##### **DETECTION AND CLASSIFICATION OF PERIAPICAL DENTAL X-RAY IMAGES USING IMAGE PROCESSING TECHNIQUE**

##### **A PROJECT REPORT**

###### *Submitted by*

##### **ABINAYA.B [REGISTER NO:211418104006]**

**GAYATHRI.R [REGISTER NO:211418104061]**

**INDHUMATHI.M [REGISTER NO:211418104089]**

***in partial fulfillment for the award of the degree of***

**BACHELOR OF ENGINEERING**

***in***

**COMPUTER SCIENCE AND ENGINEERING**



## PANIMALAR ENGINEERING COLLEGE

**(An Autonomous Institution, Affiliated to Anna University, Chennai)**

**MAY 2022**

# PANIMALAR ENGINEERING COLLEGE

###### (An Autonomous Institution, Affiliated to Anna University, Chennai)

**BONAFIDE CERTIFICATE**

Certified that this project report **“DETECTION AND CLASSIFICATION OF PERIAPICAL DENTAL X-RAY IMAGES USING IMAGE PROCESSING TECHNIQUE”** is the bonafide work of “**ABINAYA B (211418104006), GAYATHRI R (211418104061), INDHUMATHI M (211418104089)”** who carried out the project work under my supervision.

**SIGNATURE SIGNATURE**

**Dr.S.MURUGAVALLI,M.E.,Ph.D., Dr.L.JABASHEELA,MCA.,M.E.,Ph.D.,**

**HEAD OF THE DEPARTMENT SUPERVISOR**

**Professor - CSE Department**

DEPARTMENT OF CSE, DEPARTMENT OF CSE,

PANIMALAR ENGINEERING COLLEGE, PANIMALAR ENGINEERING COLLEGE,

NASARATHPETTAI, NASARATHPETTAI,

POONAMALLEE, POONAMALLEE,

CHENNAI-600 123. CHENNAI-600 123.

Certified that the above mentioned students were examined in End Semester Project Viva-Voce Examination held on...........................

**INTERNAL EXAMINER EXTERNAL EXAMINER**

**DECLARATION BY THE STUDENT**

We Abinaya B (211418104006), Gayathri R (211418104061), Indhumathi M (211418104089) hereby declare that this project report titled “DETECTION AND CLASSIFICATION OF PERIAPICAL DENTAL X-RAY IMAGES USING IMAGE PROCESSING TECHNIQUES”, under the guidance of Dr.L.Jabasheela,M.C.A.,M.E.,Ph.D., is the original work done by us and we have not plagiarized or submitted to any other degree in any university by us.

1. **ABINAYA B**

1. **GAYATHRI R**
2. **INDHUMATHI M**

**ACKNOWLEDGEMENT**

We would like to express our deep gratitude to our respected Secretary and Correspondent **Dr.P.CHINNADURAI, M.A., Ph.D.** for his kind words and enthusiastic motivation, which inspired us a lot in completing this project.

We express our sincere thanks to our Directors **Tmt.C.VIJAYARAJESWARI**, **Dr.C.SAKTHI KUMAR, M.E., Ph.D** and **Dr.SARANYASREE SAKTHI**

**KUMAR B.E.,M.B.A.,Ph.D.,** for providing us with the necessary facilities to undertake this project.

We also express our gratitude to our Principal **Dr.K.Mani, M.E., Ph.D.** who facilitated us in completing the project.

We thank the Head of the CSE Department, **Dr. S.MURUGAVALLI, M.E., Ph.D.,** for the support extended throughout the project.

We would like to thank my guide **Dr.L.Jabasheela**, **MCA., M.E., Ph.D.,** and all the faculty members of the Department of CSE for their advice and encouragement for the successful completion of the project.

**ABINAYA B**

**GAYATHRI R**

**INDHUMATHI M**

**ABSTRACT**

Periapical X-rays show the entire tooth, from the exposed crown to the end of the root and the bones that support the tooth. These X-rays are used to find dental problems below the gum line or in the jaw, such as impacted teeth, tooth fractures, abscesses, tumours and bone changes linked to some diseases. An Image procedure method is a method that is beneficial in involving direct observation of the patient diagnosis of determination. Another aspect of dental imaging is that it can be helpful in the field of biometrics. Human dental image analysis is a challenging and time-consuming process due to the unspecified and uneven structures of various teeth, and hence the manual investigation of dental abnormalities is at par excellence. However, automation in the domain of dental image segmentation and examination is essentially the need of the hour in order to ensure error-free diagnosis and better treatment planning. The radiographic features of periapical inflammatory lesions vary depending on the time course of the lesion. Because very early lesions may not show any radiographic changes, diagnosis of these lesions relies solely on the clinical symptoms that describes the innovative solution which provides efficient disease detection and deep learning with convolutional neural networks (CNN’s) has achieved great success in the classification of various dental diseases. A variety of neuron-wise and layer-wise visualization methods were applied using a CNN, trained with a publicly available dental disease given image dataset. So, it observed that neural networks can capture the colors and textures of lesions specific to respective diseases upon diagnosis, which resembles human decision-making.

**TABLE OF CONTENTS**

**CHAPTER NO. TITLE PAGE NO.**

**ABSTRACT** vi

**LIST OF TABLES** vii

**LIST OF FIGURES** ix

**LIST OF SYMBOLS, ABBREVIATIONS** x

**1. INTRODUCTION 1**

1.1. Overview 2

1.2. Problem Definition 3

**2. LITERATURE SURVEY 4**

**3. SYSTEM ANALYSIS 9**

3.1. Existing System 10

3.2. Proposed system 11

3.3. Feasibility Study 11

3.4. Hardware Environment 13

3.5. Software Environment 13

**4. SYSTEM DESIGN 14**

4.1. Data Flow Diagram 15

4.2. UML Diagrams 17

**5. SYSTEM ARCHITECTURE 21**

5.1. System overview 22

5.2. Module Design Specification 23

5.3. Algorithms 26

**6. SYSTEM IMPLEMENTATION**30

6.1. Coding 31

**7. PERFORMANCE ANALYSIS 42**

7.1.Performance Analysis 43

7.2. Results and Discussion 45

**8. CONCLUSION 46**

8.1. Conclusion and Future Enhancements 47

**APPENDICES** 48

A.1 Sample Screens 48

**REFERENCES** 51

**LIST OF FIGURES**

**FIGURE NO. TITLE PAGE.NO**

4.1. Dataflow Diagram 15

4.2. Usecase Diagram 17

4.3. Class Diagram 18

4.4. Activity Diagram 19

4.5. Sequence Diagram 20

4.7. System Architecture 22

**LIST OF SYMBOLS**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.NO** | **NOTATION**  **NAME** | **NOTATION** | **DESCRIPTION** |
| 1. | Class | *+ public*  *- private*  *# protected*  *Class Name*  *-attribute*  *-attribute*  *+operation*  *+operation*  *+operation* | Represents a collection of similar entities grouped together. |
| 2. | Association | nAME  Class B  Class A    Class A  Class B | Associations represents static relationships between classes. Roles represents the way the two classes see each other. |
| 3. | Actor | Class A  Class A  Class B  Class B | It aggregates several classes into a single classes. |
| 4. | Aggregation | Interaction between the system and external environment |
| 5. | Relation  (uses) | uses | Used for additional process communication. |
| 6. | Relation  (extends) | EXTENDS | Extends relationship is used when one use case is similar to another use case but does a bit more. |

|  |  |  |  |
| --- | --- | --- | --- |
| 7. | Communication |  | Communication between various use cases. |
| 8. | State | State | State of the process. |
| 9. | Initial State |  | Initial state of the object |
| 10. | Final state |  | Final state of the object |
| 11. | Control flow |  | Represents various control flow between the states. |
| 12. | Decision box |  | Represents decision making process from a constraint |
| 13. | Usecase |  | Interact ion between the system and external environment. |

|  |  |  |  |
| --- | --- | --- | --- |
| 14. | Component |  | Represents physical modules which  is a collection of components. |
| 15. | Node |  | Represents physical modules which  are a collection of components. |
| 16. | Data Process/State |  | A circle in DFD represents a state or process which has been triggered due to some event or action. |
| 17. | External entity |  | Represents external entities such as keyboard, sensors, etc. |
| 18. | Transition |  | Represents communication that occurs between processes. |
| 19. | Object Lifeline |  | Represents the vertical dimensions that the object communications. |
| 20. | Message |  | Represents the message exchanged. |

**LIST OF ABBREVIATION**

DFD – Data Flow Diagram

CNN – Convolutional Neural Network

**CHAPTER-1**

**INTRODUCTION**

**CHAPTER-1**

**INTRODUCTION**

* 1. **OVERVIEW**

Periapical X-rays are used to find dental problems below the gum line or in the jaw, similar as impacted teeth, tooth fractures, abscesses, tumours and bone changes linked to some conditions. The radiographic features of seditious lesions vary depending on the time course of the lesion. Because veritably early lesions may not show any radiographic changes, opinion of these lesions relies solely on the clinical symptoms. It is observed that neural networks can capture the colors and textures of lesions specific to separate conditions upon opinion, which resembles mortal decision-making. to classify dental Disease Classification over static facial images using deep learning ways was developed. This is a complex problem that has formerly been approached several times with different ways. While good results have been achieved using feature engineering, this project concentrated on feature learning, which is one of DL promises. Also feature engineering is not necessary, image pre-processing boosts classification accuracy. Then, it reduces noise on the input data. Currently, Agriculture grounded AI dental disease includes is heavily needed. The result completely based on feature learning does not seem close yet because of a major limitation. Therefore, Disease classification could be achieved by means of deep learning techniques. A variety of neuron-wise and layer-wise visualization methods were applied using a CNN, trained with a publicly available dental disease given image dataset. So, it observed that neural networks can capture the colors and textures of lesions specific to respective diseases upon diagnosis, which resembles human decision-making.

* 1. **PROBLEM DEFINITON**

The thing is to develop a deep learning model for dental disease image classification by convolutional neural network algorithm for potentially classifying the results in the form of best accuracy by comparing the CNN infrastructure. Utmost of the affiliated works classifies the condition with three classifiers, i.e. Primary Endo with Secondary Perio, Primary Periodontal Lesion, True Combined Lesions. Whether, the scope in our exploration is to resolve this classes of disfigurement. To classify dental Disease Classification over static facial images using deep learning ways was developed. This is a complex problem that has formerly been approached several times with different ways. While good results have been achieved using feature engineering, this design concentrated on feature learning, which is one of DL pledges. The result completely grounded on feature learning does not feel close yet because of a major limitation. Therefore, Disease classification could be achieved by means of deep learning techniques.

**CHAPTER-2**

**LITERATURE SURVEY**

**CHAPTER-2**

**LITERATURE SURVEY**

**1). Sindu Divakaran, Suja D, “Classification of Digital Dental X-ray Images**

**UsingMachine Learning” ,2021, IEEE,DOI:10.1109.[15]**

Dental conditions like dental anomalies, periapical  and dental caries is adding day by day in children and  adults. Artificial intelligence and neural network with its application in medical imaging is impacting the healthcare industry. Application of the fundamentals of machine learning in  dental imaging is making it easy to segment and classify  images .Based on the efficiency and performance of  classification algorithms, many of the promising ones like SVM, ANN, KNN are applied on dental data set .

