A

Project Report on

OSTEOARTHRITIS PREDICTION AND REPORT GENERATION USING COMPUTER VISION

Submitted to the partial fulfillment of the requirement for the award of the degree of

BACHELOR OF TECHNOLOGY
IN
COMPUTER SCIENCE & ENGINEERING WITH SPECIALIZATION IN BIG DATA ANALYTICS

Submitted by:
Samyak Mohelay (RA1911027030031)
Shivalika Karan Bora (RA1911027030034)
Vanika Gehani (RA1911027030048)

Supervised by:

Dr. Anand Pandey
(Assistant Professor)

INSTITUTE OF SCIENCE AND TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)
DELHI-NCR CAMPUS, GHAZIABAD (U.P)

SRM Institute of Science and Technology

Delhi NCR Campus, Modinagar,

Ghaziabad (UP)-201204

APRIL 2023

BONAFIDE CERTIFICATE

This is to certify that the project Report entitled "**Osteoarthritis Prediction and Report Generation using Computer Vision**", which is submitted by Samyak Mohelay (Reg No: RA1911027030031), Shivalika Karan Bora (Reg No: RA1911027030034) and Vanika Gehani (Reg No- RA1911027030048) in the partial fulfillment of the requirement for the award of degree B.Tech(CSE –Specialization in Big Data Analaytics) of SRM Institute of Science and Technology, Delhi-NCR Campus, Modinagar, Ghaziabad is a record of the candidate own work carried out by them under my own supervision.

……………………........
(Signature)
Dr. Anand Pandey
Project Supervisor
Associate Professor

……………………........
(Signature)
Dr. Niranjan Lal

Associate Professor(HOD CSE-Data Science)

Internal Examiner

External Examiner

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to Dr. Anand Pandey, Assistant Professor and Project Supervisor at SRM Institute of Science and Technology, Delhi-NCR Campus, Modinagar. His insightful knowledge and guidance inspired us to work diligently on our project.

We feel compelled to articulate our thankfulness to Dr. Jitendra Singh, and Dr. Rakesh Kumar Yadav, Project Coordinators, SRM Institute of Science and Technology, Delhi-NCR Campus, Modinagar, for their enlightening dexterity which was truly a perennial source of inspiration.

We would like to express our hearty gratitude to the respected Director, Dr. S. Vishwanathan, SRM Institute of Science and Technology, Delhi-NCR Campus, Modinagar, under whose auspices we were able to work on my project work.

We would like to express our hearty gratitude to Dr. D. K. Sharma, Dean, SRM Institute of Science and Technology, Delhi-NCR Campus, Modinagar for his valuable guidance and support.

Also, we extend our sincere regards and thanks to Dr. R.P. Mahapatra, Head of the Department, Computer Science and Engineering, & Dean Admission, SRM Institute of Science and Technology, Delhi-NCR Campus, Modinagar, for his suggestions and motivation during the completion of our project.

We are also indebted to all the teaching and non-teaching staff members of our college for helping us directly or indirectly by all means throughout the course of our study and project work.

Finally, we take this opportunity to thank our parents, family members and friends for their moral support and help and extended thanks to our well-wishers.

SAMYAK MOHELAY (RA1911027030031)

SHIVALIKA KARAN BORA (RA1911027030034)

VANIKA GEHANI (RA1911027030048)

DECLARATION

We, Samyak Mohelay (RA1911027030031), Shivalika Karnan Bora (RA1911027030035), and Vanika Gehani (RA1911027030048), hereby affirm that the project report titled "**Osteoarthritis Prediction and Report Generation using Computer Vision**" represents the genuine and original work conducted by us from January '23 to May '23. This report is being submitted as a partial fulfillment for the degree of "Bachelor of Technology in Computer Science and Engineering" at SRM IST, NCR Campus, Ghaziabad (U.P.). We confirm that this work has not been previously submitted to any other university or institute for the purpose of obtaining any degree or diploma.

Samyak Mohelay (RA1911027030031)
Shivalika Karan Bora (RA1911027030034)
Vanika Gehani (RA1911027030048)

ABSTRACT

The electronic reporting process takes time and skill. Radiology information systems should be designed to reduce the workload. Despite advances in deep learning in image classification and annotation tasks, creating accurate annotations for radiology journals remains challenging due to the complexity of the clinical content found. In addition, limited open access data, including medical images and reports, complicates the training of this model. To solve these problems, we offer an encoder/decoder design specifically for knee images and reports. While using multi-image views to achieve consistency, we first train the encoder using multiple knee X-ray images to accurately identify the corresponding radiological observations. We improve decision making with explanations by extracting and editing treatment strategies based on electronic information from the report. We also use encoders to extract the clinical information from x-ray pictures. This is combined with a decoding process that uses language-level colour models.

TABLE OF CONTENTS

BONAFIDE CERTIFICATE……………………………………………………………….2

ACKNOWLEDGEMENT…………………………………………………………………..3

DECLARATION………………………………………………………………………….4

ABSTRACT……………………………………………………………………………….5

LIST OF FIGURES……………………………………………………………………….7

LIST OF TABLES…………………………………………………………………….....8

1. INTRODUCTION 8-10

1.1 Introduction to System. 8

1.2 Computer Vision Introduction 9

1.3 Details about Osteoarthritis 10

2. LITERATURE SURVEY 11-14

3. EXISTING PROBLEM & PROPOSED SOLUTIONS 15-16

4. METHODOLOGY 17-20

4.1 Working of the model. 17

4.2 Technologies used. 18

4.3 Details of the model. 19

5. IMPLEMENTATION & RESULT 21-32

5.1 Creation of the Dashboard. 21

5.2 Testing of all the models. 28

5.3 Pre-Processing of the original images. 30

6. CONCLUSION & FUTURE SCOPE 33

7. REFERENCES 34

LIST OF FIGURES

Figure 1.1: How does Computer Vision VS Human Vision work? 8

Figure 1.2: An Osteoarthritis affected knee VS non affected knee 9

Figure 3.1: Class Diagram 15

Figure 3.2: Use Case Diagram. 16

Figure 4.1: State Chart Diagram 19

Figure 5.1: Login/Registration Page for the Patient. 22

Figure 5.2: User logs in or registers using his/her credentials .25

Figure 5.3: Dashboard Page 27

Figure 5.4: User’s Reports. 28

Figure 5.5: Report Example 1. 29

Figure 5.6: Report Example 2 30

Figure 5.7: Report Example 3. 31

Figure 5.8: Original Image Vs Pre-processed Images. 32

Figure 6.1: Flow Diagram. 32

Figure 6.2: Our Model Vs Industry Standard Model. 33

LIST OF TABLES

2.1 Tabular Summary of Literature Survey 7

CHAPTER 1

INTRODUCTION

1.1 Introduction to System

Basically the whole system can be added up as an ‘Osteoarthritis Grade Prediction and Concise Report Generation System’. Primarily the main provocation behind the design was that the traditional system to observe X-rays and report generation by a Radiologist takes too important time substantially due to the high population, dependence on the experience of the Radiologist, vestiges on X-ray images, operation of physical reports, and Lack of proper structure and services in pastoral regions. A full-fledged system with 1/ 5th the size of an assiduity-standard model and a personal pre-processing system that's suitable to resolve utmost blights from images. Alongside that, a robust web operation with a proper database operating system. The system is veritably modular and is designed in a way to be more scalable in the future.

1.2 Computer Vision Introduction

Computer vision is the field of AI that enables computers and machines to make decisions or make recommendations based on important information from pictures, videos and other visuals. However, computer vision allows the AI to see, monitor and understand if the computer is doing what it needs.

CV works a lot like human vision. However, human vision has some advantage of environmental continuity and teaches how to tell objects piece by piece, how low they are, whether they are moving, and whether the image is false. By combining cameras, data, and algorithms, computer vision teaches robots to perform these tasks far more quickly than they could with the help of retinas, optical jitter, and the visual brain.

