CHAPTER-1

INTRODUCTION

In this world of simulation, the identity of any individual is always questionable in situations of mass massacres and disasters. A lot of literature is available on forensic odontology tools, but still this branch of odontology is in its infancy stage in India.

Establishing a person's identity can be a very difficult process in forensic identification. Dental, fingerprints, and DNA comparisons are the most common techniques used in this context allowing fast and secure identification. However, since they cannot be always used, sometimes simple techniques can be used successfully in human identification, such as 'Palatal rugoscopy, ' which is the study of palatal rugae. Palatal rugae have been equated with fingerprints and are unique to an individual.

It can be of special interest in edentulous cases and also in certain conditions where there are no fingers to be studied, such as burned bodies or bodies that underwent severe decomposition.

Palatal rugae are irregular, asymmetric ridges of the mucous membrane extending laterally from the incisive papilla and the anterior part of the palatal raphe. The location of palatal rugae inside the oral cavity confers them with stability even when exposed to high temperatures or trauma. Their resistance to trauma and their apparent unique appearance has suggested their use as a tool for forensic identification.

1.1 Problem Statement

The uniqueness of palatal rugae pattern can be utilized similar to fingerprints and when combined with other methods, it can help in the identification of a person. Thus, these unique characteristics of the palatal rugae can be studied and can be enhanced a using Image processing and can be used as identity for the each individual. By using the data samples of palates of different individual, uniqueness of the identity can be recognized. The image processing techniques can be applied to these data samples to get the desired result.

1.2 Objectives

The objectives of project are:

- To learn different characteristics of palatal rugae.
- To implement various techniques of image processing to enhance the pattern of palatal rugae.
- To understand the features of palatal rugae (such as area), to differentiate between the various data samples of different individuals.

1.3 Advantages

The advantages of project are:

- Palatal rugae pattern is considered to be unique to an individual and hence hold the potential for identification.
- The form, layout, and characteristics of rugae are not affected by either eruption or loss of teeth. They are stable and resist decomposition for up to 7 days after death.
- Low utilization cost makes palatal rugae an ideal forensic identification parameter.

CHAPTER-2

LITERATURE SURVEY

[Harjeet Kaur Sekhon, Keya Sircar1, Sanjeet Singh, Deepti Jawa2, Priyanka Sharma] (2014), this study was undertaken to study and record the rugae pattern with respect to biometric characteristics of shape, size, direction, number and position in a cross section of the population in Uttar Pradesh. The individuality of the rugae pattern and their correlation with sex of the individual was also evaluated. Identification is the establishment of identity of an individual. The basis of dental identification is based on the observation that no two individuals can have same dentition. Palatal rugae are irregular, asymmetric ridges of the mucous membrane extending laterally from the incisive papilla and the anterior part of the palatal raphe. The location of palatal rugae inside the oral cavity confers them with stability even when exposed to high temperatures or trauma. Their resistance to trauma and their apparent unique appearance has suggested their use as a tool for forensic identification.

To record the biometric characteristics of shape, size, direction, number and position of palatal rugae and analyze whether palatal rugoscopy can be used as a tool for personal identification and for sex determination. This study concluded that rugae pattern are highly individualistic and can be used as a supplementary method for personal identification and sex determination.

Further inter-observer and intra-observer variability were not found to be significant, which further validates the use of rugoscopy as a forensic tool. The total number of rugae was determined on right and left side and the total number of rugae in males and females were compared. The largest number of rugae was 16 and the least was 6. The average number of rugae was slightly more in females than in males, but it was statistically insignificant.

[Namrata Harchandani, Swati Marathe, Rahul Rochani, Shams Ul Nisa1](2015), this paper consists of observation and compare the distribution of various palatal rugae patterns in western and northern Indian populations and to study the variations in male and female subjects respectively .The study consisted of 100 subjects, 50 each from the two groups of geographically different regions of western and northern India. After obtaining informed consent, an alginate impression of maxillary arch was made for interpretation.

The number, type, and unification were followed according to Thomas and Kotze's classification and the shape was recorded according to Kapali et al.'s classification.The shape of rugae was compared between the two study groups and was found to be highly significant between western Indian and northern Indian subjects. The number and shape of rugae differed significantly between the genders, with males having a highly significant difference as compared to the females. The western Indian group showed wavy shape predominantly in males and females had straight rugae. Similarly, the northern Indian male participants also had wavy shape; however, females in this group had more curved shaped rugae.

The uniqueness of palatal rugae pattern can be utilized similar to fingerprints and when combined with other methods, it can help in the identification of a person. Literature review shows that the two different populations of the geographical regions of western and northern India which were selected for the present study by the researchers were never studied earlier. The low-cost utilization, simplicity, and reliability have added strength to the study.

A large amount of data can be stored and quick retrieval of information will be possible which may assist in immediate and effective identification of an individual. It would be beneficial to conduct further studies in large samples and taking more parameters for palatal rugae analysis in all races of the world, so that a national data can be prepared. The palatal rugae pattern can act as a fingerprint in identification of a person. The analysis of palatal rugae combined with other methods is an important alternative and complementary technique for human identification.

[Aparna Paliwal, Sangeeta Wanjari, and Rajkumar Parwani](2010),The purpose of this study was to compare the palatal rugae patterns in 2 different populations in India (Madhya Pradesh and Kerala), and furthermore, to assess the predominant pattern if any in the selected groups.

Palatal rugae are irregular, asymmetric ridges of the mucous membrane extending laterally from the incisive papilla and the anterior part of the palatal raphe. The uniqueness and the overall stability of palatal rugae suggest their use for forensic identification.

After analyzing the rugae patterns in both the groups and between the 2 sides of the palate, the wavy pattern was found to be predominant followed by curved, straight, unification, circular, and nonspecific in decreasing order in the overall population. Straight rugae pattern on the right side of the palate in the male subjects was found to be significantly predominant in the MP population, whereas wavy shape was predominant in Keralites; however, rugae patterns on the right side of the palate in female subjects exhibited no significant difference.

In the literature, the consensus of opinion is that the rugae remain fairly stable in number and morphology except when there is trauma, such as loss of tooth, persistent pressure, extreme finger sucking, orthodontic tooth movement, which may modify the alignment. Thomas and Kotze (1983) studied the rugae patterns of 6 South African populations to analyze the interracial difference. They found that rugae were unique to each ethnic group and that it can be used successfully as a medium for genetic research.

CHAPTER-3

REQUIREMENTS SPECIFICATION

In this chapter, we have mentioned software and hardware requirement specification of our project. Then we have briefly described about some of the most important components.