**2). Amaar Obaid Hassan, Gregory Y. H. Lip, “Acute Dental Periapical Abscess**

**and New-Onset Atrial Fibrillation: A Nationwide, Population-Based Cohort**

**Study”,2021, MDPI ,DOI:10.3390.[2]**

There are limited data on the relationship of acute dental infections with hospitalization and new-onset atrial fibrillation (AF). This study aims to assess the relationship that is between acute periapical abscess and incident AF. This was a retrospective cohort study from a French national database of cases rehabilitated in 2013 (3.4 million patients) with at least five years of follow up. In total, 3,056,291 grown-ups (55.1% lady) needed hospital admission in French hospitals in 2013 while not having a history of AF. Of 4693 cases classified as having dental periapical abscess, 435 (9.27%) developed AF, contrast to 326,241 (10.69%) without dental periapical abscess that developed AF over a mean follow-up of 4.8 ± 1.7 years.

**3). Anuj Kumar, “Descriptive analysis of dental X-ray images using various**

**practical methods” ,2021,** **PeerJ, DOI:10.7717.[3]**

In dentistry, interpreters interpret numerous dental X-ray imaging modalities to identify tooth-related problems, abnormalities, or teeth structure changes. Another aspect of dental imaging is that it can be helpful in the plot of biometrics. Human dental image analysis is a challenging and time-consuming process due to the unidentified and uneven structures of various teeth, and hence the manual disquisition of dental abnormalities is at par excellence.

**4). Kasra KARAMIFAR, Afsoon TONDARI, Mohammad Ali SAGHIRI,**

**“Endodontic Periapical Lesion: An Overview on the Etiology, Diagnosis and**

**Current Treatment Modalities”, 2020, EUROPEAN ENDODONTIC**

**DOI:10.14744.[7]**

Nonsurgical and surgical endodontic treatments have a highest success rate in the treatment and prevention of apical periodontitis when carried out according to standard and accepted clinical principles. Nevertheless, endodontic periapical lesions remain in some cases.

**5). Vicente Rueda, Ulises Velazquez, “Using the Periapical Index to evaluate**

**the healing of periapical lesions after root canal treatment”,2020,** **Giornale**

**Italiano di Endodnzia DOI:10.32067.[17]**

Root canal treatment serves to help or cure periapical periodontitis. The aim of our study was to evaluate the remission of periapical lesions radiographically in patients who had experienced root canal treatment.

**6). Parkavi. B, Roshini. T.J, Maheswari. M, “Segmentation of Periapical Dental**

**X-Ray Images by Applying Morphological Operation”, 2019, IFERP,Volume 5**

Segmentation of Dental X-ray pictures is done by using various image processing ways that are helpful in medical diagnosis, clinical functions and time period operations. These ways aim to outline the segmentation of various tooth structures present within the Dental X-ray pictures which can be used for the first discovery of decay, tooth fractures,  passage treatment.

**7). Jiafa Mao , Kaihui Wang, Yahong Hu, Weiguo Sheng, Qixin Feng,**

**“Grabcut Algorithm for dental X-RAY images based on full threshold**

**Segmentation”, 2018, IET Volume 12.[6]**

Teeth are delicate to be destroyed due to their corrosion resistance, high melting point and hardness. Dental biometrics can thus give assistance in human forensic identification, especially to the unknown corpses. One of the key issue in dental based Mortal identification is the segmentation of Dental X-ray images. In this paper, a new segmentation algorithm has been proposed for this purpose.

**8). Sibel Koçak, Mustafa Murat, “Periapical Health Related to the Quality of**

**Coronal Restorations and Root Fillings in a Turkish Population” , 2013,**

**CumhuriyetDent J DOI:10.7126.[14]**

The aim of the present study was to estimate the quality of root canal treatments and coronal restorations investigating their influence on the periapical status of endodontically-treated teeth in a Turkish population grounded on radiographic.

**9). Abdolvahab Ehsani Rad, Rosely KumoiMohd Shafry Mohd Rahim, Alireza**

**Norouzi , “ Dental X-Ray Image Segmentation and Multiple Feature**

**Extraction”, 2012, AWERProcedia, DOI:10.1340.[1]**

The thing of this research is to automate the process of representation and extracting textural features of dental x-ray images to use in further operations. In order to gain this aim we need to automate the process of the dental x-ray images segmentation and distinguish the teeth from background and other tissues. The segmentation of dental x-ray images could be delicate due to the shape variation and intensity variation within the same dental x-ray images and from one image to another. There are enormous researches about dental image analysis systems but it has to be estimated to find the appropriate method.

**10). P.N.R. Nair, “Pathogenesis of apical periodontities and the causes of**

**Endodontic Failures” ,2004, [11].**

Apical periodontitis is a effect to endodontic infection and manifests itself as the host defense response to microbial challenge expiring from the root canal system. It is viewed as a dynamic hassle between microbial factors and host defenses at the interface between infected radicular pulp and periodontal ligament that results in original inflammation, resorption of hard tissues, destruction of other periapical tissues, and eventual formation of various histopathological orders of apical periodontitis, generally referred to as periapical lesions. The treatment of apical periodontitis, as a disease of root canal infection, consists of eradicating microbes or mainly reducing the microbial load from the root canal and preventing reinfection by orthograde root stuffing.

### 

### CHAPTER-3

### SYSTEM ANALYSIS

### CHAPTER-3

### 

### SYSTEM ANALYSIS

### 3.1. EXISTING SYSTEM

Dental conditions like dental anomalies, periapical and dental caries is adding day by day in children and grown-ups. Artificial intelligence and neural network with its application in medical imaging is impacting the healthcare industry. X –ray imaging is the generally employed technique to diagnose condition of the teeth. Segmentation of differing dental anomalies using neural network is proving to be a boon to the dental field. Application of neural algorithms aids in carrying images with better discovery accuracy.It describes the experimental results of the classification methods using digital dental X ray images for different classes of dental conditions. Dental images, belonging to various dental conditions like vertical impaction, periapical abscess, distal pulp horn caries, missed canal in root canal, etc are stored in the dental image database Automated discovery reduces the workload of a dentist with classification being accurate. In this work, it is suggested that by exercising GLCM features and SVM, KNN and classifiers, the teeth affected by dental caries can be set apart from the normal teeth in a more detailed manner.

**Drawback**

* It has not concentrated on identifying other dental conditions with CNN as classifier.
* It has not concentrated on increasing the recognition.

### 3.2. PROPOSED SYSTEM

To classify the dental condition. We planned to design deep learning fashion so that a normal person can also be suitable to use it fluently. It proposed system to analyze dental conditions. It tells about the experimental analysis of our methodology. Different number of images is collected for each conditions that was classified into database images and input images. The primary attributes of the image are relied upon the shape and texture acquainted features. The sample screenshots displays the dental disease discovery using color grounded segmentation model.

**Advantages**

* Reducing subjectiveness arising from human experts in detecting the dental conditions.
* It is essential to detect a particular disease. In our country, numerous are not so educated to get correct information about all conditions.

**3.3. FEASIBILITY STUDY**

**1. TECHNICAL FEASIBILITY**

**PYTHON**

While complex algorithms and versatile workflows stand behind ML and AI, Python's simplicity allows developers to write reliable systems. Developers get to put all their effort into solving an ML issues instead of focusing on the technical nuances of the language.

**2. ECONOMIC FEASIBILITY**

Economical Feasibility study using COCOMO model was carried out to find the cost/ benefits analysis of the project .

Advanced COCOMO model with organic, semi-detached categories of software project modelling is used.

|  |  |  |  |
| --- | --- | --- | --- |
| **MODULE** | **CATEGORY** | **EFFORT (PM)** | **DEVELOPMENT TIME (Months)** |
| MANUAL NET | SEMI**-**DETACHED | 3.0(KLOC)1.12 | 2.5(Effort)0.35 |
| ALEXNET | ORGANIC | 2.4(KLOC)1.05 | 2.5(Effort)0.38 |
| LENET | ORGANIC | 2.4(KLOC)1.05 | 2.5(Effort)0.38 |

**Manual Net**

Effort Adjustment Factor EAF = 1

**Estimation of development effort**:

Effort = 3.0(KLOC)1.12 PM = 3 x 0.3583 = 1.07 PM

**Estimation of development time:**

Tdev = 2.5(Effort)0.35 Months = 2.5 x 1.025 = 2.56 Months

**Alexnet**

**Estimation of development effort:**

Effort = 2.4(KLOC)1.05 PM = 2.4(0.15)1.05 = 2.4 x 0.1575 =0.38 PM

**Estimation of development time:**

Tdev = 2.5(Effort)0.38 Months = 2.5(0.38)0.38 = 2.5 x 0.1444 =0.37 Months

**Lenet**

**Estimation of development effort:**

Effort = 2.4(KLOC)1.05 PM = 2.4(0.15)1.05 = 2.4 x 0.1575 =0.38 PM

**Estimation of development time:**

Tdev = 2.5(Effort)0.38 Months = 2.5(0.38)0.38 = 2.5 x 0.1444 =0.37 Months

**Total effort = 1.07+0.38+0.38 = 1.83 PM**

**Total development time = 2.56+0.37+0.37 = 3.3 Months**

**3. SOCIAL FEASIBILITY**

Ithelps the people with easy identification of stage of disease. It takes less time and it is inexpensive. Because of early prediction it is highly useful for people. So our project is socially feasible.

**3.4. HARDWARE REQUIREMENTS**

Processor : Intel i5

Hard disk : Minimum 80 GB

RAM : Minimum 2 GB

**3.5. SOFTWARE REQUIREMENTS**

Operating System : Windows

Development Tool : Anaconda with Jupyter Notebook

**CHAPTER-4**

**SYSTEM DESIGN**

**CHAPTER-4**

**SYSTEM DESIGN**

**4.1. DATAFLOW DIAGRAM**

A data flow diagram (DFD) is a graphical representation of the "inflow" of data through an information system, modeling its process aspects. A DFD is frequently used as a primary step to create an overview of the system without going into great detail, which can latterly be developed.