As a result of the fact that a system trained to examine items / product assets can examine thousands of products or process nanoseconds, it can quickly surpass mortal capabilities, noting unpleasant blights or problems. Computer vision requires a lot of data. It continually analyses the data until it can spot the differences and identify the photos. To teach a computer, for instance, to make a machine's tyres fatter, it needs to know the differences in a large amount of tire images and tire-related details, and feed it with no particular ambiguity to stimulate the tires.

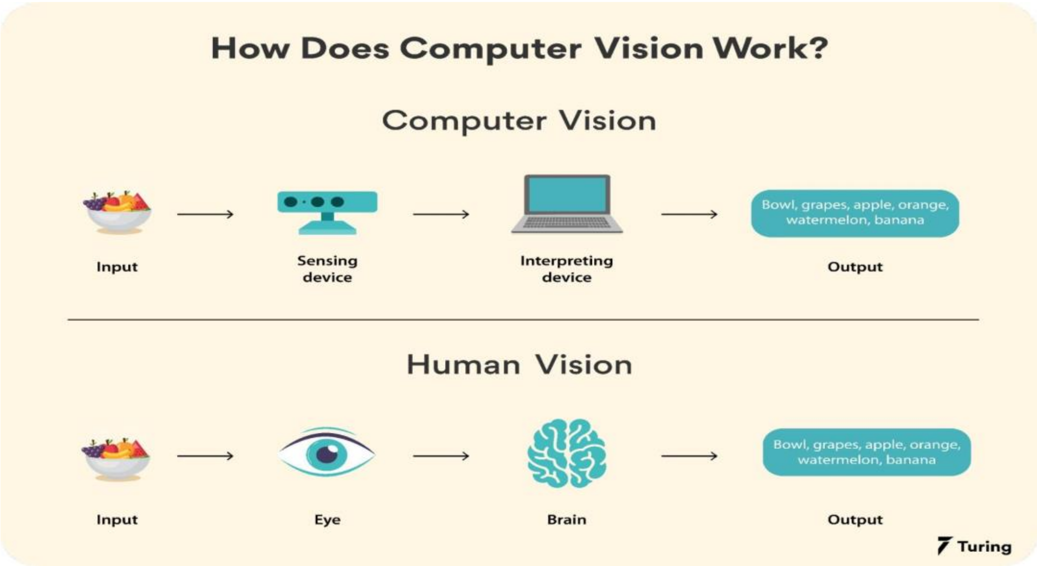


Figure 1.1: How does Computer Vision VS Human Vision work?

1.3 Details about Osteoarthritis

Osteoarthritis, one of the most common types of arthritis, affects millions of people worldwide. It manifests itself when the protective cartilage that covers the ends of one's bones gradually erodes. With a prevalence of 22% to 39% and more than 10 million cases in the previous year, osteoarthritis is India's most prevalent joint illness and the second most common rheumatic condition. The knee is one of the joints that is commonly affected by osteoarthritis.

After sustained stress, the cartilage in the knee can begin to break down, osteoarthritis is the result of the knee bones rubbing against one another. To check it, the radiologist observes the X-ray report of the patient. Arthritic knee X-rays may show joint space narrowing, bony changes, and the formation of bone spurs (osteophytes). After determining the osteoarthritis grade, the radiologist gives corresponding comments and then reports are finalized.

Figure 1.2: An Osteoarthritis affected knee VS non affected knee

However, there is a significant time involved in this process (can vary from 2 to 4 days in India) of generating the final report and especially in areas where the population is considerably high and number of medicine practitioners are comparably low. Another thing to take into consideration especially for a developing country like India is that in private hospitals, the cost of treatment is high and in government hospitals there is a lot of overhead and managing physical reports can be difficult. In rural regions, there is an unavailability of infrastructure and healthcare services and considering results are highly dependent on the experience of a radiologist it would be quite effective to automate this whole process.

CHAPTER 2

LITERATURE SURVEY

Here this chapter contains the literature Survey of all the research papers that we have studied for this project.

The extensive use of medical images in the medical profession for the treatment and diagnosis of various diseases is the subject of research papers authored by Changchang Yin, Buyu Qian, Jishang Wei, and others, among others. However, it is a laborious and time- I consuming operation to analyse medical images and summarise their insights, which might cause bottlenecks in the clinical diagnosis procedure. Automatic report production may be useful in resolving this issue. But producing medical reports comes with two significant difficulties. First, it can be difficult to precisely identify every abnormality, particularly rare disorders. Second, compared to natural image captions, medical image reports typically contain multiple paragraphs and phrases, which makes it difficult to generate accurate and diverse reports.

To address these issues, the authors propose a new framework that accurately detects anomalies and automatically creates medical records from initial medical photographs. The Hierarchical Recurrent Neural Network (HRNN) that the report generating model is based on employs a topic matching technique to increase the precision and variety of the generated reports. The HRNN model also incorporates a soft-focus mechanism to improve overall performance.

The authors also suggest the Global Label Pooling method, which is found to perform better than the current GFP (Global Feature Pooling) mechanism, for anomaly identification. Experimental findings on two datasets of image-paragraph pairs demonstrate that the given framework performs better than all current state of the art approaches.

Overall, the framework presents a promising solution for addressing the challenges of generating accurate and diverse medical reports from medical images, which can improve the efficiency and effectiveness of clinical diagnosis.

A research effort on automatic radiology report production based on multimodal picture fusion and medical insight enrichment is presented by the Department of Informatics. It takes too much of time, effort, and talent to produce reliable automated radiology reports, which are time-consuming tasks. The complexity of medical visual content and the need for precise natural language descriptions make writing radiology reports difficult, even though DL. techniques are successfully applied to picture annotation and image classification. Additionally, the size of open databases that contain matched reports and medical images is constrained. The authors suggest an encoder decoder model that concentrates on X-ray of chest pictures and its report and incorporates these changes to get around this problem.

The authors provide a brand-new encoder-decoder approach for the creation of radiology reports that integrates a variety of visual focus radiological data types. The hierarchical LSTM decoder's semantics are captured by the model using medical terms. The suggested model extracts and visualises ambiguous radiography findings as a priceless added advantage, alerting more scientists to these uncertainties for further investigation in practise. Overall, the proposed approach uses improved visual and textual material to attain cutting-edge performance.

Research by J Poon, MMA Monshi, V Chung from the University of Sydney shows significant progress in developing an automated radiology reporting model based on deep learning (DL). This progress is facilitated by the availability of large medical image/text databases. Recent academic attention has focused on creating sequential paragraphs in radiology reports, a more practical and complex program that requires bridging visual medical features and radiologist texts. To achieve this goal, a DL model combining CNN for image analysis, RNN for natural language processing, and natural language generation (NLG) has been developed. The most common approach is to use publicly available databases to train these models.

We hope that this field of research will be expanded in the near future. In our work, we explore key challenges such as understanding radiology text and image structure and database, using deep learning (mainly CNN and RNN) algorithms, generating radiology text, and enhancing current DL-based models and evaluation metrics. We also provide critical analysis and suggestions for future research. Our questionnaire will be useful for researchers interested in in-depth study, especially those who intend to use it in radiology reports.