3.1 Software Requirements

- Programming Language : Matlab, Python 3
- Libraries : Simulink, Block, Python Standard libraries
- Dataset : Palatal cast
- Platform : Mathworks, Pycharm 2019.2.1
- IDE : Matlab R2018a, Pycharm 2019.2.1
- Operating System : Windows 10

3.2 Hardware Requirements

- Processor : 2 GHz dual core processor
- RAM : 2 GB RAM (system memory)
- Hard disk : 25 GB of hard-drive space
- Display : Standard Output Display

MATLAB

MATLAB is a programming language developed by MathWorks. It started out as a matrix programming language where linear algebra programming was simple. It can be run both under interactive sessions and as a batch job. This tutorial gives you aggressively a gentle introduction of MATLAB programming language. It is designed to give students fluency in MATLAB programming language. Problem-based MATLAB examples have been given in simple and easy way to make your learning fast and effective.

MATLAB is developed by MathWorks. MATLAB (matrix laboratory) is a fourth-generation high-level programming language and interactive environment for numerical computation, visualization and programming. It has numerous built-in commands and math functions that help you in mathematical calculations, generating plots, and performing numerical methods.

Features of MATLAB

Following are the basic features of MATLAB −

- It is a high-level language for numerical computation, visualization and application development.
- It also provides an interactive environment for iterative exploration, design and problem solving.
- It provides vast library of mathematical functions for linear algebra, statistics, Fourier analysis, filtering, optimization, numerical integration and solving ordinary differential equations.
- It provides built-in graphics for visualizing data and tools for creating custom plots.
- MATLAB's programming interface gives development tools for improving code quality maintainability and maximizing performance.
- It provides tools for building applications with custom graphical interfaces.
- It provides functions for integrating MATLAB based algorithms with external applications and languages such as C, Java, .NET and Microsoft Excel.

Uses of MATLAB

MATLAB is widely used as a computational tool in science and engineering encompassing the fields of physics, chemistry, math and all engineering streams. It is used in a range of applications including −

- Signal Processing and Communications
- Image and Video Processing
- Control Systems
- Test and Measurement
- Computational Finance
- Computational Biology

CHAPTER-4

METHODOLOGY

4.1 Project Architecture

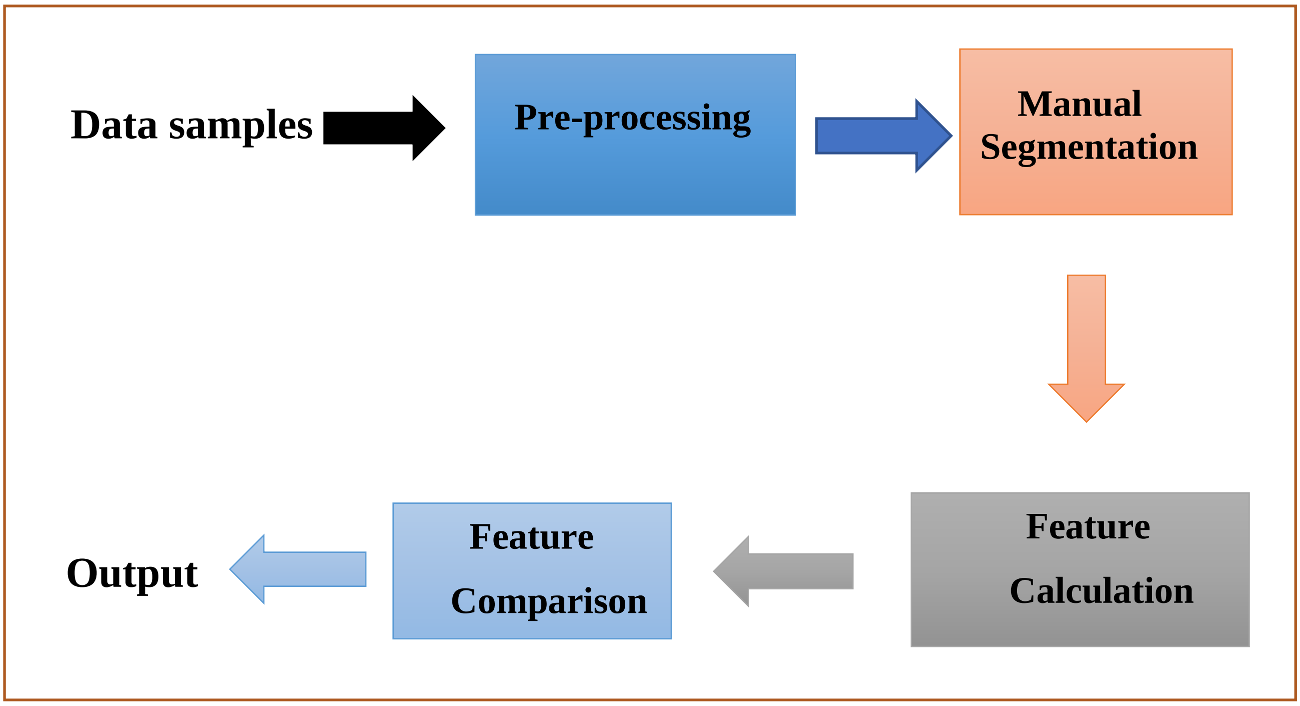

Figure 4.1 architecture of identification of image patterns

4.1.1. DATA SAMPLES

The sample consist of 10 subjects between 18 and 25 years. Maxillary impressions were made with elastomeric impression material and dental stone was used to make models. The palatal rugae patterns were traced and analyzed with a Digital Camera.The samples were collected from the various Dental clinics of davanagere. These samples were coded with numbers for easy understanding of the different individual samples.

Figure 4.2 Data sample 7 Palatal rugae

4.1.2. PRE-PROCESSING

Image processing:

Image Processing is any form of signal processing for which our input is an image, such as photographs or frames of video and output can be either an image or a set of characteristics or parameters related to the image a method to convert an image into digital form and perform some operations on it, in order to get an enhanced image or to extract some useful information from it. It is a type of signal dispensation in which input is image, like video frame or photograph and output may be image or characteristics associated with that image. Usually Image Processing system includes treating images as two-dimensional signals while applying already set signal processing methods to them. It is among rapidly growing technologies today, with its applications in various aspects of a business. Image Processing forms core research area within engineering and computer science disciplines too.

Image processing basically includes the following three steps:

- Importing the image with optical scanner or by digital photography.
- Analyzing and manipulating the image which includes data compression and image enhancement and spotting patterns that are not to human eyes like satellite photographs.
- Output is the last stage in which result can be Processed/ altered image or report that is based on image analysis.