Image

Disease Classification

**FIG 4.1. Level 0 DFD Diagram of Dental Disease Classification**

Test image

dental disease prediction

Feature extraction

Training dataset

**FIG 4.2. Level 1 DFD Diagram of Dental Disease Classification**

Testing dataset

Training dataset

**FIG 4.3. Level 2 DFD Diagram of Dental Disease Classification**

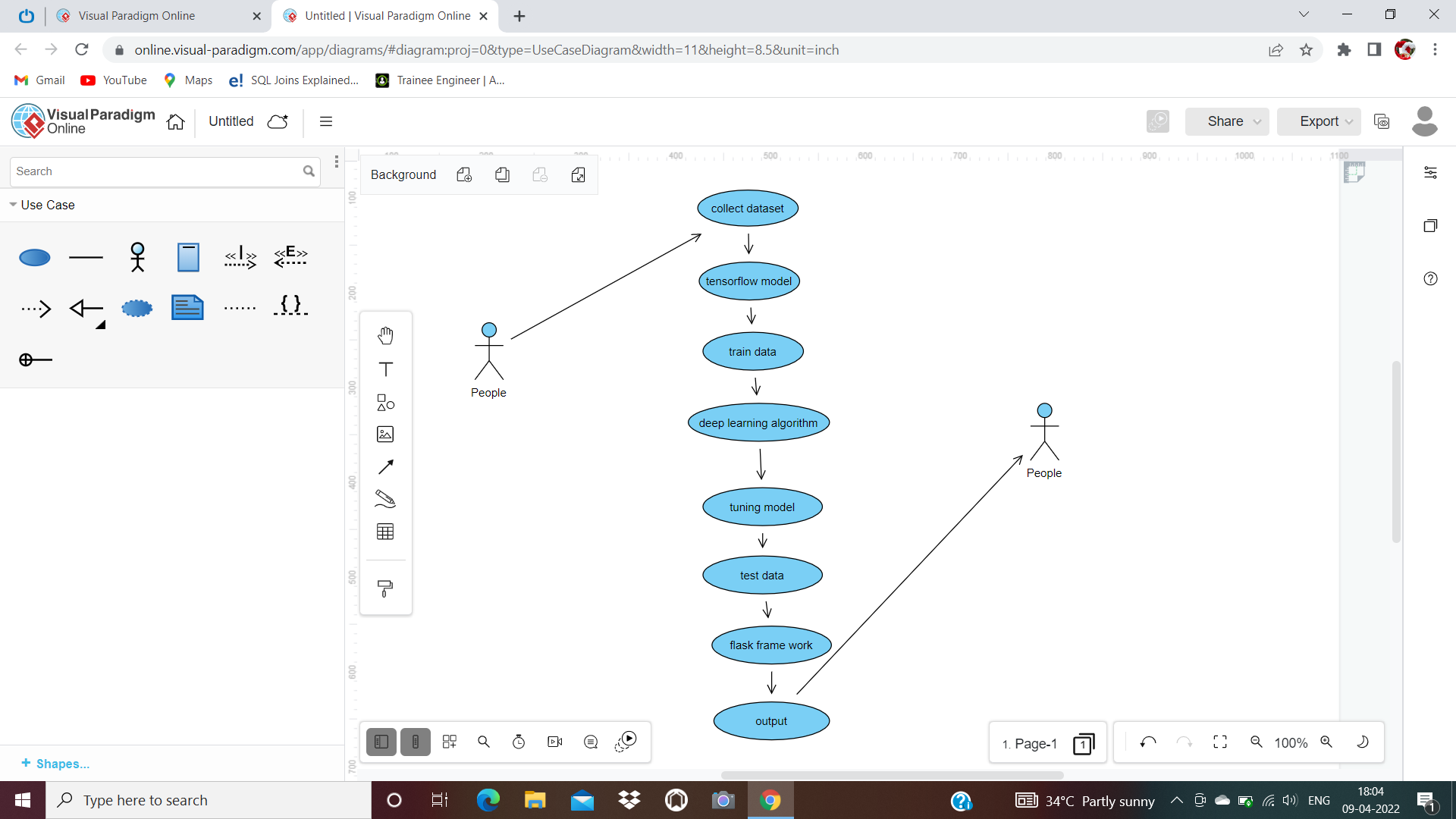
Classified dental disease

dental Feature

**FIG 4.4. Level 3 DFD Diagram of Dental Disease Classification**

**4.2. USECASE DIAGRAM**

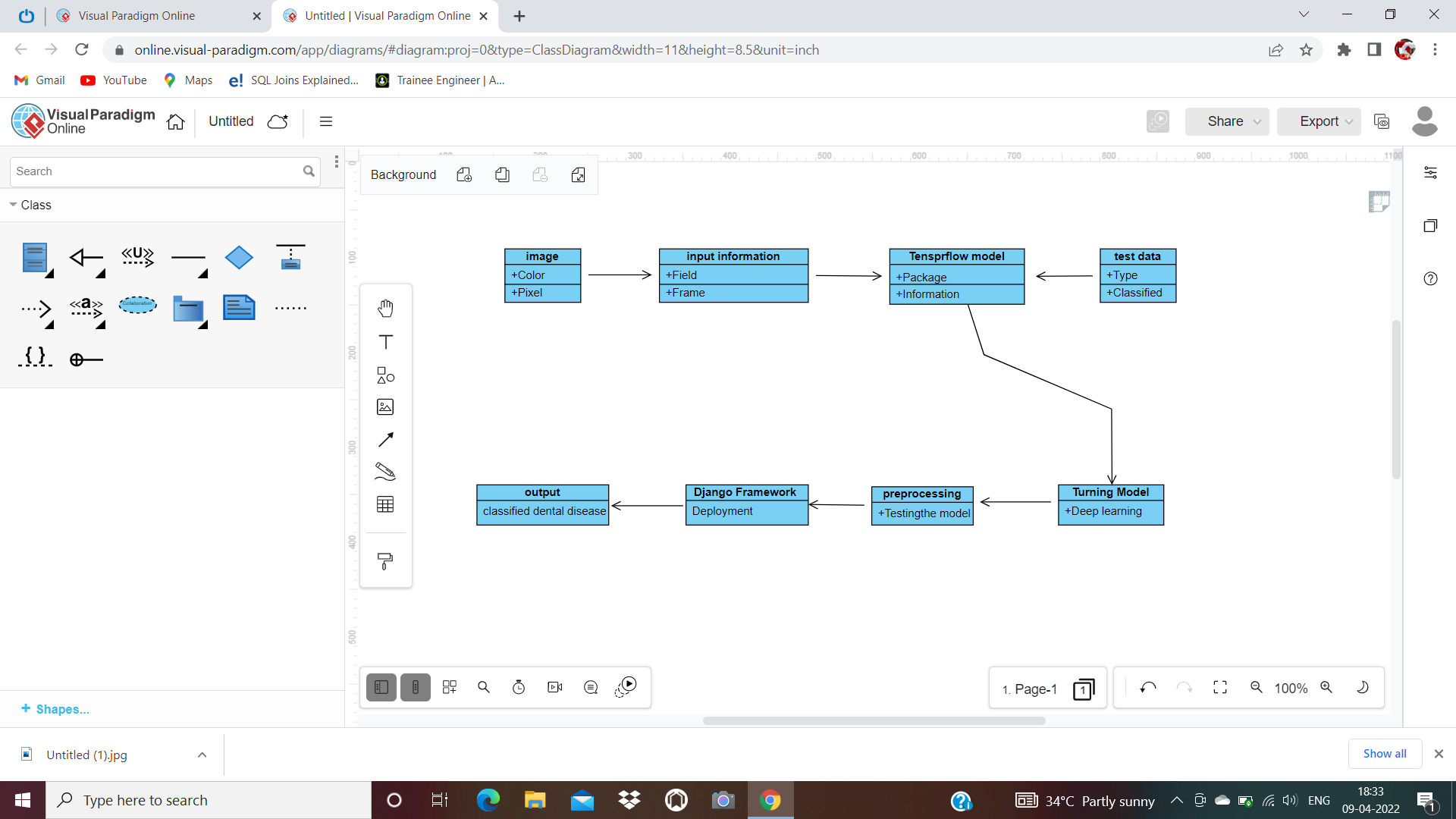
Use case diagrams are considered for high position requirement analysis of a system. So when the conditions of a system are analyzed the functionalities are captured in use cases. So, it can say that uses cases are nothing but the system functionalities written in an systematized manner.



**FIG 4.5. USECASE Diagram of Dental Disease Classification**

**4.3. CLASS DIAGRAM**

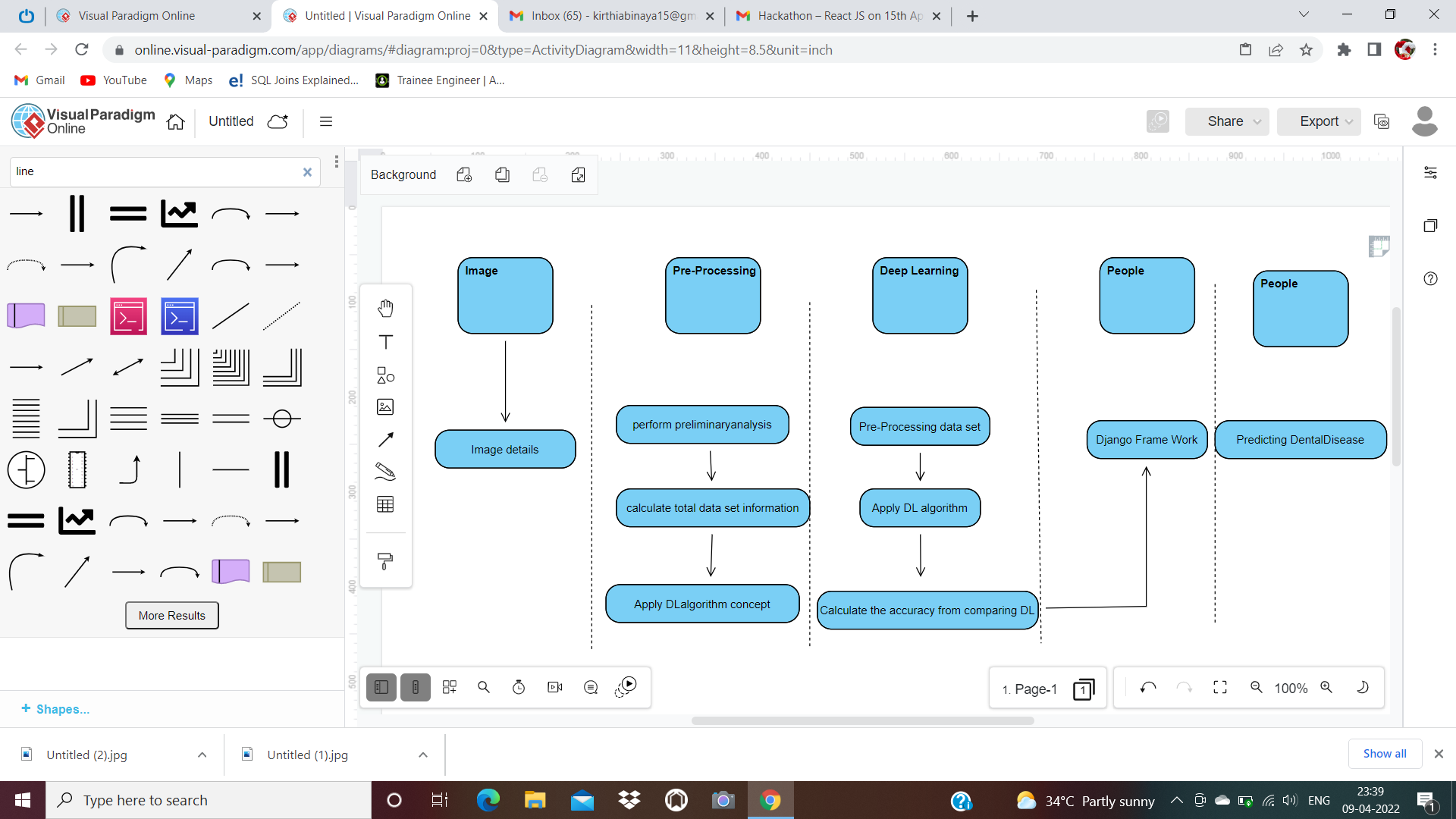
Class diagram is basically a graphical representation of the stationary view of the system and represents different aspects of the operation. So a collection of class illustration represent the whole system. The name of the class illustration should be meaningful to describe the aspect of the system. Each element and their connections should be identified in advance Responsibility (attributes and methods) of each class should be easily identified for each class minimal number of properties should be specified and because, unnecessary properties will make the illustration complicated.