Table 2.1: Tabular Summary of Literature Survey

S. NO.	ARTICLE	KEY CONTRIBUTIONS	RESULTS
1	**Pingjun Chen, Linlin Gao, Xiaoshuang Shi, Kyle Allen, Lin Yang's "Fully automatic knee osteoarthritis severity grading using deep neural networks with a novel ordinal loss", 2019**	This article discusses the impact of knee osteoarthritis (OA) on older adults, highlighting the importance of early detection and intervention in slowing down the progression of the disease. The current grading system, based on visual inspection, is subject to interpretation and can vary widely depending on the experience of the physician. To address this issue, the article proposes the use of two deep convolutional neural networks (CNNs) to automatically measure the severity of knee OA, as assessed by the Kellgren-Lawrence grading system.	The paper discusses the use of a customised YOLOv2 model for detecting knee joints and fine-tuning CNN models with a novel ordinal loss for knee KL grading. The approach achieves state-of-the-art performance for both knee joint detection and knee KL grading. The YOLOv2 model is found to be well-suited for detection tasks with less varied object size, based on its performance in knee joint detection. Furthermore, the proposed ordinal loss helps improve classification accuracy and reduces the mean absolute error (MAE)
2	**Changchang Yin, Buyue Qian, Jishang Wei, Xiaoyu Li's "Automatic Generation of Medical Imaging Diagnostic Report with Hierarchical Recurrent Neural Network", 2019**	This article presents a novel framework for accurately detecting abnormalities and automatically generating medical reports. The topic matching system is incorporated into the report generating model, which is based on a HRNN, in order to increase the precision and variety of the reports produced. The HRNN model also incorporates a soft attention mechanism for further refinement.	The paper introduces a new framework that aims to detect diseases and generate medical reports from initial images. The proposed framework includes a GLP mechanism, which outperforms GFP in the abnormality detection experiment.
3	**Jianbo Yuan , Haofu Liao, Rui Luo, and Jiebo Luo's "Automatic Radiology Report Generation based on Multi-view Image Fusion and Medical Concept Enrichment ", 2019**	The generation of radiology reports is a time-consuming process that requires a high level of expertise. To address this, there is a need for radiology report generation to alleviate the workload. While deep learning techniques have been successful in tasks such as image classification and captioning, generating radiology reports is challenging due to the need for understanding and linking complex medical visual content with accurate natural language descriptions.	Thanks to modern techniques for training convolutional neural networks, even the most basic architectures can achieve remarkable performance. For instance, networks consisting solely of convolutions and subsampling operations can outperform, or at least match, state-of-the-art models on CIFAR-10 and CIFAR-100. A similar architecture can also deliver competitive outcomes on ImageNet. It is noteworthy that contrary to previous findings, including explicit pooling operations like max-pooling doesn't always enhance performance of CNNs. This is especially true when the network is large enough to learn all the necessary invariances using convolutional layers alone for the given dataset.

CHAPTER 3

EXISTING PROBLEMS & PROPOSED SOLUTIONS

Problem

- The most common form of arthritis is osteoarthritis, also known as degenerative joint disease. Osteoarthritis develops as people age. Osteoarthritis changes usually occur slowly over many years, although there are exceptions.
- Joint inflammation and injury cause bone changes, tendon and ligament deterioration, and cartilage breakdown resulting in pain, swelling, and joint deformity.
- To examine Osteoarthritis, a Radiologist observes the X-ray Report of the Patient. X-rays of arthritic knees can narrow the joint space, changes in bones, and the formation of bone spurs. After determining the grade of osteoarthritis, the Radiologist gives corresponding comments and then reports are finalised.
- However, there is a significant time involved in this process (can vary from 2 to 4 days in India) of generating the final report and especially in areas where the population is considerably high and number of medicine practitioners are comparably low.

Solution

On the other hand Automated Osteoarthritis report generation helps identify patients and reduce the workload for doctors. Reports produced by automated medical report generators must be reliable, easy to understand, and accurate to be used effectively in practice. The quality of the explanation of how the report is made and how the diagnosis is achieved is a key factor in achieving this goal. Having an interpretable system allows developers to identify any flaws or inefficiencies, and clinicians have confidence in the decisions they make with this system.

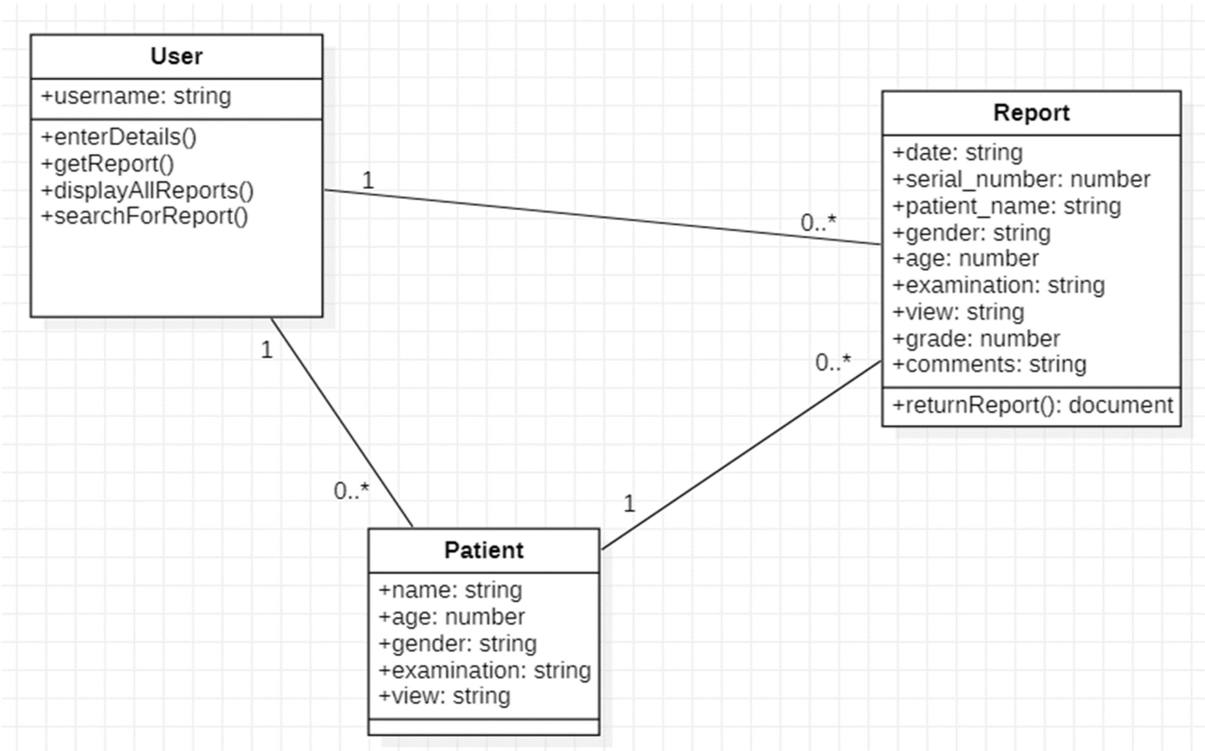


Figure 3.1: Class Diagram

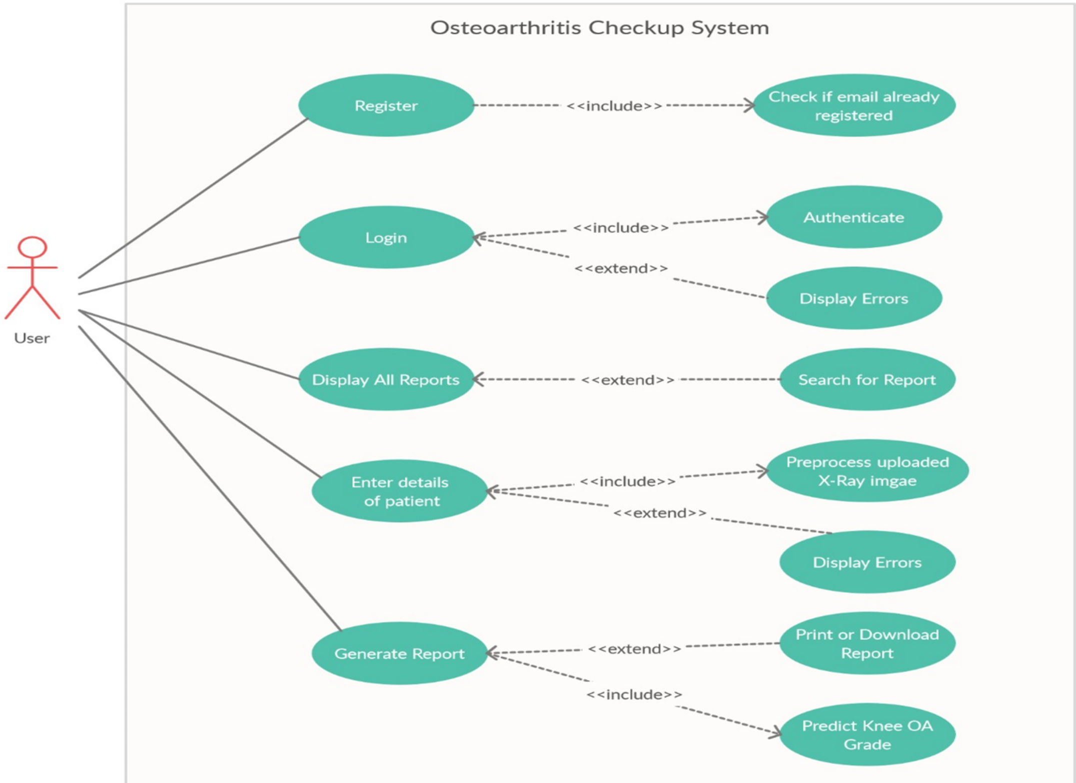


Figure 3.2: Use Case Diagram

CHAPTER 4

METHODOLOGY

The approach that we brought into use while constructing this system was the **agile development approach**.