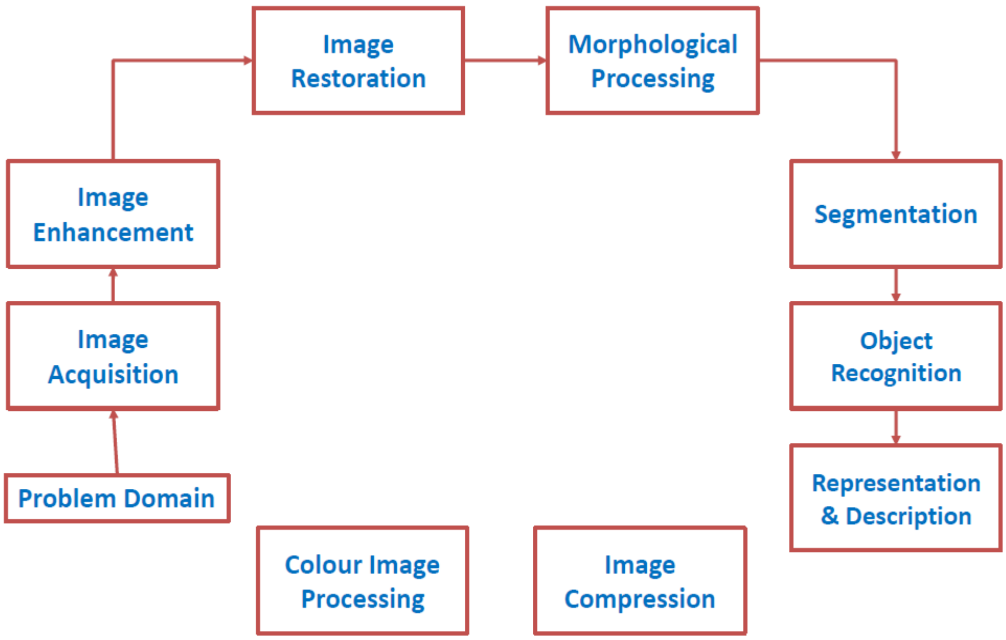

Figure 4.3 Steps involved in image processing

Purpose of Image processing:

The purpose of image processing is divided into groups.

They are:

- Visualization - Observe the objects that are not visible.
- Image sharpening and restoration - To create a better image.

- Image retrieval - Seek for the image of interest.
- Measurement of pattern – Measures various objects/ Parameters/ Feature in an image.
- Image Recognition – Distinguish the objects in an image.

Types:

- **Analog Image Processing:**

Image processing task conducted on two-dimensional analog signals by analog means.

- **Digital Image Processing**:

Processing digital image, we can also say that it is a use of computer algorithms, in order to get enhanced image either to extract some useful information.

Image Pre-processing:

Steps to be taken for the image pre-processing are-

- Read Image
- Resize Image
- Remove Noise (Image Filtering and Enhancement)

4.1.2.1 Reading image in the workspace of MATLAB

In this step, we store the path to our image dataset into a variable then we created a function to load folders containing image into arrays.

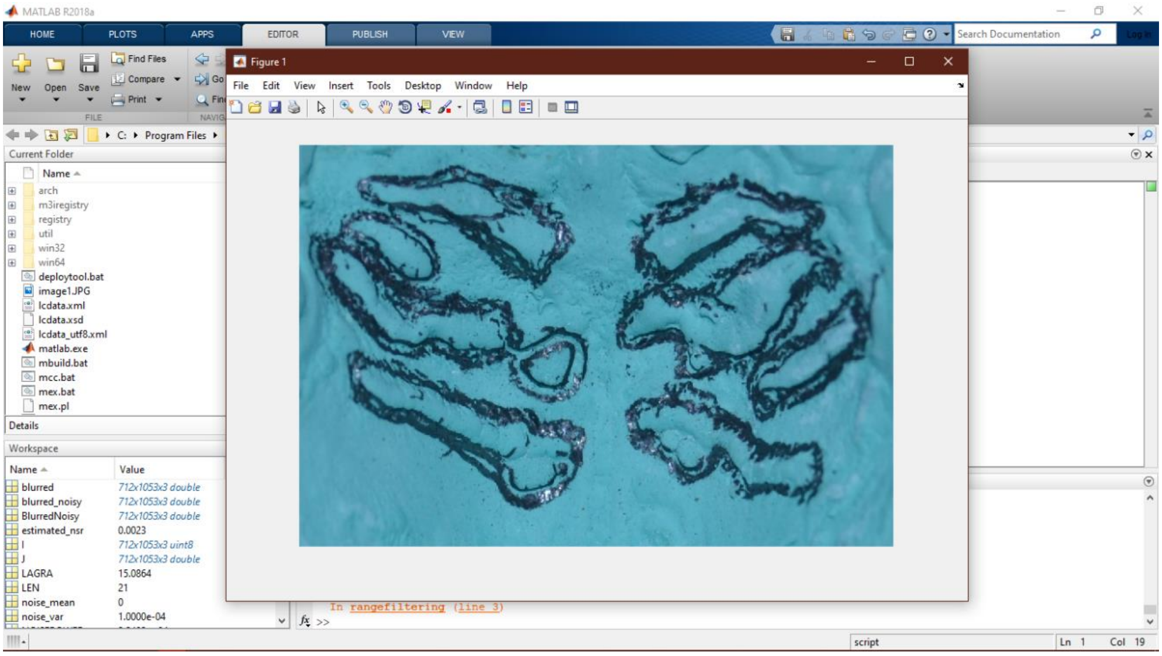

Figure 4.4 Reading Image into the Workspace of MATLAB

4.1.2.2 Resize Image

Some images captured by a camera and fed vary in size, therefore, we should establish a base size for all images.

4.1.2.3 Image Filtering and Enhancement

Image enhancement is the process of adjusting digital images so that the results are more suitable for display or further image analysis. For example, you can remove noise, sharpen, or brighten an image, making it easier to identify key features.

The steps we have taken to achieve Enhanced and Filtered image are:

1. **Histogram Equalization:**

Histogram of an image, like other histograms also shows frequency. But an image histogram, shows frequency of pixels intensity values. In an image histogram, the x axis shows the gray level intensities and the y axis shows the frequency of these intensities.