**FIG 4.6. CLASS Diagram of Dental Disease Classification**

**4.4. ACTIVITY DIAGRAM**

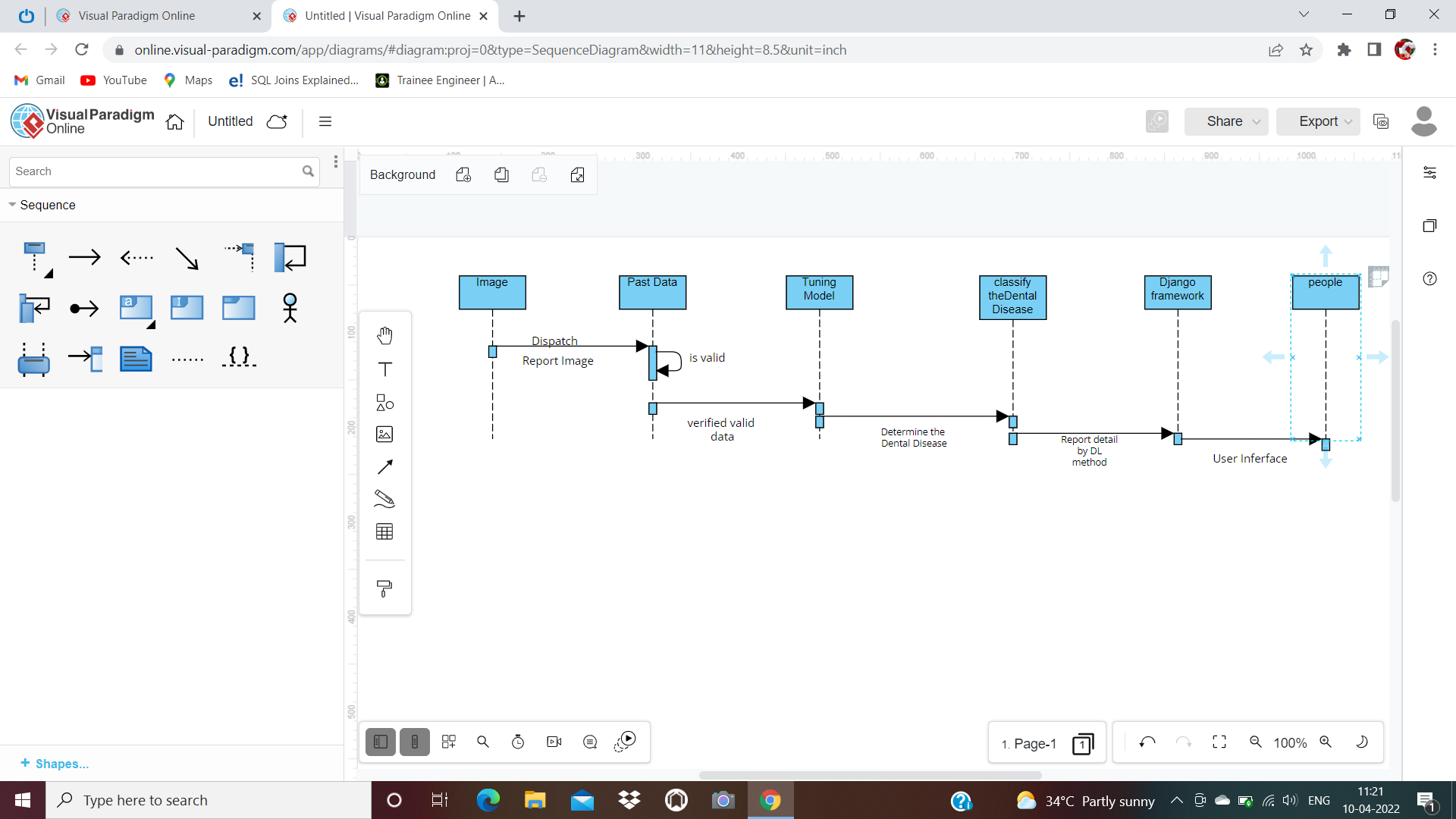
Activity is a particular operation of the system. Activity diagrams are not only used for imaging dynamic nature of a system but they are also used to construct the executable system by using forward and reverse engineering ways. The only missing thing in activity diagram is communication part. It does not show any communication inflow from one activity to another.



**FIG 4.7. ACTIVITY Diagram of Dental Disease Classification**

**4.5. SEQUENCE DIAGRAM**

Sequence diagrams model the inflow of logic within your system in a visual manner, enabling you both to document and validate your logic, and are commonly used for both analysis and design purposes. Sequence diagrams are the most popular UML artifact for dynamic modelling, which focuses on identifying the behaviour within your system.



**FIG 4.8. SEQUENCE Diagram of Dental Disease Classification**

**CHAPTER-5**

**SYSTEM ARCHITECTURE**

**CHAPTER-5**

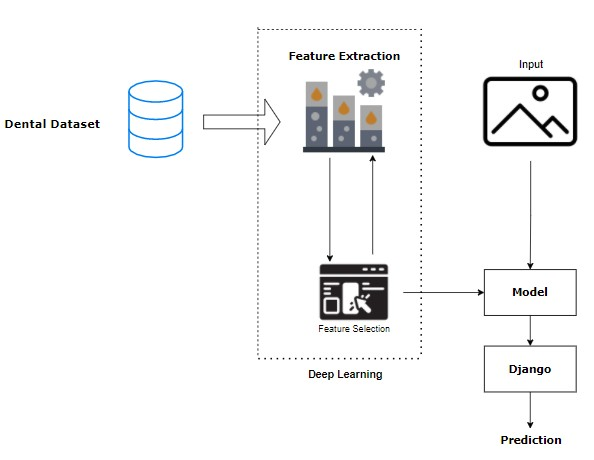
**SYSTEM ARCHITECTURE**

**5.1. SYSTEM OVERVIEW**

In the System Architecture the input will be given in the form of X-Ray images and by collecting the data from the dataset.

Then, with the help of Feature Extraction and by Future Selection Model on Deep Learning the machine will be trained with these models.

And by the Django Framework we are predicting the results whether the patient is affected by the disease or not.



**FIG 5.1. System Architecture of Dental Disease Classification**

**5.2. MODULE DESCRIPTION**

**LIST OF MODULES**

1. Import The Given Image From Dataset

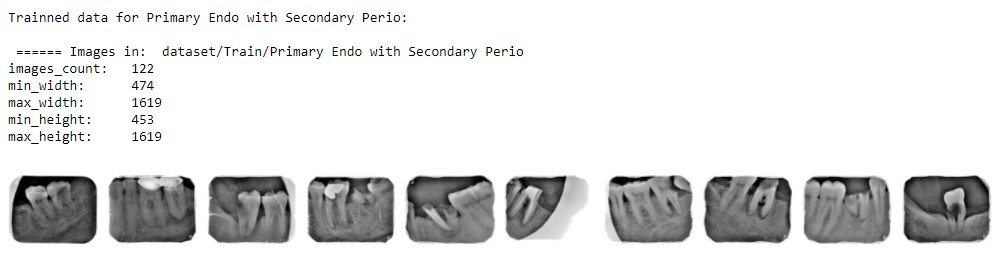
2. To Train by Given Image Dataset

3. Dental Disease Classifications

**1. Import The Given Image From Dataset**

## We have to import our data set using keras preprocessing image data creator function also we create size, rescale, range, zoom range, horizontal flip. Also we import our image dataset from folder through the data creator function. Then we set train, test, and confirmation also we set target size, batch size and class-mode from this function we have to train using our own created network by adding layers of CNN.

**Primary Endo with Secondary Perio**



**FIG 5.1. Primary Endo with Secondary Perio**

**Primary Periodontal Lesion**



**FIG 5.2. Primary Periodontal Lesion**

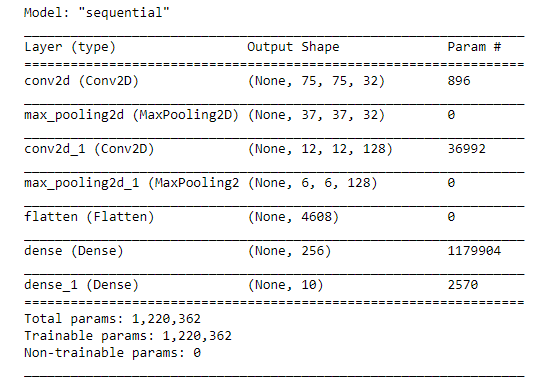
**True Combined Lesions**



**FIG 5.3. True Combined Lesions**

**2. To Train By Given Image Dataset**

To train our dataset using classifier and fit creator function also we make training way per epoch’s also total number of epochs, validation data and confirmation way using this data we can train our dataset.

****

**FIG 5.4. To Train By Given Image Dataset**

**3. DENTAL DISEASE CLASSIFICATIONS**

We give input as figure using keras preprocessing package. That input Image converted into array value using pillow and image to array function package. We have formerly classified dental in our dataset. It classifies what are Disease. Also we have to predict our Disease using predict function.

Given dataset

CNN

Model

Feature Extractions

dental Disease Classification

Input image

**FIG 5.5.Dental Disease Classification**

The dental recognition method is grounded on a two-channel architecture that is suitable to recognize. The dental parts are cropped and uprooted and also used as the input into the inception layer of the CNN. The Training phase involves the feature extraction and classification using CNN.

**5.3. ALGORITHM**

A Convolutional Neural Network (CNN) is a DL algorithm which can take in an input image, assign significance (learnable weights and impulses) to various aspects/objects in the image and be suitable to differentiate one from the other.