The following features were included in the system:

- Register/Login user
- Generating new report
- Download/Print report
- User details and report management
- View-download-search patients reports based on :
  - Serial no.
  - Gender
  - Grade
  - Name
  - Age
- Change system's language (108 languages including Hindi and Punjabi)

Working of the system and the technologies used-

The whole system is based on the working of the frontend and the backend wherein **Node.Js** along with **Express Framework** is used in the backend and **HTML, CSS along with Javascript** is used to build the frontend part.

Working:

- On opening of the web application, the user is directed to a login/registration page. The login system used the passport local strategy to authenticate users. **MongoDB** is used as the database to store user information as well as the patient information and report data.
- The passwords created or entered by the user at the “login your credentials” page is encrypted and is made secure by **bcryptjs**.
- On successful login by the user, he/she gains an access to the dashboard from which the user can generate a new report or visit his/her old reports stored in the database in their leisure time. Most of the webpages are dynamic and are made possible by the **EJS Package in Node**.
- After entering the details of a new patient and uploading their X-ray scan, a deep learning model predicts the grade of osteoarthritis using **python script** on the basis of the given information and generates an HTML report which is converted into a PDF format using the **puppeteer API**. The user now has the option to print/download the report and save the information onto the database.
- Going back to the dashboard, the details of the patients are added and the report can be accessed from the “View Reports” section. The server queries the database for information and displays the information of all the patients in a tabular format.
- Reports can be re-downloaded and the user also has an option to search for patients according to their serial no. , name, age, etc.
- System also has the feature to change the language of the entire interface using google-translate-script. (108 languages including Hindi and Punjabi).
- Finally, the user can logout to end the session.

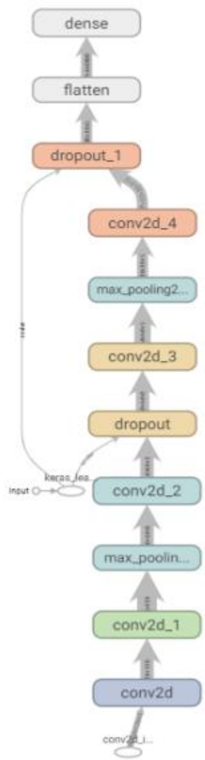
Details of the model:

The basic structure of the model is inspired by the VGG series with a 2D iterative convolutional layer and a max-print layer increases by units when we move to the bottom of model. Dropout is used alternatively in layers without any sort of regularisation to get the most effective neurons in that layer at the end of the process. For layers without any dropout, the model utilises L2 regularisation to maintain the weights to a comparably smaller amount.

Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 120, 120, 64)	1792
conv2d_1 (Conv2D)	(None, 120, 120, 256)	147712
max_pooling2d (MaxPooling2D)	(None, 60, 60, 256)	0
conv2d_2 (Conv2D)	(None, 60, 60, 256)	590080
dropout (Dropout)	(None, 60, 60, 256)	0
conv2d_3 (Conv2D)	(None, 60, 60, 512)	1180160
max_pooling2d_1 (MaxPooling2	(None, 30, 30, 512)	0
conv2d_4 (Conv2D)	(None, 30, 30, 512)	2359808
dropout_1 (Dropout)	(None, 30, 30, 512)	0
flatten (Flatten)	(None, 460800)	0
dense (Dense)	(None, 5)	2304005

Total params: 6,583,557
Trainable params: 6,583,557
Non-trainable params: 0



The diagram below depicts different states the system can exist in, during runtime and it helps in getting a view of the system’s working as a whole.

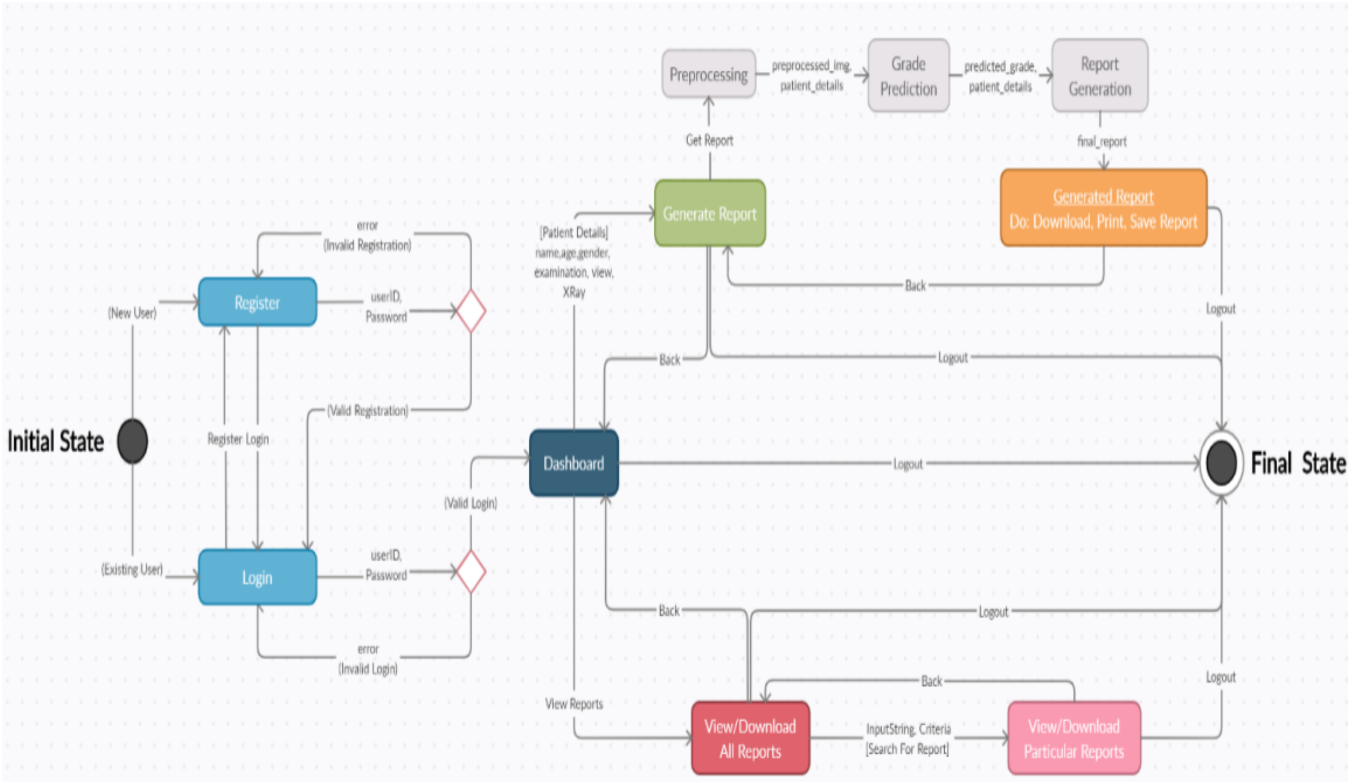


Figure 4.1: State Chart Diagram

CHAPTER 5

IMPLEMENTATION AND RESULTS

Module 1: Creation of the login/registration page

The source code for the osteoarthritis prediction and report generation using computer vision is:

In [1]:
```
import numpy as np
import cv2
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Activation, Dense, Flatten, BatchNormalization, Conv2D, MaxPool2D
from tensorflow.keras.layers import Dropout
from keras import regularizers
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.metrics import categorical_crossentropy
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from sklearn.metrics import confusion_matrix
import itertools
train_path = 'KneeXrayData/train'
valid_path = 'KneeXrayData/val'
test_path = 'KneeXrayData/test'
```

In [2]:
```
def preprocess(img):
    img = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
    #img = tf.keras.preprocessing.image.img_to_array(img,dtype=np.uint8)
    #img = np.expand_dims(img, axis=0)
    img = np.array(img,dtype=np.uint8)
    img = cv2.GaussianBlur(img, (3, 3), 7)
    mean = int(np.mean(img))
    for i in range(0,img.shape[0]):
        for j in range(0,img.shape[1]):
            img[i,j] = max(0,img[i,j]-mean)
    clahe = cv2.createCLAHE()
    img = clahe.apply(img)
    img = img.reshape(-1, 224, 224, 1)
    return img/255.0

#Image Loading and Preprocessing
train_batches = ImageDataGenerator(preprocessing_function=preprocess).flow_from_directory(train_path, target_size=(224,224), classes=['0', '1','2','
valid_batches = ImageDataGenerator(preprocessing_function=preprocess).flow_from_directory(valid_path, target_size=(224,224), classes=['0', '1','2','
test_batches = ImageDataGenerator(preprocessing_function=preprocess).flow_from_directory(test_path, target_size=(224,224), classes=['0', '1','2','3'
```

In [ ]:
```
import matplotlib.pyplot as plt
# plots images with labels within jupyter notebook
def plots(ims, figsize=(12,6), rows=1, interp=False, titles=None):
    if type(ims[0]) is np.ndarray:
        ims = np.array(ims).astype(np.uint8)
        if (ims.shape[-1] != 3):
            ims = ims.transpose((0,2,3,1))
    f = plt.figure(figsize=figsize)
    cols = len(ims)//rows if len(ims) % 2 == 0 else len(ims)//rows + 1
    for i in range(len(ims)):
        sp = f.add_subplot(rows, cols, i+1)
        sp.axis('Off')
        if titles is not None:
            sp.set_title(titles[i], fontsize=16)
        plt.imshow(ims[i], interpolation=None if interp else 'none')
imgs, labels = next(train_batches)
plots(imgs, titles=labels)
```

In [ ]:
```
#train_batches = train_batches.reshape(-1, 224, 224, 1)
imgs, labels = next(train_batches)
print(imgs[0].shape)
#print(type(train_batches))
```

```
In [ ]: #Regularized Model
model = Sequential()
model.add(Conv2D(input_shape=(224,224,3),filters=64,kernel_size=(3,3),padding="same", activation="relu"))
model.add(MaxPool2D(pool_size=(2,2),strides=2))
model.add(Conv2D(filters=256, kernel_size=(3,3), padding="same", activation="relu", kernel_regularizer=regularizers.l2(0.01)))
model.add(Conv2D(filters=256, kernel_size=(3,3), padding="same", activation="relu", kernel_regularizer=regularizers.l2(0.01)))
model.add(Dropout(0.3))
model.add(MaxPool2D(pool_size=(2,2),strides=2))
model.add(Conv2D(filters=256, kernel_size=(3,3), padding="same", activation="relu", kernel_regularizer=regularizers.l2(0.01)))
model.add(Conv2D(filters=256, kernel_size=(3,3), padding="same", activation="relu", kernel_regularizer=regularizers.l2(0.01)))
model.add(Dropout(0.3))
model.add(MaxPool2D(pool_size=(2,2),strides=2))
model.add(Flatten())
model.add(Dense(units=5, activation="softmax"))
model.compile(Adam(lr=.0001), loss='categorical_crossentropy', metrics=['accuracy'])
```

```
In [ ]: #Extensive
from tensorflow.keras.layers import Dropout
from keras import regularizers
model = Sequential()
model.add(Conv2D(input_shape=(224,224,3),filters=64,kernel_size=(3,3),padding="same", activation="relu", kernel_regularizer=regularizers.l2(0.1)))
model.add(Dropout(0.5))
model.add(MaxPool2D(pool_size=(2,2),strides=2))
model.add(Conv2D(filters=256, kernel_size=(3,3), padding="same", activation="relu", kernel_regularizer=regularizers.l2(0.1)))
model.add(Dropout(0.5))
model.add(Conv2D(filters=256, kernel_size=(3,3), padding="same", activation="relu", kernel_regularizer=regularizers.l2(0.1)))
model.add(Dropout(0.5))
model.add(MaxPool2D(pool_size=(2,2),strides=2))
model.add(Conv2D(filters=256, kernel_size=(3,3), padding="same", activation="relu", kernel_regularizer=regularizers.l2(0.1)))
model.add(Dropout(0.5))
model.add(Conv2D(filters=256, kernel_size=(3,3), padding="same", activation="relu", kernel_regularizer=regularizers.l2(0.1)))
model.add(Dropout(0.5))
model.add(MaxPool2D(pool_size=(2,2),strides=2))
model.add(Flatten())
model.add(Dense(units=5, activation="softmax"))
model.compile(Adam(lr=.0001), loss='categorical_crossentropy', metrics=['accuracy'])
```

```
In [ ]: model.summary()
```

```
In [ ]: model.fit(train_batches, batch_size=10 , validation_data=valid_batches, epochs=13, verbose=2)
```

```
In [ ]: #2
model.save('dataset/OF_2.h5')
```

```
In [ ]: predictions = model.predict(test_batches, batch_size=10, verbose=0)
```

```
In [ ]: Grade_predictions = np.argmax(predictions,axis=-1)
```

```
In [ ]: print(predictions)
```

```
In [ ]: for i in Grade_predictions:
    print(i)
```

```
In [ ]: test_labels = test_batches[1]
```

```
In [ ]: cm = confusion_matrix(test_labels, Grade_predictions)
```

The outputs for the code mentioned above are as follows:

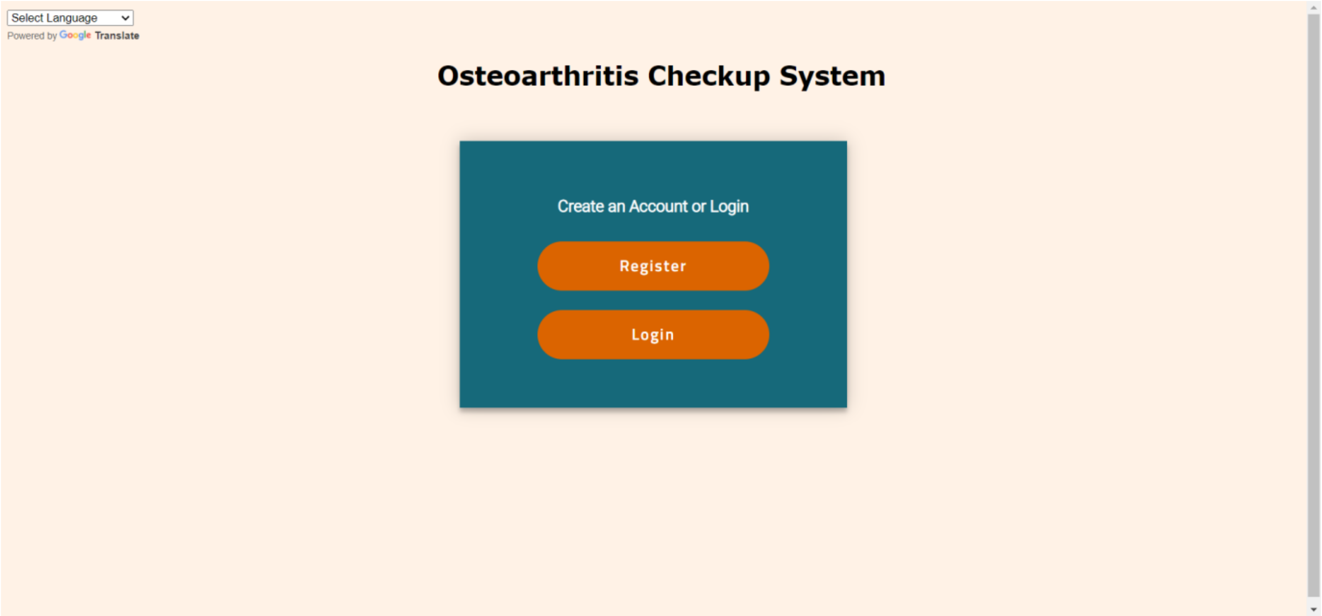


Figure 5.1: Login/Registration Page for the Patient

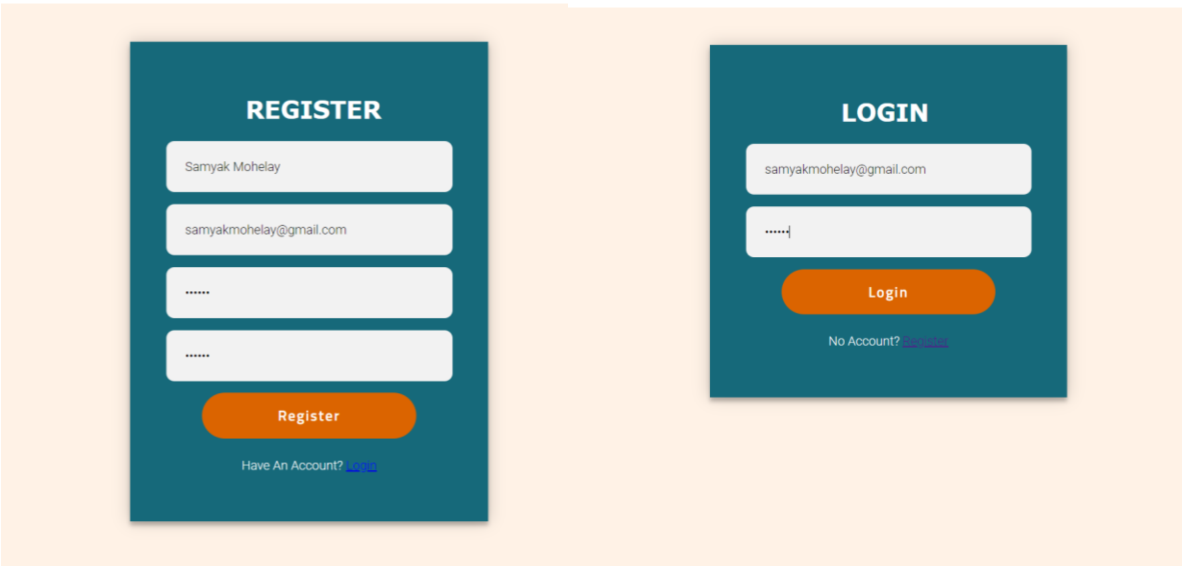


Figure 5.2 : User logs in or registers using his/her credentials

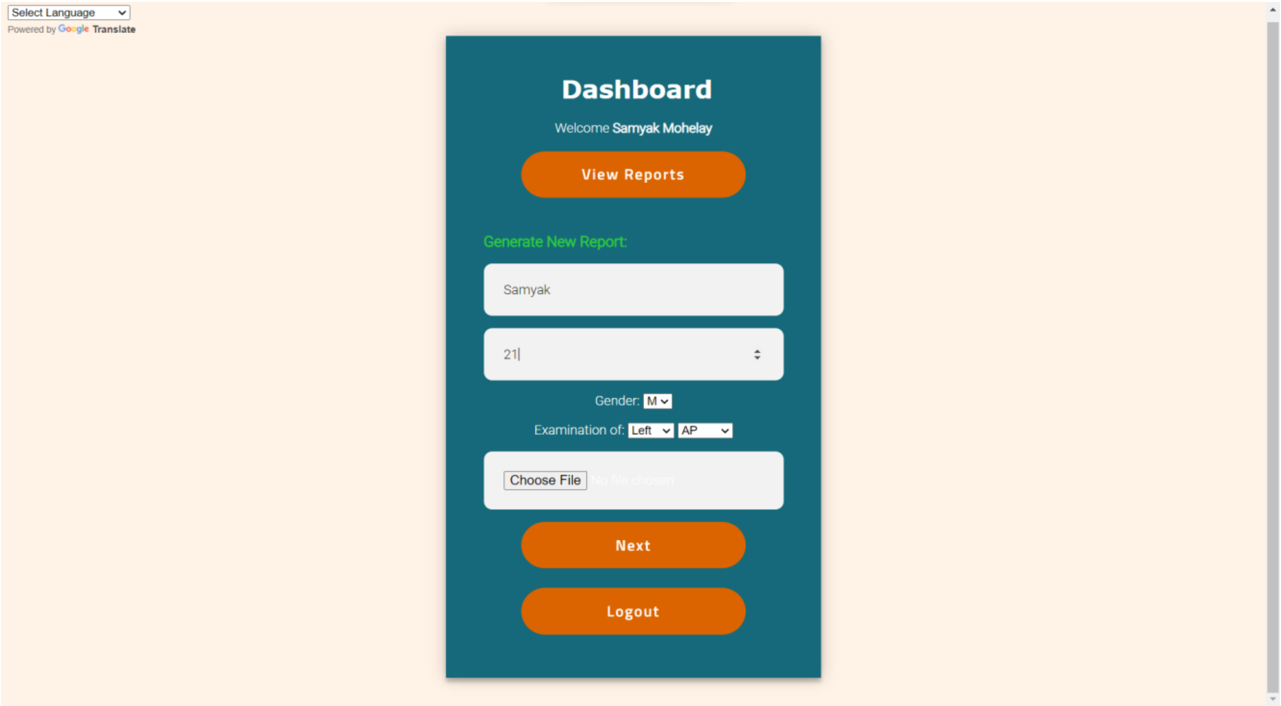


Figure 5.3: Dashboard Page

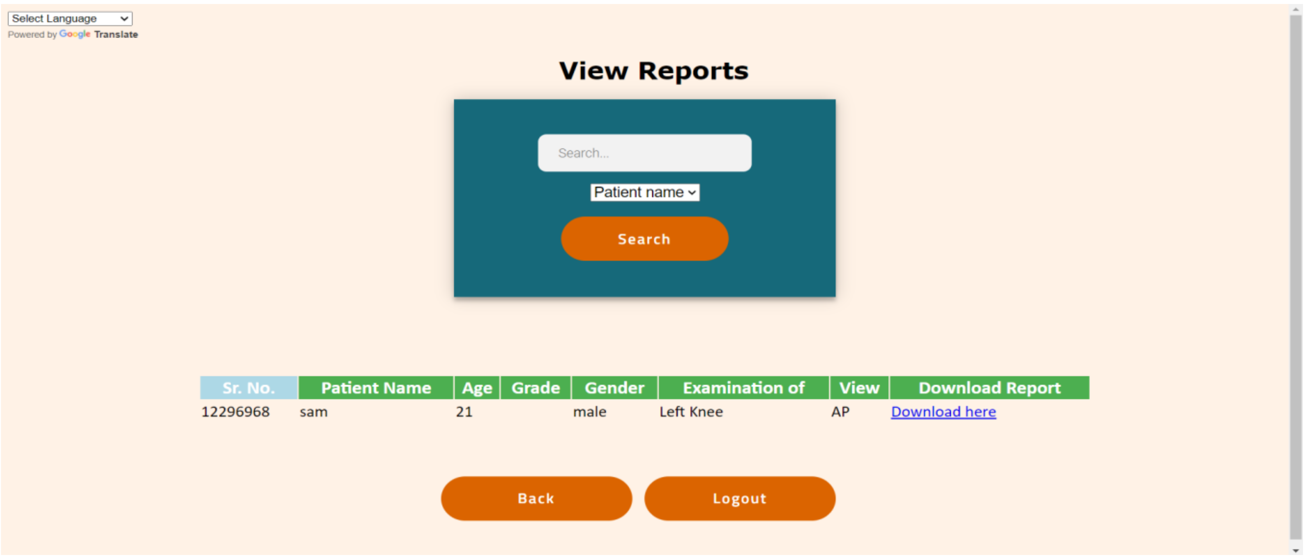


Figure 5.4: User’s Reports

Module 2: Finding all the models:

The source code for all the testing models are as follows:

```
import numpy as np
import pandas as pd
import keras
import tensorflow as tf

from keras.models import Sequential
from keras.optimizers import Adam
from keras.losses import categorical_crossentropy
from keras.layers import Dense, Flatten, Conv3D, MaxPooling3D, DropoutDense, Flatten, Conv2D, MaxPooling2D, Dropout,GlobalAveragePooling2D,BatchNormalization
from sklearn.model_selection import train_test_split
'''These are models that were used to experiment and these were run on sample dataset.These models are originally done on Cifar-10 Dataset and used for classification pur
Lack of Regularization was the most important factor that observed through these models. Also some changes are also made in layers so as to adjust to our dataset.'''
'''Takes the Input Shape(inputShape as parameter is passed) and returns the model trained over dataset'''
def ConvPool_CNN_C(inputShape):
  model = Sequential()
  model.add(Conv2D(96,kernel_size=(3,3),activation='relu',padding='same'))
  model.add(Conv2D(96,kernel_size=(3,3),activation='relu',padding='same'))
  model.add(Conv2D(96,kernel_size=(3,3),activation='relu',padding='same'))
  model.add(MaxPooling2D(pool_size=(3,3),strides=2))
  model.add(Conv2D(192,(3,3),activation='relu',padding='same'))
  model.add(Conv2D(192,(3,3),activation='relu',padding='same'))
  model.add(Conv2D(192,(3,3),activation='relu',padding='same'))
  model.add(MaxPooling2D(pool_size=(3,3),strides=2))
  model.add(Conv2D(192,(3,3),activation='relu',padding='same'))
  model.add(Conv2D(192,(1,1),activation='relu'))
  model.add(Conv2D(5,(1,1)))
  model.add(GlobalAveragePooling2D())
```

```
  model.add(Flatten())
  model.add(Dense(5, activation='softmax'))
  model.build(inputShape)
  model.compile(loss=categorical_crossentropy,optimizer=keras.optimizers.Adam(0.001),metrics=['accuracy'])
  return model

def all_cnn_c(inputShape):
  model = Sequential()
  model.add(Conv2D(96,kernel_size=(3,3),activation='relu',padding='same'))
  model.add(Conv2D(96,kernel_size=(3,3),activation='relu',padding='same'))
  model.add(Conv2D(96,kernel_size=(3,3),activation='relu',padding='same'))
  model.add(Conv2D(192,(3,3),activation='relu',padding='same'))
  model.add(Conv2D(192,(3,3),activation='relu',padding='same'))
  model.add(Conv2D(192,(3,3),activation='relu',padding='same'))
  model.add(Conv2D(192,(3,3),activation='relu',padding='same'))
  model.add(Conv2D(192,(1,1),activation='relu'))
  model.add(GlobalAveragePooling2D())
  model.add(Dense(5, activation='softmax'))
  model.build(inputShape)
  model.compile(loss=categorical_crossentropy,optimizer=Adam(0.001),metrics=['accuracy'])

  return model

def nin_cnn_c(inputShape):
  model = Sequential()
  model.add(Conv2D(32,kernel_size=(5,5),activation='relu',padding='valid'))
  model.add(Conv2D(32,kernel_size=(5,5),activation='relu'))
  model.add(Conv2D(32,kernel_size=(5,5),activation='relu'))
```

```
  model.add(MaxPooling2D(pool_size=(3,3),strides=2))
  model.add(Dropout(0.5))
  model.add(Conv2D(64,(3,3),activation='relu',padding='same'))
  model.add(Conv2D(64,(1,1),activation='relu',padding='same'))
  model.add(Conv2D(64,(1,1),activation='relu',padding='same'))
  model.add(MaxPooling2D(pool_size=(3,3),strides=2))
  model.add(Dropout(0.5))
  model.add(Conv2D(128,(3,3),activation='relu',padding='same'))
  model.add(Conv2D(32,(1,1),activation='relu'))
  model.add(Conv2D(5,(1,1)))
  model.add(GlobalAveragePooling2D())
  model.add(Flatten())
  model.add(Dense(5, activation='softmax'))
  model.build(inputShape)

  model.compile(loss=categorical_crossentropy,optimizer=Adam(0.001),metrics=['accuracy'])
  return model

def SimpleModel(inputShape):
        model = tf.keras.models.Sequential([
    keras.layers.Dense(512, activation='relu', input_shape = inputShape),
    keras.layers.Dropout(0.2),
    keras.layers.Dense(10)
    ])

        model.compile(optimizer='adam',
                loss=tf.losses.SparseCategoricalCrossentropy(from_logits=True),
                metrics=[tf.metrics.SparseCategoricalAccuracy()])

        return model
```

Sample Reports

Report

22/10/2020 11:12:33

Serial no. : 32

Patient : Saarisht

Gender: M

Age : 20

Examination of : Left Knee

View : AP

Grade : 0

Findings

1. Indicates a definite absence of X-Ray changes of Osteoarthritis
2. No Osteoarthritis changes are detected.
3. Normal X-Ray is seen

Comments

1. No need to consult doctors is recommended
2. No Osteoarthritis X-Ray changes are detected
3. No treatment is needed

Figure 5.5: Report Example 1

Report

Serial no. : 16507038

Patient : kl

Age : 25

Gender : male

Examination of : Left Knee

View : AP

Grade : 3

29/10/2020 17:21:46

Findings

1. Moderate Osteoarthritis with moderate multiple osteophytes
2. definite narrowing of joint space and some sclerosis
3. Possible deformity of bone ends is seen

Comments

1. "Moderate" stage of condition is present
2. Coricosteroids may be recommended
3. Do consult Doctor

Figure 5.6: Report Example 2

Report

Serial no. : 11813545

Patient : Dev

Age : 29

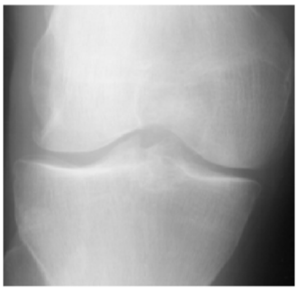
Gender : male

Examination of : Left Knee

View : AP

Grade : 2

7/11/2020 17:20:7



Findings

1. Minimal Osteoarthritis with Definite osteophytes
2. Possible joint space narrowing is seen
3. Osteoarthritis is definitely present although with minimum severity

Comments

1. "Mild" stage of the condition is present
2. Protecting joints from exertion is recommended
3. Consulting Doctors is recommended

Figure 5.7: Report Example 3

Module 3: Pre-Processing the original image:

The source code is as follows:

```python
import cv2
import numpy as np


def draw_lines(img, lines):
    try:
        for line in lines:
            coords = line[0]
            cv2.line(img, (coords[0], coords[1]),
                     (coords[2], coords[3]), [255, 255, 255], 3)
    except:
        pass

def EdgeDetection(original_image):
    processed_img = original_image
    processed_img = cv2.GaussianBlur(processed_img, (3, 3), 7)
    processed_img = cv2.Canny(processed_img, threshold1=70, threshold2=150)
    #edges
    lines = cv2.HoughLinesP(processed_img, 1, np.pi/180, 180, 20, 15)
    draw_lines(processed_img, lines)
    return processed_img

def ApplyContour(img):
    ret, thresh = cv2.threshold(img, 100, 255, 0)
    contours, hierarchy = cv2.findContours(
        thresh, cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE)
    img = np.dstack((img, img, img))
    cv2.drawContours(img, contours, -1, (0, 255, 0), 2)
    return img

#Histogram Equalisation
def ApplyCLAHE(img):
    clahe = cv2.createCLAHE(clipLimit=2.0, tileGridSize=(8,8))
    img_CLAHE = clahe.apply(img)
    return img_CLAHE

def Roi(img):
    #         y1:y2    x1:x2
    img = img[250:500, 90:450]
    return img

def Preprocess(img):
    img_roi = img#Roi(img)
    img_CLAHE = ApplyCLAHE(img_roi)
    img_edge = EdgeDetection(img_roi)
    final_img = ApplyContour(img_edge)
    #cv2.imshow('final_img', final_img)
    return final_img,img_CLAHE
```