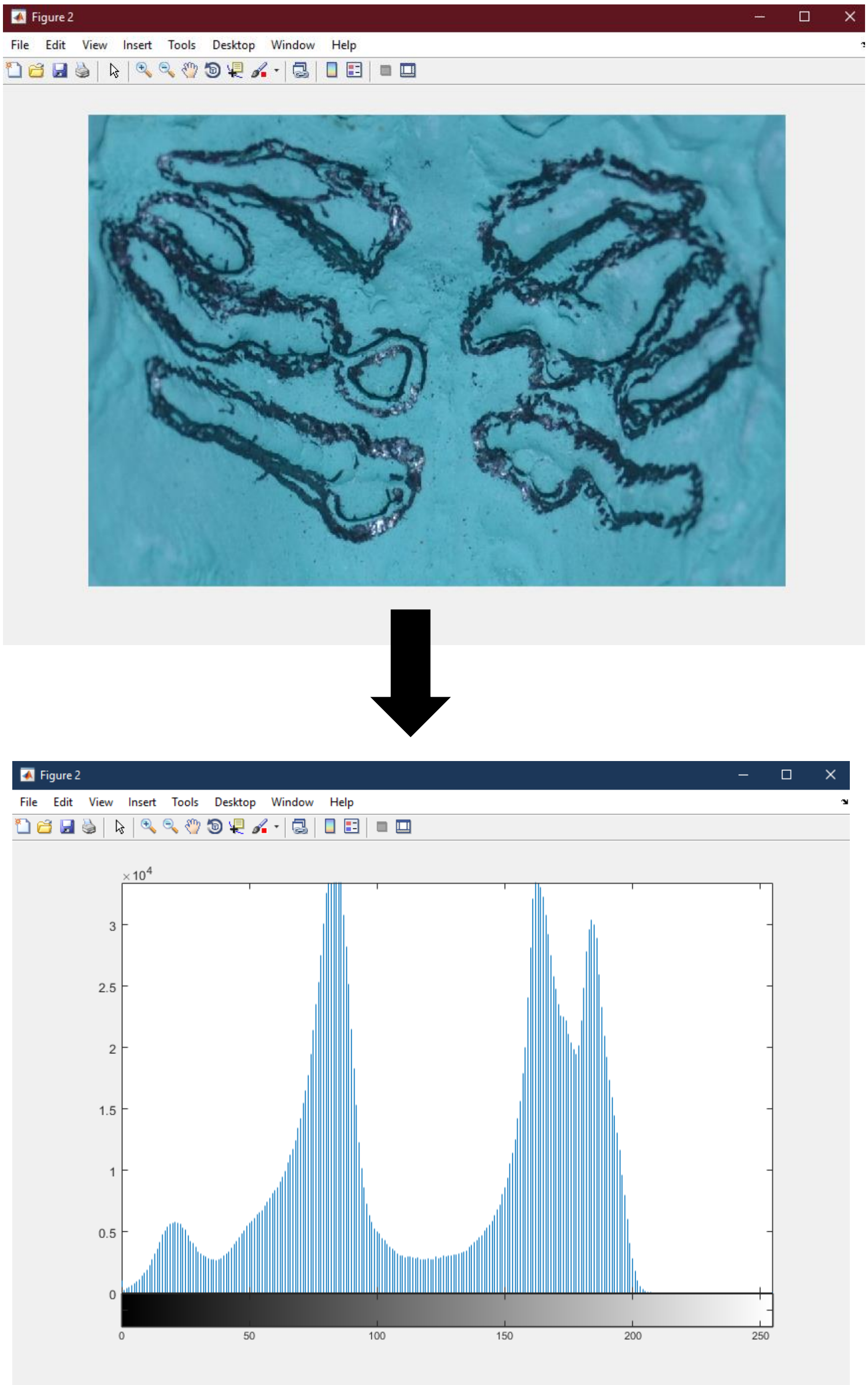

Figure 4.5 Histogram of sample image

After applying **histeq** function for the fig 4.5 we get the enhanced image which all the intensities are equalized this is called Histogram Equalization which can be seen in fig 4.6.