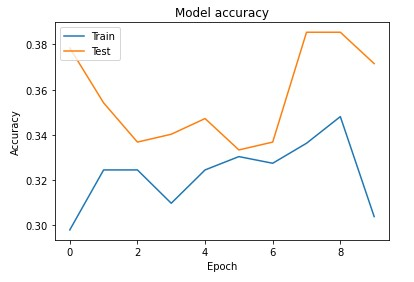
The pre-processing needed in a CNN is much lower as compared to other classification algorithms. While in primitive styles filters are hand-engineered, with enough training, CNN have the ability to learn these filters/characteristics. The architecture of a CNN is similar to that of the connectivity pattern of Neurons in the Mortal Brain and was inspired by the association of the Visual Cortex. Individual neurons respond to stimulants only in a confined region of the visual field known as the open Field. Their network consists of four layers with 1,024 input units, 256 units in the first hidden layer, eight units in the second hidden subcaste, and two output units.

**1. Input Layer**

Input layer in CNN contain image data. Figure data is represented by three dimensional matrixes. It needs to be reshaped it into a single column. Suppose you have image of dimension 28 x 28 =784, it need to convert it into 784 x 1 before feeding into input.

## 2. Convo Layer

Convo layer is occasionally called feature extractor subcaste because features of the image are get uprooted within this layer. First of all, a part of image is connected to Convo layer to perform complication operation as we saw before and calculating the dot product between open field (it is a original region of the input image that has the same size as that of filter) and the filter. Result of the operation is single integer of output volume. Also the filter over the coming open field of the same input image by a Stride and do the same operation again. It will repeat the same process again and again until it goes through the whole image. The output will be the input for the coming layer.



**FIG 5.6. CNN model trained dataset accuracy**

## 3. Pooling Layer

## Pooling layer is used to reduce the spatial volume of input image after complication. It is used between two Convo layers. If it applies FC after Convo layer without applying pooling or maximum pooling, then it will be computationally expensive. So, the maximum pooling is only way to reduce the spatial volume of input image. It has applied maximum pooling in single depth slice with Stride of 2. It can observe the 4 x 4 dimension input will going to reduce to 2 x 2 dimensions.

## 4. Fully Connected Layer (FC)

Fully connected layer involves weights, impulses, and neurons. It connects

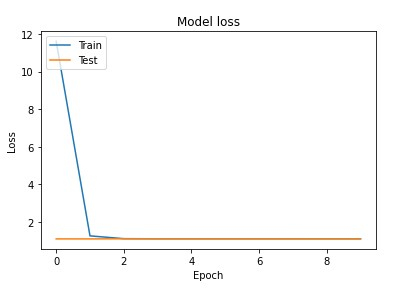
neurons in one subcaste to neurons in another subcaste. It is used to classify images

between different orders by training.

## 5. Softmax / Logistic Layer

Softmax or Logistic layer is the last subcaste of CNN. It resides at the end of FC

subcaste. Logistic is used for double classification and softmax.



**FIG 5.7. CNN model trained dataset loss values**

## 6. Output Layer

Output layer contains the marker which is in the form of one-hot encoded. Now

you have a well understanding of CNN.

**CHAPTER-6**

## SYSTEM IMPLEMENTATION

**CHAPTER-6**

**SYSTEM IMPLEMENTATION**

**6.1. CODING**

**MODULE – 1**

**import** os

**import** numpy **as** np *# linear algebra*

**import** matplotlib.pyplot **as** plt

*# Dl framwork - tensorflow, keras a backend*

**import** tensorflow **as** tf

**import** tensorflow.keras.backend **as** K

**from** tensorflow.keras.models **import** Model, Sequential

**from** tensorflow.keras.layers **import** Input, Dense, Flatten, Dropout, BatchNormalization

**from** tensorflow.keras.layers **import** Conv2D, SeparableConv2D, MaxPool2D, LeakyReLU, Activation

**from** tensorflow.keras.optimizers **import** Adam

**from** tensorflow.keras.preprocessing.image **import** ImageDataGenerator

**from** tensorflow.keras.callbacks **import** ModelCheckpoint, ReduceLROnPlateau, EarlyStopping

**from** IPython.display **import** display

**from** os **import** listdir

**from** os.path **import** isfile, join

**from** PIL **import** Image

**import** glob

**from** tensorflow.keras.preprocessing.image **import** ImageDataGenerator

**from** tensorflow.keras.layers **import** Convolution2D

**from** tensorflow.keras.layers **import** MaxPooling2D

**from** tensorflow.keras.layers **import** Flatten

**from** tensorflow.keras.layers **import** Dense

**import** warnings

warnings**.**filterwarnings('ignore')

dir\_name\_train\_Primary\_Endo\_with\_Secondary\_Perio **=** 'dataset/Train/Primary Endo with Secondary Perio'

dir\_name\_train\_Primary\_Periodontal\_Lesion **=** 'dataset/Train/Primary Periodontal Lesion'

dir\_name\_train\_True\_Combined\_Lesions **=** 'dataset/Train/True Combined Lesions'

**def** plot\_images(item\_dir, n**=**6):

all\_item\_dir **=** os**.**listdir(item\_dir)

item\_files **=** [os**.**path**.**join(item\_dir, file) **for** file **in** all\_item\_dir][:n]

plt**.**figure(figsize**=**(80, 40))

**for** idx, img\_path **in** enumerate(item\_files):

plt**.**subplot(7, n, idx**+**1)

img **=** plt**.**imread(img\_path)

plt**.**imshow(img, cmap**=**'gray')

plt**.**axis('off')

plt**.**tight\_layout()

**def** Images\_details\_Print\_data(data, path):

print(" ====== Images in: ", path)

**for** k, v **in** data**.**items():

print("%s:\t%s" **%** (k, v))

**def** Images\_details(path):

files **=** [f **for** f **in** glob**.**glob(path **+** "\*\*/\*.\*", recursive**=True**)]

data **=** {}

data['images\_count'] **=** len(files)

data['min\_width'] **=** 10**\*\***100 *# No image will be bigger than that*

data['max\_width'] **=** 0

data['min\_height'] **=** 10**\*\***100 *# No image will be bigger than that*

data['max\_height'] **=** 0

**for** f **in** files:

im **=** Image**.**open(f)

width, height **=** im**.**size

data['min\_width'] **=** min(width, data['min\_width'])

data['max\_width'] **=** max(width, data['max\_height'])

data['min\_height'] **=** min(height, data['min\_height'])

data['max\_height'] **=** max(height, data['max\_height'])

Images\_details\_Print\_data(data, path)

print("")

print("Trainned data for Primary Endo with Secondary Perio:")

print("")

Images\_details(dir\_name\_train\_Primary\_Endo\_with\_Secondary\_Perio)

print("")

plot\_images(dir\_name\_train\_Primary\_Endo\_with\_Secondary\_Perio, 10)

print("")

print("Trainned data for Primary Periodontal Lesion:")

print("")

Images\_details(dir\_name\_train\_Primary\_Periodontal\_Lesion)

print("")

plot\_images(dir\_name\_train\_Primary\_Periodontal\_Lesion, 10)

print("")

print("Trainned data for True Combined Lesions:")

print("")

Images\_details(dir\_name\_train\_True\_Combined\_Lesions)

print("")

plot\_images(dir\_name\_train\_True\_Combined\_Lesions, 10)

Classifier**=**Sequential()

Classifier**.**add(Convolution2D(32,(3,3),input\_shape**=**(128,128,3),activation**=**'relu'))

Classifier**.**add(MaxPooling2D(pool\_size**=**(2,2)))

Classifier**.**add(Flatten())

Classifier**.**add(Dense(38, activation**=**'relu'))

Classifier**.**add(Dense(3, activation**=**'softmax'))

Classifier**.**compile(optimizer**=**'rmsprop',loss**=**'categorical\_crossentropy',metrics**=**['accuracy'])

train\_datagen**=**ImageDataGenerator(rescale**=**1.**/**255,shear\_range**=**0.2,zoom\_range**=**0.2,horizontal\_flip**=True**)

test\_datagen**=**ImageDataGenerator(rescale**=**1.**/**255)

training\_set**=**train\_datagen**.**flow\_from\_directory('dataset/Train',target\_size**=**(128,128),batch\_size**=**32,class\_mode**=**'categorical')

test\_set**=**test\_datagen**.**flow\_from\_directory('dataset/Test',target\_size**=**(128,128),batch\_size**=**32,class\_mode**=**'categorical')

epochs **=** 10

batch\_size **=** 32

*#### Fitting the model*

history **=** Classifier**.**fit\_generator(

training\_set, steps\_per\_epoch**=**training\_set**.**samples **//** batch\_size,

epochs**=**epochs,

validation\_data**=**test\_set,validation\_steps**=**test\_set**.**samples **//** batch\_size)

**def** graph():

*#Plot training & validation accuracy values*

plt**.**plot(history**.**history['accuracy'])

plt**.**plot(history**.**history['val\_accuracy'])

plt**.**title('Model accuracy')

plt**.**ylabel('Accuracy')

plt**.**xlabel('Epoch')

plt**.**legend(['Train', 'Test'], loc**=**'upper left')

plt**.**show()

*# Plot training & validation loss values*

plt**.**plot(history**.**history['loss'])

plt**.**plot(history**.**history['val\_loss'])

plt**.**title('Model loss')

plt**.**ylabel('Loss')

plt**.**xlabel('Epoch')

plt**.**legend(['Train', 'Test'], loc**=**'upper left')

plt**.**show()

graph()

**MODULE - 2**

*# Dl framwork - tensorflow, keras a backend*

**import** tensorflow **as** tf

**import** tensorflow.keras.backend **as** K

**from** tensorflow.keras.models **import** Model

**from** tensorflow.keras.models **import** Sequential

**from** tensorflow.keras.layers **import** Input

**from** tensorflow.keras.layers **import** Dense

**from** tensorflow.keras.layers **import** Flatten

**from** tensorflow.keras.layers **import** Conv2D

**from** tensorflow.keras.layers **import** MaxPooling2D

**from** tensorflow.keras.layers **import** Dropout

**from** tensorflow.keras.layers **import** LeakyReLU

**from** tensorflow.keras.layers **import** Activation

**from** tensorflow.keras.optimizers **import** Adam

**from** tensorflow.keras.preprocessing.image **import** ImageDataGenerator

**from** tensorflow.keras.callbacks **import** ModelCheckpoint

**from** tensorflow.keras.callbacks **import** ReduceLROnPlateau

**from** tensorflow.keras.callbacks **import** EarlyStopping

**import** warnings

warnings**.**filterwarnings('ignore')

model **=** Sequential()

*# 1st Convolutional Layer*

model**.**add(Conv2D(filters**=**96, input\_shape**=**(224,224,3), kernel\_size**=**(11,11), strides**=**(4,4), padding**=**'valid'))

model**.**add(Activation('relu'))

*# Max Pooling*

model**.**add(MaxPooling2D(pool\_size**=**(2,2), strides**=**(2,2), padding**=**'valid'))

*# 2nd Convolutional Layer*

model**.**add(Conv2D(filters**=**256, kernel\_size**=**(11,11), strides**=**(1,1), padding**=**'valid'))

model**.**add(Activation('relu'))

*# Max Pooling*

model**.**add(MaxPooling2D(pool\_size**=**(2,2), strides**=**(2,2), padding**=**'valid'))

