structure was researched and learned.
* Convulational neural network structure which will be used in the project was learned.
* The learning algorithm and libraries to be used were decided.
* The data sets which will be used to train the NN were collected.
* Other studies which similar with our project have been reviewed.
* The main screens for user entries are designed.
* We are still learning Python because we have not worked with the Python language before.
  1. Task Log
* Problem Description and Solution Suggestion

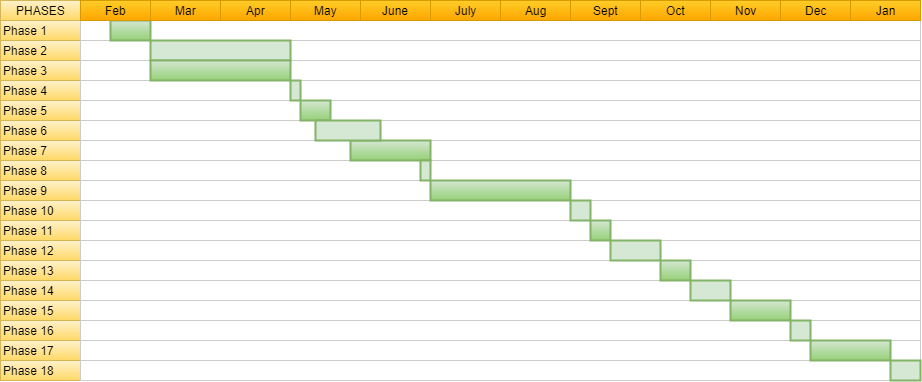
15.02.2018 – 25.02.2018 - 10 Hours Total

* Project Description and Determining Phases  
   25.02.2018 – 07.03.2018 - 14 Hours Total
* General Architecture Design Discussion  
   07.03.2018 – 19.03.2018 - 17 Hours Total
* Determining Technologies   
   19.03.2018 – 22.03.2018 - 6 Hours Total
* Determining Algorithms and Dataset to be used

22.03.2018 – 01.04.2018 - 12 Hours Total

* Collecting Datesets   
   01.04.2018 – 08.04.2018 - 7 Hours Total
* Obtaining Additioanal Dataset   
   08.04.2018 – 16.04.2018 - 5 Hours Total
* Starting to Learn Python Language   
   29.04.2018 – 20.05.2018 - 35 Hours Total

* 1. Task Plan With Milestones
* Milestone **1**: Preparation and Research
  + Phase **1**: Getting general information about Deep Learning and Neural Network.
  + Phase **2**: Taking courses on Stanford University about DL and NN.
  + Phase **3**: Getting deep information from writings.
  + Phase **4**: Creating new code standards for implementation.
  + Phase **5**: Learning Python programming language
* Milestone **2 :** Analyzing Libraries
  + Phase **6**: Analyzing and making decision about libraries for DL (Tensorflow, Keras, Theano etc.)
  + Phase **7**: Examining examples for the library.
* Milestone **3 :** Collecting Data
  + Phase **8**: Downloading Datasets
* Milestone **4 :** Implementation
  + Phase **9**: Implementing the python source code
  + Phase **10**: Creating Test Cases and calculating top-five accuracy
  + Phase **11**: Identifying sensitivity and specificity
  + Phase **12**: Making decision about the Process Model
  + Phase **13**: Creating Android Application Design
* Milestone **5 :** Integration
  + Phase **14**: Research about integration of Android and Python
  + Phase **15**: Implementing Android Application
* Milestone **6 :** Testing
  + Phase **16**: Application Unit Test
  + Phase **17**: Integration of Python Code and General System Testing
* Milestone **7 :** Closure
  + Phase **18**: Preparing Application for distribution



*TIMELINE*

## **References**

[1] Y.-D. Kim, E. Park, S. Yoo, T. Choi, L. Yang, and D. Shin, “Compression of Deep Convolutional Neural Networks for Fast and Low Power Mobile Applications,” in ICLR, 2016.

[2] N. Codella, Q.-B. Nguyen, S. Pankanti, D. Gutman, B. Helba, A. Halpern, and J. R. Smith, “Deep learning ensembles for melanoma recognition in dermoscopy images”, 2016.

[3] C. Barata, M. Ruela, M. Francisco, T. Mendonc¸a, and J. S. Marques, “Two systems for the detection of melanomas in dermoscopy images using texture and color features,” IEEE Systems Journal, vol. 8, no. 3, pp. 965–979, 2014.

[4] J. Kawahara, A. BenTaieb, and G. Hamarneh, “Deep features to classify skin lesions” IEEE International Symposium on Biomedical Imaging (IEEE ISBI), pp. 1397–1400.