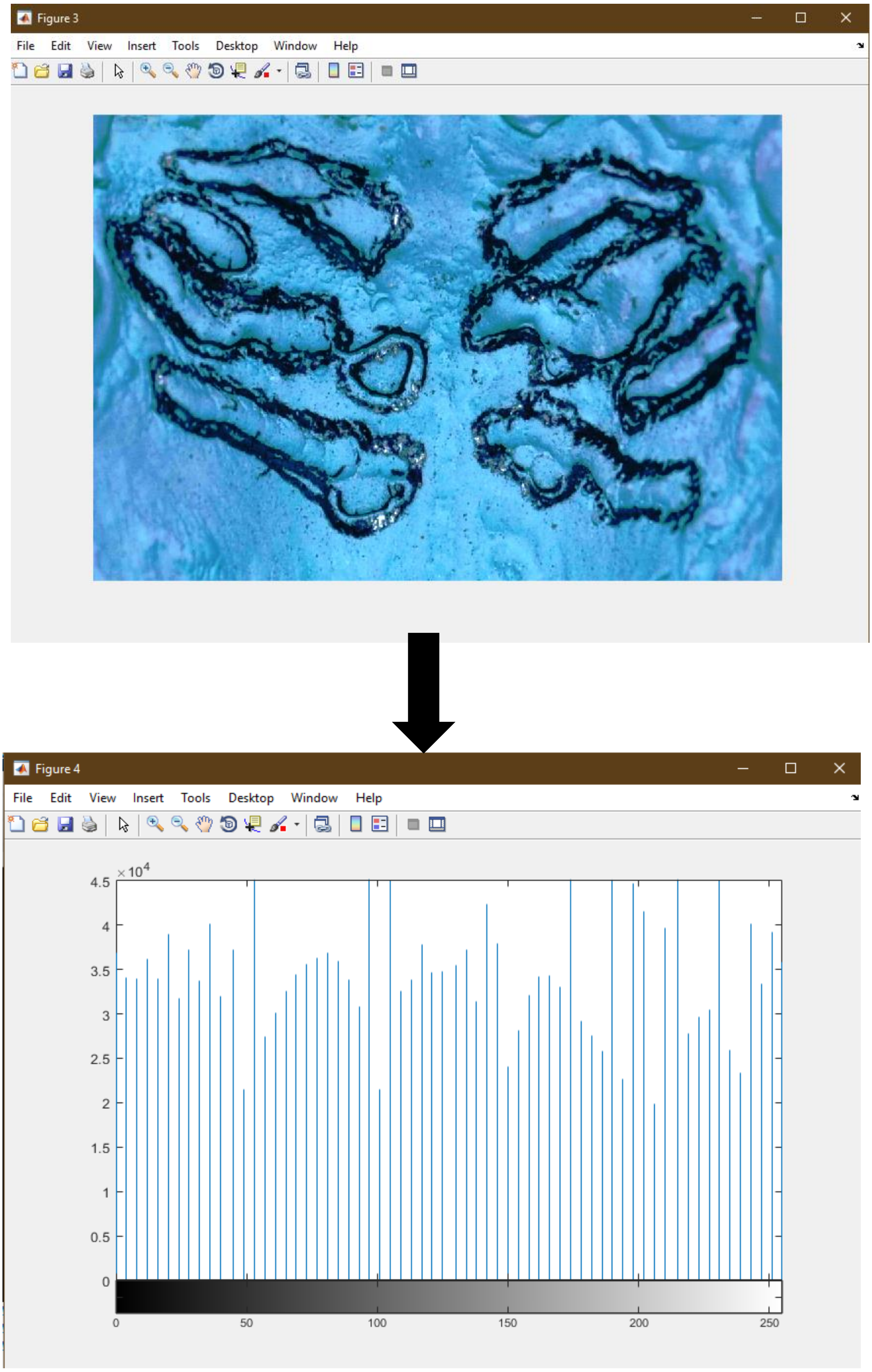

Figure 4.6 Histogram Equalization Process

The process of adjusting intensity values can be done automatically using histogram equalization. Histogram equalization involves transforming the intensity values so that the histogram of the output image approximately matches a specified histogram. By default, the histogram equalization function, histeq, tries to match a flat histogram with 64 bins, but you can specify a different histogram instead.

As you can see from the graph, that most of the bars that have high frequency lies in the first half portion which is the darker portion. That means that the image we have got is darker. And this can be proved from the fig 4.6.

Applications of Histograms

Histograms has many uses in image processing. The first use as it has also been discussed above is the analysis of the image. The second use of histogram is for brightness purposes. The histograms have wide application in image brightness. Not only in brightness, but histograms are also used in adjusting contrast of an image.

2. RGB To Gray Conversion:

The rgb2gray function converts RGB images to grayscale by eliminating the hue and saturation information while retaining the luminance. "rgb2gray" converts RGB values to grayscale values by forming a weighted sum of the R, G, and B components:

0.2989 * R + 0.5870 * G + 0.1140 * B

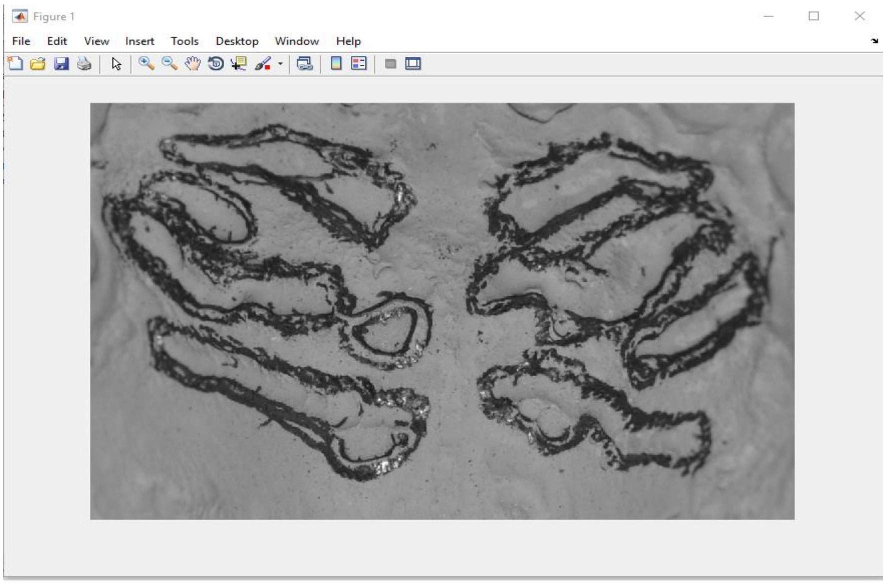

Figure 4.7 Rgb2Gray Conversion

3. Median Filter:

Median filtering is a nonlinear operation often used in image processing to reduce "salt and pepper" noise. A median filter is more effective than convolution when the goal is to simultaneously reduce noise and preserve edges.