*# 3rd Convolutional Layer*

model**.**add(Conv2D(filters**=**384, kernel\_size**=**(3,3), strides**=**(1,1), padding**=**'valid'))

model**.**add(Activation('relu'))

*# 4th Convolutional Layer*

model**.**add(Conv2D(filters**=**384, kernel\_size**=**(3,3), strides**=**(1,1), padding**=**'valid'))

model**.**add(Activation('relu'))

*# 5th Convolutional Layer*

model**.**add(Conv2D(filters**=**256, kernel\_size**=**(3,3), strides**=**(1,1), padding**=**'valid'))

model**.**add(Activation('relu'))

*# Max Pooling*

model**.**add(MaxPooling2D(pool\_size**=**(2,2), strides**=**(2,2), padding**=**'valid'))

*# Passing it to a Fully Connected layer*

model**.**add(Flatten())

*# 1st Fully Connected Layer*

model**.**add(Dense(4096, input\_shape**=**(224**\***224**\***3,)))

model**.**add(Activation('relu'))

*# Add Dropout to prevent overfitting*

model**.**add(Dropout(0.4))

*# 2nd Fully Connected Layer*

model**.**add(Dense(4096))

model**.**add(Activation('relu'))

*# Add Dropout*

model**.**add(Dropout(0.4))

*# 3rd Fully Connected Layer*

model**.**add(Dense(1000))

model**.**add(Activation('relu'))

*# Add Dropout*

model**.**add(Dropout(0.4))

*# Output Layer*

model**.**add(Dense(3))

model**.**add(Activation('softmax'))

model**.**summary()

*# Compile the model*

model**.**compile(loss **=** 'categorical\_crossentropy', optimizer**=**'adam', metrics**=**['accuracy'])

train\_datagen**=**ImageDataGenerator(rescale**=**1.**/**255,shear\_range**=**0.2,zoom\_range**=**0.2,horizontal\_flip**=True**)

test\_datagen**=**ImageDataGenerator(rescale**=**1.**/**255)

training\_set**=**train\_datagen**.**flow\_from\_directory('dataset/Train',target\_size**=**(224,224),batch\_size**=**32,class\_mode**=**'categorical')

test\_set**=**test\_datagen**.**flow\_from\_directory('dataset/Test',target\_size**=**(224,224),batch\_size**=**32,class\_mode**=**'categorical')

epochs **=** 10

batch\_size **=** 32

*#### Fitting the model*

history **=** model**.**fit(

training\_set, steps\_per\_epoch**=**training\_set**.**samples **//** batch\_size,

epochs**=**epochs,

validation\_data**=**test\_set,validation\_steps**=**test\_set**.**samples **//** batch\_size)

**import** matplotlib.pyplot **as** plt

**def** graph():

*#Plot training & validation accuracy values*

plt**.**plot(history**.**history['accuracy'])

plt**.**plot(history**.**history['val\_accuracy'])

plt**.**title('Model accuracy')

plt**.**ylabel('Accuracy')

plt**.**xlabel('Epoch')

plt**.**legend(['Train', 'Test'], loc**=**'upper left')

plt**.**show()

*# Plot training & validation loss values*

plt**.**plot(history**.**history['loss'])

plt**.**plot(history**.**history['val\_loss'])

plt**.**title('Model loss')

plt**.**ylabel('Loss')

plt**.**xlabel('Epoch')

plt**.**legend(['Train', 'Test'], loc**=**'upper left')

plt**.**show()

graph()

print("[INFO] Calculating model accuracy")

scores **=** model**.**evaluate(test\_set)

print(f"Test Accuracy: {scores[1]**\***100}")

**MODULE – 3**

**from** tensorflow.keras.callbacks **import** ModelCheckpoint, ReduceLROnPlateau, EarlyStopping

**from** tensorflow.keras.models **import** Sequential

**from** tensorflow.keras.layers **import** Convolution2D

**from** tensorflow.keras.layers **import** MaxPooling2D

**from** tensorflow.keras.layers **import** Dense

**import** warnings

warnings**.**filterwarnings('ignore')

Classifier**=**Sequential()

Classifier**.**add(Convolution2D(32,3,3,input\_shape**=**(225,225,3),activation**=**'relu'))

Classifier**.**add(MaxPooling2D(pool\_size**=**(2,2)))

Classifier**.**add(Convolution2D(128,3,3,activation**=**'relu'))

Classifier**.**add(MaxPooling2D(pool\_size**=**(2,2)))

Classifier**.**add(Flatten())

Classifier**.**add(Dense(256, activation**=**'relu'))

Classifier**.**add(Dense(3, activation**=**'softmax'))

Classifier**.**compile(optimizer**=**'rmsprop',loss**=**'categorical\_crossentropy',metrics**=**['accuracy'])

Classifier**.**summary()

**from** tensorflow.keras.preprocessing.image **import** ImageDataGenerator

train\_datagen**=**ImageDataGenerator(rescale**=**1.**/**255,shear\_range**=**0.2,zoom\_range**=**0.2,horizontal\_flip**=True**)

test\_datagen**=**ImageDataGenerator(rescale**=**1.**/**255)

training\_set**=**train\_datagen**.**flow\_from\_directory('dataset/Train',target\_size**=**(225,225),batch\_size**=**32,class\_mode**=**'categorical')

test\_set**=**test\_datagen**.**flow\_from\_directory('dataset/Test',target\_size**=**(225,225),batch\_size**=**32,class\_mode**=**'categorical')

epochs **=** 20

batch\_size **=** 32

Classifier**.**fit\_generator( training\_set, steps\_per\_epoch**=**training\_set**.**samples **//** batch\_size,

epochs**=**epochs,

validation\_data**=**test\_set,validation\_steps**=**test\_set**.**samples **//** batch\_size)

**import** h5py

Classifier**.**save('dental.h5')

**from** keras.models **import** load\_model

model**=**load\_model('dental.h5')

**import** numpy **as** np

**from** tensorflow.keras.preprocessing **import** image

test\_image**=**image**.**load\_img('Primary Periodontal Lesion.jpg',target\_size**=**(225,225))

**import** matplotlib.pyplot **as** plt

img **=** plt**.**imshow(test\_image)

test\_image**=**image**.**img\_to\_array(test\_image)

test\_image**=**np**.**expand\_dims(test\_image,axis**=**0)

result**=**model**.**predict(test\_image)

result

prediction **=** result[0]

classes**=**training\_set**.**class\_indices

classes

prediction**=**list(prediction)

prediction

classes**=**['Primary Endo with Secondary Perio','Primary Periodontal Lesion','True Combined Lesions']

output**=**zip(classes,prediction)

output**=**dict(output)

output

**if** output['Primary Endo with Secondary Perio']**==**1.0 :

print('Primary Endo with Secondary Perio')

**elif** output['Primary Periodontal Lesion']**==**1.0:

print('Primary Periodontal Lesion')

**elif** output['True Combined Lesions']**==**1.0:

print("True Combined Lesions")

**PyCharm:**

**Views.py**

from django.shortcuts import render

from django.http import HttpResponseRedirect

from django.urls import reverse\_lazy

from django.views.generic import TemplateView

from employee.forms import EmployeeForm

from django.views.generic import DetailView

from employee.models import Employee

class EmployeeImage(TemplateView):

    form = EmployeeForm

    template\_name = 'emp\_image.html'

    def post(self, request, \*args, \*\*kwargs):

        form = EmployeeForm(request.POST, request.FILES)

        if form.is\_valid():

            obj = form.save()

            return HttpResponseRedirect(reverse\_lazy('emp\_image\_display', kwags={'pk': obj.id}))

        context = self.get\_context\_data(form=form)

        return self.render\_to\_response(context)

    def get(self, request, \*args, \*\*kwargs):

        return self.post(request, \*args, \*\*kwargs)

class EmpImageDisplay(DetailView):

    model = Employee

    template\_name = 'emp\_image\_display.html'

    context\_object\_name = 'emp'

def cancer(request):

    result1 = Employee.objects.latest('id')

    import numpy as np

    import tensorflow as tf

    from tensorflow import keras

    import h5py

    models = keras.models.load\_model('C:/Users/SPIRO/Desktop/own/2.WORKING/Periapical\_Xrays/Deploy/employee/dental.h5')

    from tensorflow.keras.preprocessing import image

    test\_image = image.load\_img('C:/Users/SPIRO/Desktop/own/2.WORKING/Periapical\_Xrays/Deploy/media/' + str(result1),

                                target\_size=(225, 225))

    test\_image = image.img\_to\_array(test\_image)

    test\_image = np.expand\_dims(test\_image, axis=0)

    result = models.predict(test\_image)

    prediction = result[0]

    prediction = list(prediction)

    classes=['Primary Endo with Secondary Perio','Primary Periodontal Lesion','True Combined Lesions']

    output = zip(classes, prediction)

    output = dict(output)

    if output['Primary Endo with Secondary Perio'] == 1.0:

        a = "Primary Endo with Secondary Perio"

    elif output['Primary Periodontal Lesion'] == 1.0:

        a = "Primary Periodontal Lesion"

    elif output['True Combined Lesions'] == 1.0:

        a = "True Combined Lesions"

    return render(request, "result.html", {"out": a})