The output is as follows:

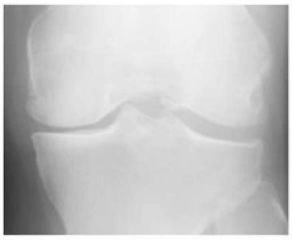
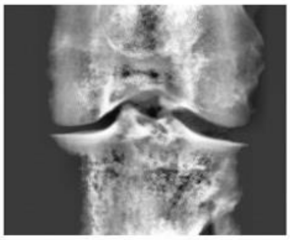
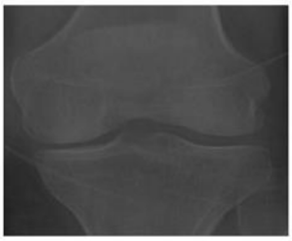
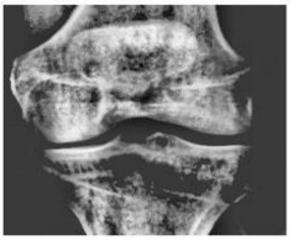
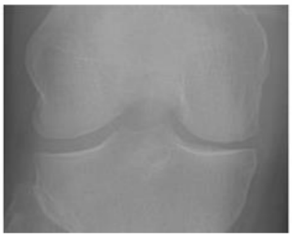
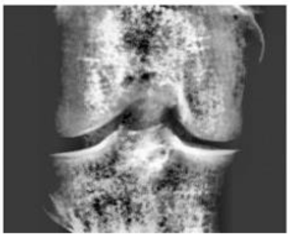
Original Image	Pre-processed Image
	
	
	

Figure 5.8: Original Image Vs Pre-processed Images

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

A prevalent joint condition that affects millions of people worldwide is osteoarthritis. Conventional diagnosis of osteoarthritis includes imaging tests such as X-rays, MRIs and CT scans.

However, these methods can be time-consuming and expensive, and often require specialised training to interpret the results. CV is a rapidly evolving area of Al which could revolutionise how osteoarthritis is diagnosed and treated.

By using computer algorithms to analyse medical images, computer vision can quickly and accurately detect patterns and anomalies that the human eye might miss. The future applications of osteoarthritis diagnosis and treatment using computer vision are enormous. For example, using computer vision to detect early signs of osteoarthritis, enabling early intervention and treatment.

It also helps monitor disease progression, leading to a more individualised treatment plan. In addition, computer vision can help assess the effectiveness of treatment options and ultimately improve treatment outcomes.

Additionally, computer vision can help develop new treatments for osteoarthritis. By analysing large amounts of medical data, computer algorithms can identify potential targets for drug therapy and help researchers better understand the underlying mechanisms of disease.

In summary, the future of osteoarthritis diagnosis and treatment using computer vision is bright, with the potential to improve patient outcomes, reduce healthcare costs, and revolutionise the medical field.

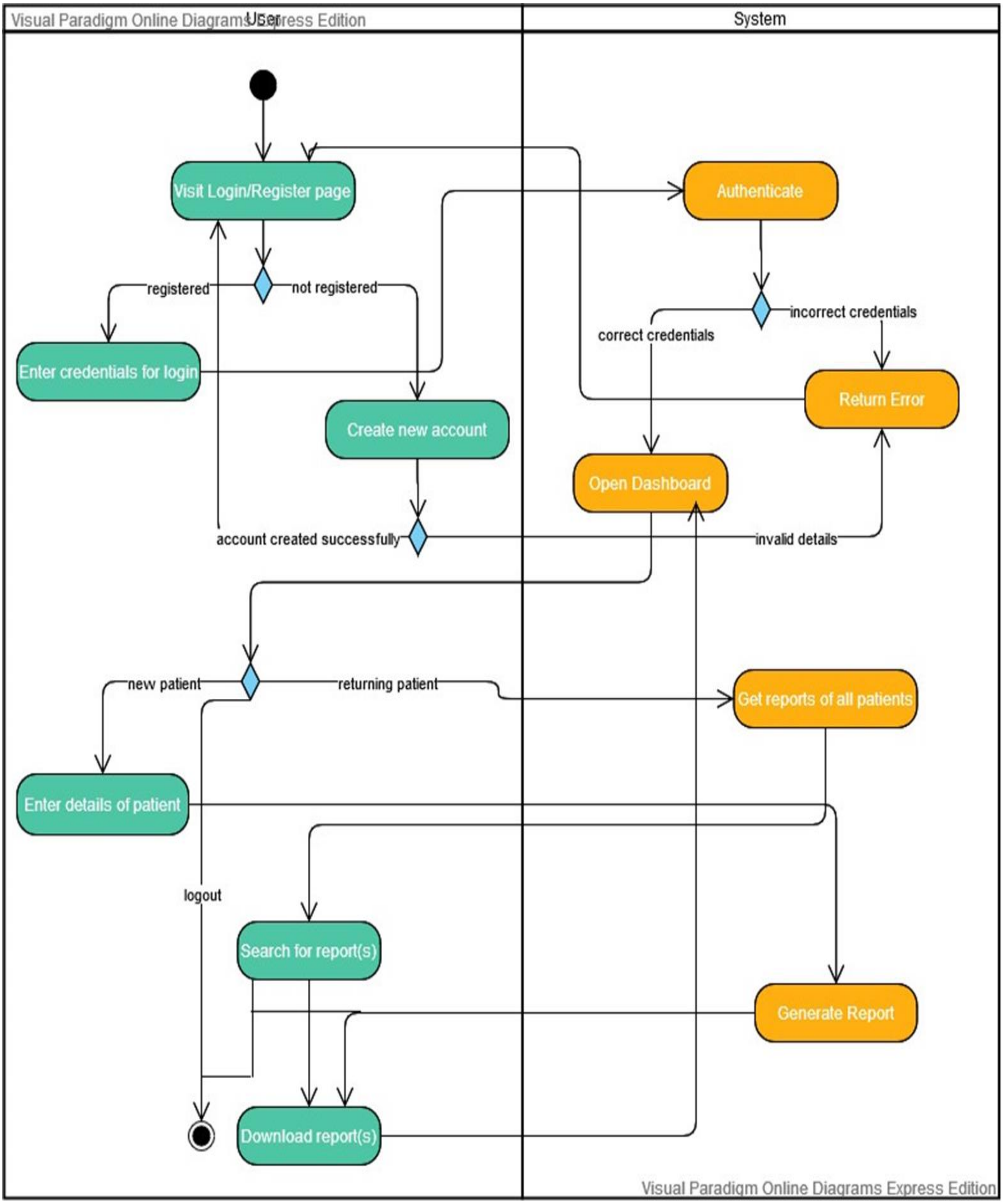


Figure 6.1: Flow Diagram

REFERENCES

1. Pingjun Chen, Linlin Gao, Xiaoshuang Shi, Kyle Allen, Lin Yang's "Fully automatic knee osteoarthritis severity grading using deep neural networks with a novel ordinal loss", 2019.

2. Changchang Yin, Buyue Qian, Jishang Wei, Xiaoyu Li's "Automatic Generation of Medical Imaging Diagnostic Report with Hierarchical Recurrent Neural Network", 2019

3. Jianbo Yuan, Haofu Liao, Rui Luo, and Jiebo Luo's "Automatic Radiology Report Generation based on Multi-view Image Fusion and Medical Concept Enrichment ", 2019

4. Knee Osteoarthritis Severity Grading Dataset - Mendeley Data :- or Dataset and Industry Standard Model Metrics.

5. 1412.6806.pdf (arxiv.org) :- For models to experiment with Dataset.