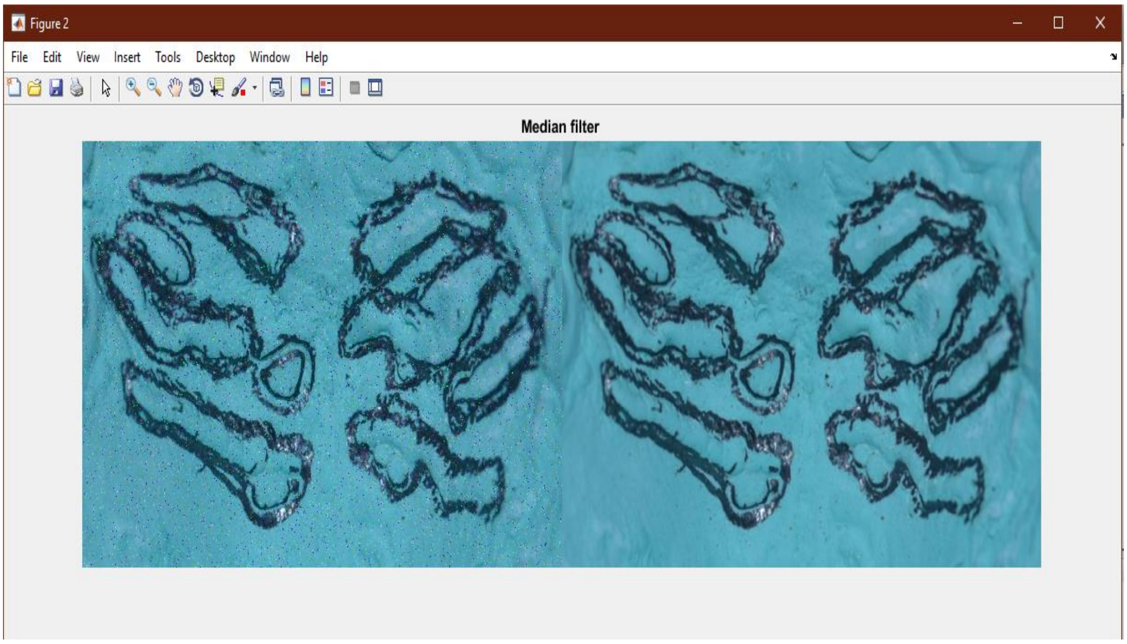

Figure 4.8 Median filtering Process

4. Regularized Filter:

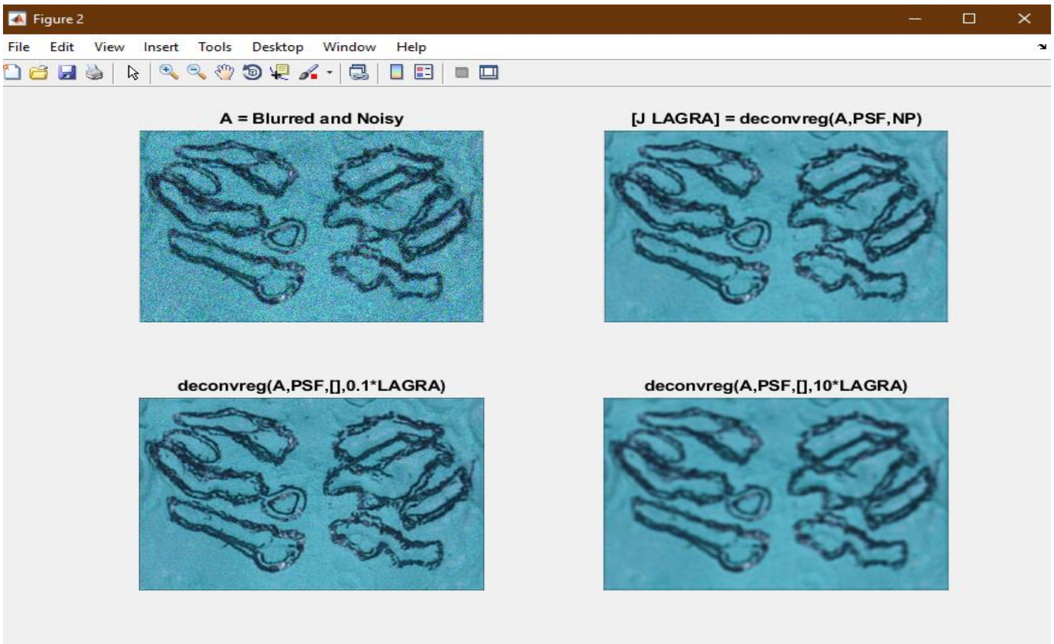

Figure 4.9 Regularized Filter Processing

Regularized deconvolution can be used effectively when limited information is known about the additive noise and constraints (such as smoothness) are applied on the recovered image. The blurred and noisy image is restored by a constrained least square restoration algorithm that uses a regularized filter.

4.1.3 Manual Segmentation and Cropping

Image segmentation is a commonly used technique in digital image processing and analysis to partition an image into multiple parts or regions, often based on the characteristics of the pixels in the image. Image segmentation could involve separating foreground from background, or clustering regions of pixels based on similarities in color or shape.

In this project we have used segmentation technique based on the pattern that have been classified using classification of Thomas and Kotze and the shape was recorded based on Kapali *et al*.'s classification.

Pattern is the subset of similar objects in a larger set (a class or a cluster).Also used for the entire similarity structure in a collection of objects as well as for a single object which is typical for a set of similar objects.

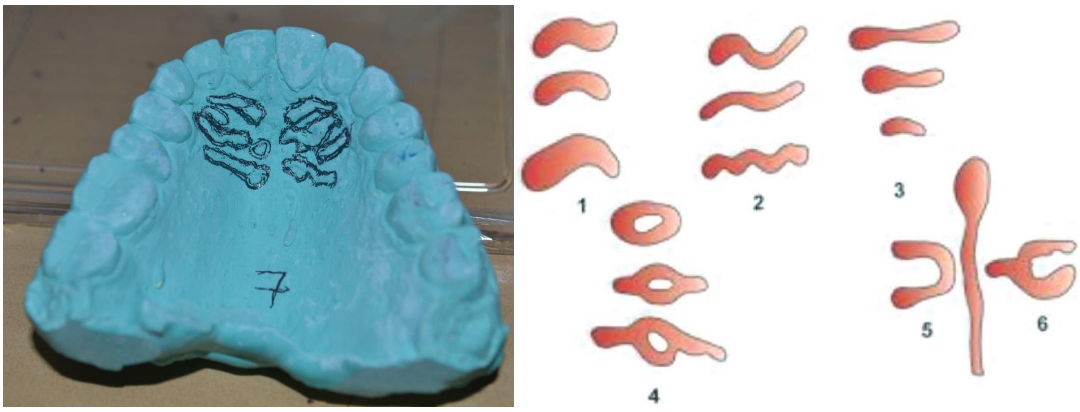

Figure 4.10 Types and unification of palatal rugae patterns: 1) Curve; 2) Wavy; 3) Straight; 4) Circular; 5) Convergent; and 6) Divergent

The number, type, and unification pattern were recorded in accordance with the classification of Thomas and Kotze and the shape was recorded based on Kapali *et al*.'s classification, as shown in Figure4.10.

1. Cropping of The Image (imcrop):

The “imcrop” function is used to crop the enhanced image based on the region of interest identified using the segmentation. imcrop creates an interactive Crop Image tool associated with the image displayed in the current figure. The Crop Image tool blocks the MATLAB command line until you complete the operation. The Crop Image tool is a moveable, resizable rectangle that you can position over the image and perform the crop operation interactively using the mouse.

When the Crop Image tool is active in a figure, the pointer changes to cross hairs + when you move it over the target image. Using the mouse, you specify the crop rectangle by clicking and dragging the mouse. You can move or resize the crop rectangle using the mouse. When you are finished sizing and positioning the crop rectangle, create the cropped image by double-clicking the left mouse button. You can also choose **Crop Image** from the context menu.