**CHAPTER-7**

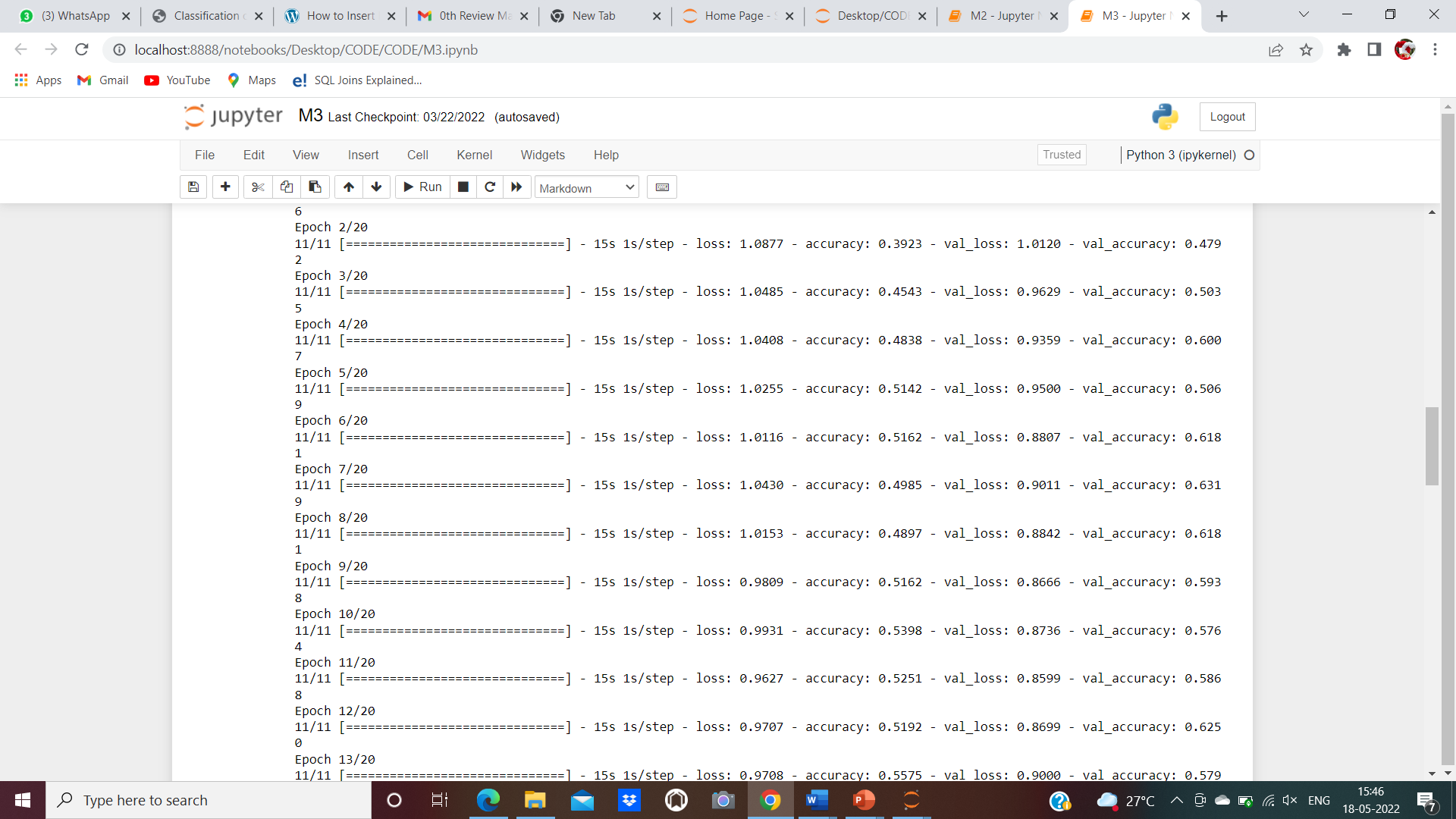
**PERFORMANCE ANALYSIS**

**CHAPTER-7**

**PERFORMANCE ANALYSIS**

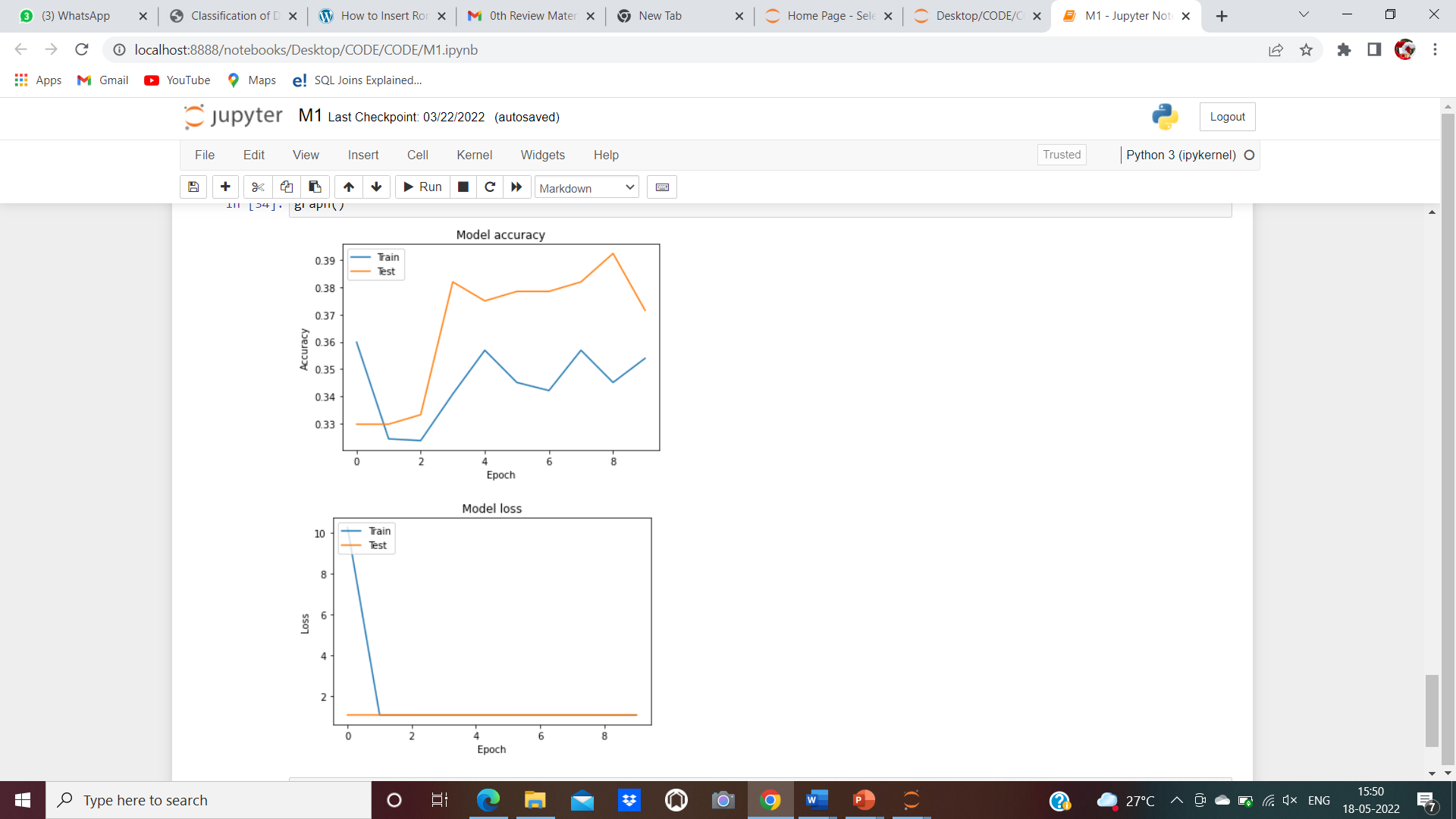
**7.1. PERFORMANCE ANALYSIS**

It finds Accuracy for all the modules and finally the third module LENET gave high accuracy so we have done project using LENET module**.**

****

**FIG 7.1. Accuracy Calculation for Dental Disease Classification**

This graph represents the training and testing we are giving for the machine.

****

**fig 7.2. Graph of Testing And Training for Dental Disease Classification**

**7.2. RESULTS AND DISCUSSION**

In this project, a research to classify dental Disease Classification over static facial images using deep learning ways was developed. This is a complex problem that has formerly been approached several times with different ways. While good results have been achieved using feature engineering, this project concentrated on feature learning, which is one of DL promises. Also feature engineering is not necessary, image pre-processing boosts classification accuracy. Then, it reduces noise on the input data. Currently, Agriculture grounded AI dental disease includes is heavily needed. The result completely based on feature learning does not seem close yet because of a major limitation. Therefore, Disease classification could be achieved by means of deep learning techniques.

**CHAPTER-8**

**CONCLUSION**

**CHAPTER-8**

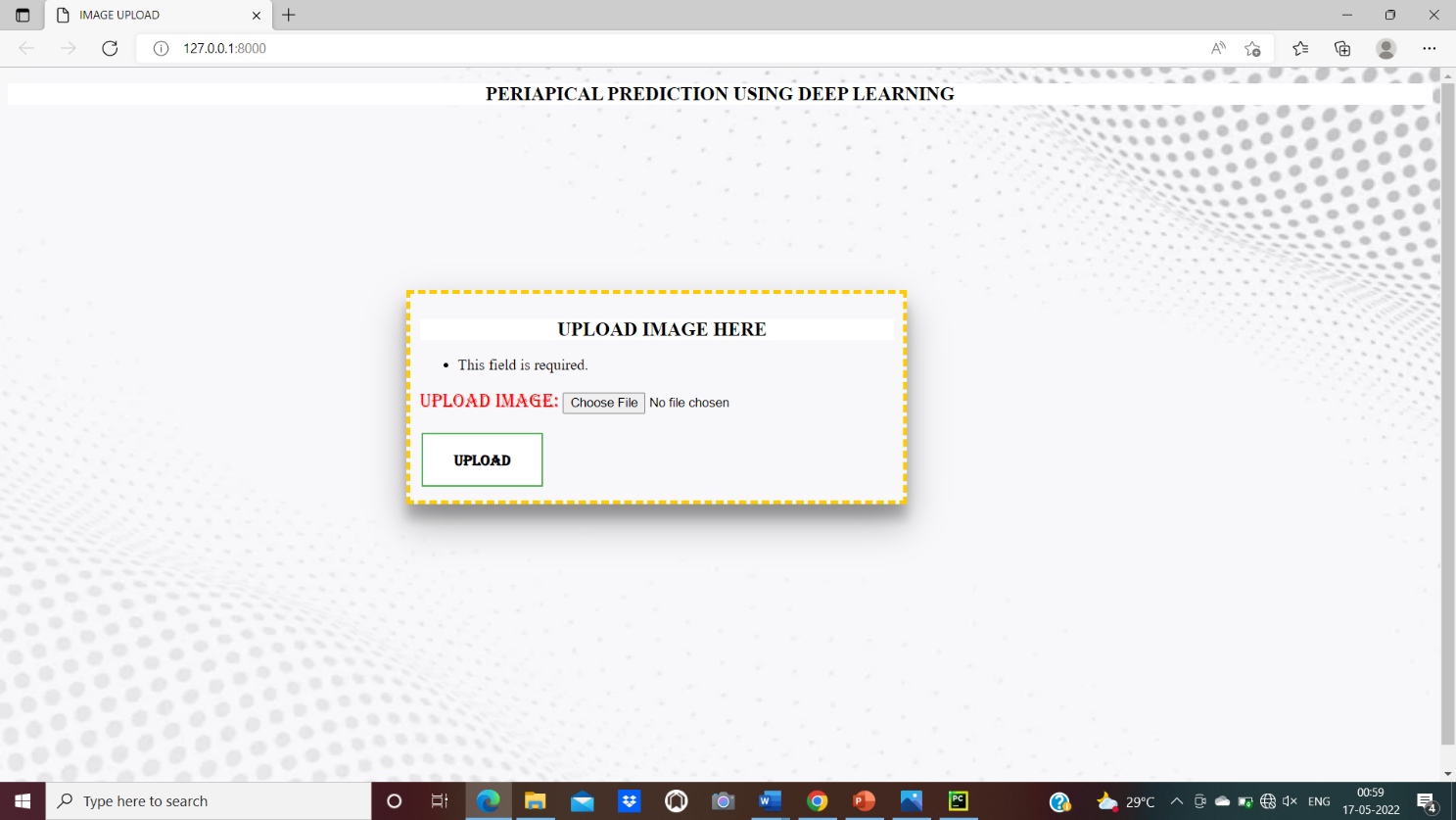
**CONCLUSION**

**8.1. Conclusion and Future Enhancements**

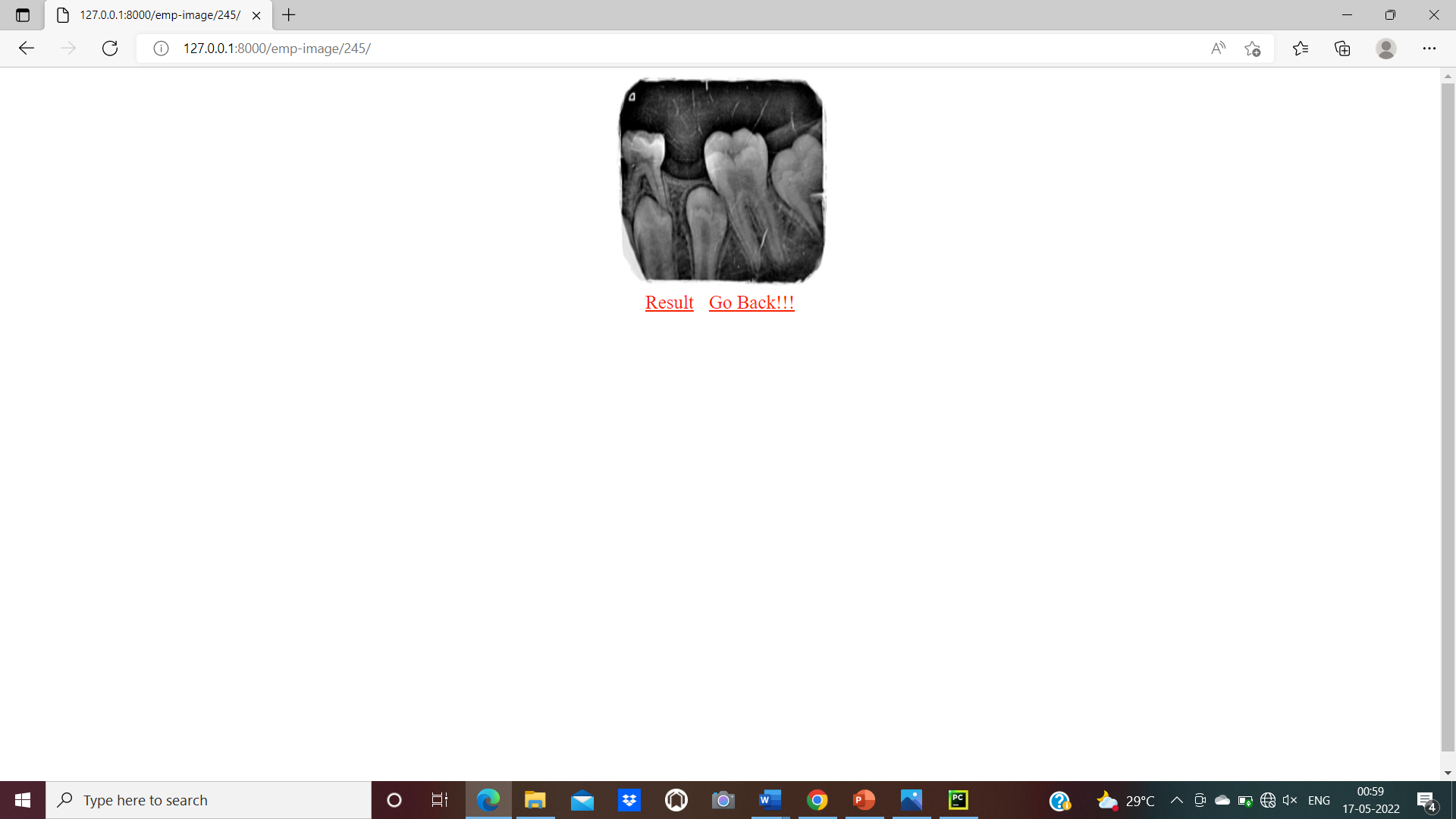
Further enhancement on the network’s accuracy and generalization can be achieved through the ensuing practices. The first one is to use the whole dataset during optimization. Using batch optimization is more suitable for longer datasets. Another technique is to estimate emotions one by one. This can lead to detect which emotions are more delicate to classify. Eventually, using a larger dataset for training seems beneficial. Still, such a dataset might not exist currently. Using several datasets might be a result, but a careful procedure to normalize them is needed. Eventually, using full dataset for training, pre-training on each dental, and using a larger dataset seem to have the possibility to improve the network’s performance. Therefore, they should be addressed in future research on this topic. Therefore, with further optimization of the dental dataset and improvements in the algorithm, a computer-aided detection system can be expected to become an effective and efficient method of diagnosing the dental diseases and by using the deep learning concept the third module lenet gave the more accuracy compared to other 2 modules. So by using the lenet we train the machine for more accuracy. Hence we demonstrated that the deep learning CNN algorithm was useful for assessing the diagnosis and predictability.

**APPENDICES**

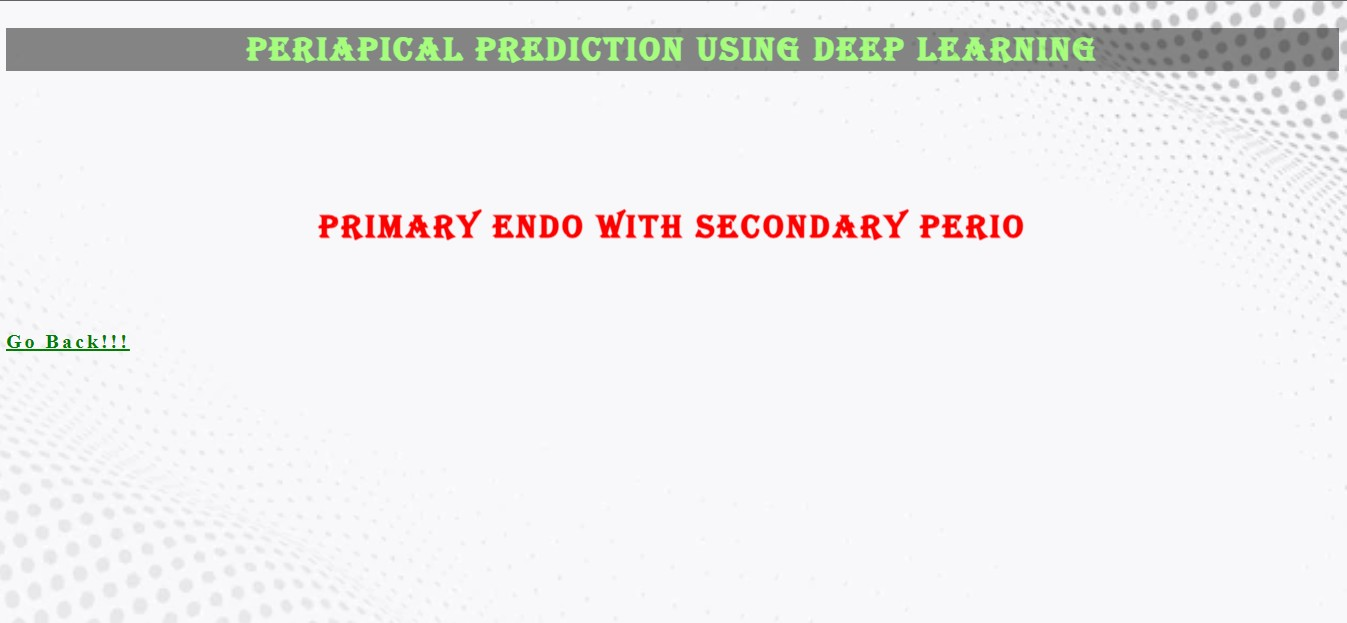
**A.1. Sample ScreenShots**



**A.1. Uploading Teeth Image**



**A.2. Click on Results**



**A.3. Specify the Type**

**REFERENCES**

[1]. Abdolvahab Ehsani Rad, Rosely KumoiMohd Shafry Mohd Rahim, Alireza Norouzi , “Dental X-Ray Image Segmentation and Multiple Feature Extraction”, 2012, AWERProcedia DOI:10.13140

[2]. Amaar Obaid Hassan, Gregory Y. H. Lip, “Acute Dental Periapical Abscess and New-Onset Atrial Fibrillation: A Nationwide, Population-Based Cohort Study” ,2021,MDPI, DOI:10.3390

[3]. Anuj Kumar, “Descriptive analysis of dental X-ray images using various practical methods” ,2021, PeerJ, DOI:10.7717

[4]. Anuj Kumar, H. S. Bhadauria, Nitin Kumar, “Fuzzy Clustering with Level Set Segmentation for Detection of Dental Restoration area”,2018,IEEE.

[5]. E. H. Said, D E M Nassar, G. Fahmy, H. Ammar , “Teeth segmentation In digitized dental X-ray films using mathematical morphology”,2006.

[6]. Jiafa Mao, Kaihui Wang, Yahong Hu, Weiguo Sheng, Qixin Feng, “GrabCut algorithm for dental X-ray images based on full threshold segmentation”,2018, IET,Volume 12.

[7]. Kasra KARAMIFAR, Afsoon TONDARI, Mohammad Ali SAGHIRI, “Endodontic Periapical Lesion: An Overview on the Etiology, Diagnosis and Current Treatment Modalities”,2020, EUROPEAN ENDODONTIC DOI:10.14744

[8]. María Prados-Privado, Javier García Villalon, Carlos Hugo Martínez-Martínez and Carlos Ivorra, “Dental Images Recognition Technology and Applications”,2020, DOI:10.2856.

[9]. Parkavi. B, Roshini. T.J, Maheswari. M, “Segmentation of Periapical Dental X-Ray Images by Applying Morphological Operation” 2019, IFERP,Volume 5.

[10]. Park WJ and Jun-Beompark, “History and application of artificial neural network in dentistry” ,2018 dec: 12(4):594-601.

[11]. P.N.R. Nair, “Pathogenesis of apical periodontities and the causes of Endodontic Failures” ,2004.

[12]. Rad A.E, Rahim .M, Shafry .M, Rehman.A, Altameem. A, and Saba. T , “Evaluation of Current Dental Radiographs Segmentation Approaches in Computer aided Applications” ,2013,Volume 30.

[13]. Santhosh B, “Review on emerging techniques on detects oral cancer”, 2015,1(1); 41-6.

[14]. Sibel Koçak, Mustafa Murat, “Periapical Health Related to the Quality of Coronal Restorations and Root Fillings in a Turkish Population”,2013, CumhuriyetDent J DOI:10.7126.

[15]. Sindu Divakaran, Suja D, “Classification of Digital Dental X-ray Images Using Machine Learning” ,2021,IEEE, DOI:10.1109.

[16]. Sudhakar. T, Bethanney Janney.J, Haritha. D, Juliet Sahaya. M and Parvathy. V, “Automatic Detection and Classification of Brain Tumor using Image Processing Techniques” ,2017,10(11), 1994-1998.

[17]. Vicente Rueda, Ulises Velazquez, “Using the Periapical Index to evaluate the healing of periapical lesions after root canal treatment”,2020, Giornale Italiano di Endodnzia, DOI:10.32067.