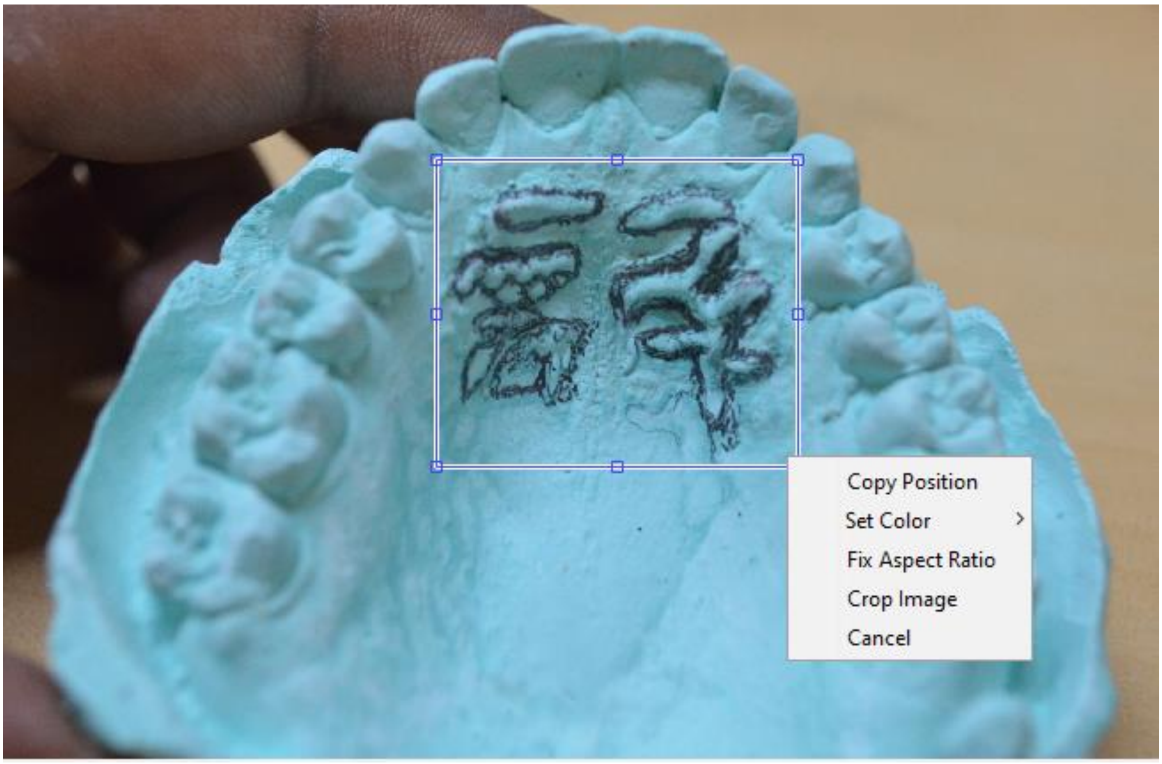

Figure 4.11. Cropping of Image using Crop Interactive Tool (imcrop function)

2. CONVERSION OF GRAY TO BLACK AND WHITE (BINARY) IMAGE:

We use **“imbinarize”** function to convert grayscale image to binary image or black and white image. imbinarize(I) creates a binary image from 2-D or 3-D grayscale image I by replacing all values above a globally determined threshold with 1s and setting all other values to 0s. By default, imbinarize uses Otsu's method, which chooses the threshold value to minimize the intraclass variance of the thresholded black and white pixels. imbinarize uses a 256-bin image histogram to compute Otsu's threshold.

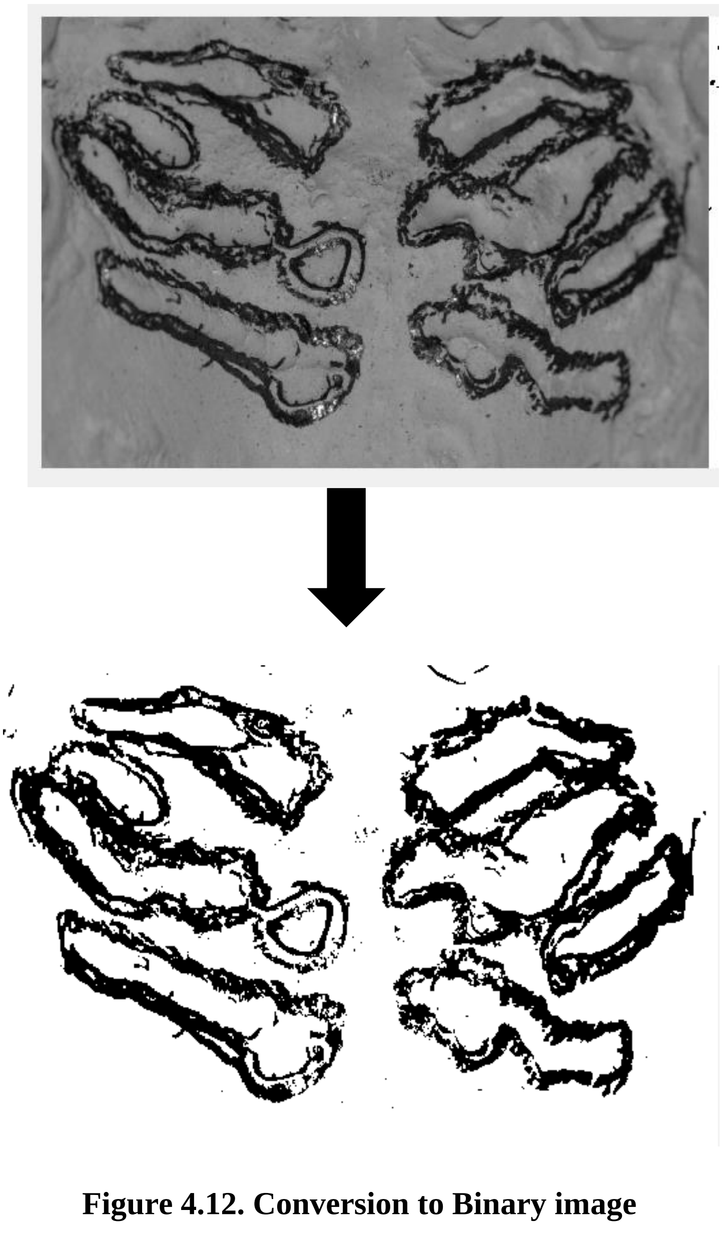

Figure 4.12. Conversion to Binary image

4.1.4 FEATURE CALCULATION

Calculation of Area:

In this project, We have considered the Area of Region of Interest (i.e palatal rugae pattern) as the feature. After the segmentation technique and conversion of grayscale image into the binary image the next important step is calculation of the area.

For the area calculation we have developed the algorithm as follows:

Algorithm to Calculate Area of palatal rugae pattern:

Input: Binary(black and white) image of the Region of Interest i.e Palatal rugae pattern.

Output: Area of the ROI i.e palatal rugae pattern pixels.

Step 1 : Invert the Black and white image i.e Binary image. So that Black pixels of the image are converted to White pixels and White pixels to Black.

BW= ~BW where BW is Binary image.

Step 2 : Rows and Columns are calculated using size function.

[r , c] = size(BW)

Step 3 : Black and White pixels count is initialized to 1.

Step 4 : Use the for loop to iterate to the last row and use another for loop inside the first for loop to iterate till the last column

```
for i=1: last row ( r )
    for j=1: last column ( c )
        if BW(i,j)==0
            Increment Black pixel count  by 1;
        else
            Increment White pixel count by 1;
```

Step 5 : Print the white pixel count i.e Area of the palatal rugae pattern

4.1.5. FEATURE COMPARISON

After the calculation of area for each sample, the next step is to compare each calculated area with the other obtained area values that is areas of the other samples.

This method works on the example of handshaking property where in 20 people are standing in a circle and each one of the individual must handshakes with 19 remaining different individuals. Thus, to get this done we have developed a python code wherein we have used the list to get all the 20 values as input and the output will be produced if any values of areas match or no areas match with any of the values.

CHAPTER-5

IMPLEMENTATION

This chapter deals with the implementation details of Image processing of Palatal rugae.

5.1 Reading the image in the Workspace of MATLAB

Read an image into the workspace, using the imread command. This code reads one of the sample images included with the toolbox, an image of palate sample 7 named image7 , and stores it in an array named I . imread infers from the file that the graphics file format is jpg.

The code to read the image into the workspace is as follows:

I = imread('image7.jpg');

figure,

imshow(I)

Display the image, using the imshow function. You can also view an image in the Image Viewer app. The imtool function opens the Image Viewer app which presents an integrated environment for displaying images and performing some common image processing tasks.

5.2 Image Filtering and Enhancement

Filtering is a technique for modifying or enhancing an image. For example, you can filter an image to emphasize certain features or remove other features. Image processing operations implemented with filtering include smoothing, sharpening, and edge enhancement.

The steps we have taken to achieve Enhanced and Filtered image are:

1. Histogram Equalization:

This code shows how to use histogram equalization to adjust the contrast of a grayscale image. The original image has low contrast, with most pixel values in the middle of the intensity range. histeq produces an output image with pixel values evenly distributed throughout the range.

The code for histogram equalization is as follows:

```
I = imread('image7.jpg');
figure, imshow(I)
figure
imhist(I)
I2 = histeq(I);
figure
imshow(I2)
figure
imhist(I2)
```

This code adjusts the intensity levels of the sample image 7. By this way Histogram Equalization is done to enhance the sample images of palate.

2. Applying the Median filter for the enhanced image samples:

Median filtering is a nonlinear operation often used in image processing to reduce "salt and pepper" noise.If the input image I is of an integer class, then all the output values are returned as integers. If the number of pixels in the neighborhood (i.e., m*n) is even, then some of the median values might not be integers. In these cases, the fractional parts are discarded.

J = medfilt2(I) performs median filtering of the image I in two dimensions. Each output pixel contains the median value in a 3-by-3 neighbourhood around the corresponding pixel in the input image.

J = medfilt2(I,[m n]) performs median filtering, where each output pixel contains the median value in the m-by-n neighbourhood around the corresponding pixel in the input image.

The code used to enhance the image using Median Filtering method is as follows:

```
#Read image into workspace and display it
I = imread('image7.jpg');
figure, imshow(I)
# Add salt and pepper noise
J = imnoise(I,'salt & pepper',0.02)
# Use a median filter to filter out the noise.
K = medfilt3(J)
imshow(K)
# Display results, side-by-side.
imshowpair(J,K,'montage')
title('Median filter ')
```

This is how the median filter is applied.

3. Applying the Regularized filter for the enhanced image samples:

This code shows how to use regularized deconvolution to deblur images. Regularized deconvolution can be used effectively when limited information is known about the additive noise and constraints (such as smoothness) are applied on the recovered image. The blurred and noisy image is restored by a constrained least square restoration algorithm that uses a regularized filter.

The code used to enhance the sample image7 using Regularized filter is as follows:

```
I = im2double(imread('image7.jpg'));
imshow(I);
PSF = fspecial('gaussian',7,10);
V = .01;
```

```
BlurredNoisy = imnoise(imfilter(I,PSF),'gaussian',0,V);
NOISEPOWER = V*prod(size(I));
[J LAGRA] = deconvreg(BlurredNoisy,PSF,NOISEPOWER);
subplot(221); imshow(BlurredNoisy);
title('A = Blurred and Noisy');
subplot(222); imshow(J);
title('[J LAGRA] = deconvreg(A,PSF,NP)');
subplot(223); imshow(deconvreg(BlurredNoisy,PSF,[],LAGRA/10));
title('deconvreg(A,PSF,[],0.1*LAGRA)');
subplot(224); imshow(deconvreg(BlurredNoisy,PSF,[],LAGRA*10));
title('deconvreg(A,PSF,[],10*LAGRA)')
```

Note: This method of enhancement is a Substitute step that can be used in the place of Median filter if the previous methods or filters doesn't provide the good results. This method has been used for the testing and has not been included in the main process.

5.3 Manual Segmentation and Feature Calculation

As we have seen the above processes or methods which have been described as separate methods, the non-programmer might find it difficult to carry out each process separately to get the results. The problem they might face is to take input image at each step, apply a method and obtain the output image and take that output image as input for the next step and so on. So, to tackle this complexity, we have developed the following code which is combination of each step into the one code. This code reduces the complexity of the process and any non-programmer can find it user friendly solution to calculate the area of palatal rugae.

```
%% Take image
[file, path] = uigetfile('*.jpg','select the color palate image');%user selects the image.
I = imread([path file]);%read the image
subplot(3,2,1),imshow(I),title('Original Image')

%% Gray conversion
[r,c,d] = size(I);%calculating the size of the image
if (d==3)
    gray = rgb2gray(I);%
    subplot(3,2,2),imshow(gray),title('Grayscaled Image')
    %gray=imresize(gray,[28,28]);%using interpolation
else
    gray = I ;
    subplot(3,2,2),imshow(gray),title('Image is already grayscale')
    %gray=imresize(gray,[28,28]);
end

%% Pre-processing using median filter
fil = medfilt2(gray);%filtering the image
subplot(3,2,3),imshow(fil),title('Median filtered Image')

%% cropping the image
[Crop, rect] = imcrop(fil);%crop the image manually and store in crop
subplot(3,2,4),imshow(Crop),title('Manually cropped Image')
```

```
%% convert the image into black and white
%BW = Crop > thresh;
BW = imbinarize(Crop)
BW = ~BW
subplot(3,2,5),imshow(BW),title('white and black Cropped Image')

%% area calculation using pixels
[r,c]=size(BW);
bc=1;
wc=1;
for i=1:r
    for j=1:c
        if BW(i,j)==0
            bc=bc+1;
        else
            wc=wc+1;
        end
    end
end
msgbox(string(wc),'Area')
```

In this project, the following code is used to calculate the Feature i.e area of the Palatal Rugae.

5.4 FEATURE COMPARISON

The python code that is being developed for the comparison of the derived results from the various sample images is as follows:

```
def areDistinct(arr):
    n = len(arr)
    # Put all array elements in a map
    s = set()
    for i in range(0, n):
        s.add(arr[i])

        # If all elements are distinct,
    # size of set should be same array.
    return (len(s) == len(arr))

# Driver code
arr = list(map(int,input().split()))
print(*arr, sep = "\n")
if (areDistinct(arr)):
    print("All Areas  are Distinct")
else:
    print("some Areas are Matching")
```

This is the code that helps us to determine the final outcome of this project i.e "All Areas are Distinct".

CHAPTER-6

RESULTS

The result includes various snapshots that are obtained while running the project.

6.1 SNAPSHOTS

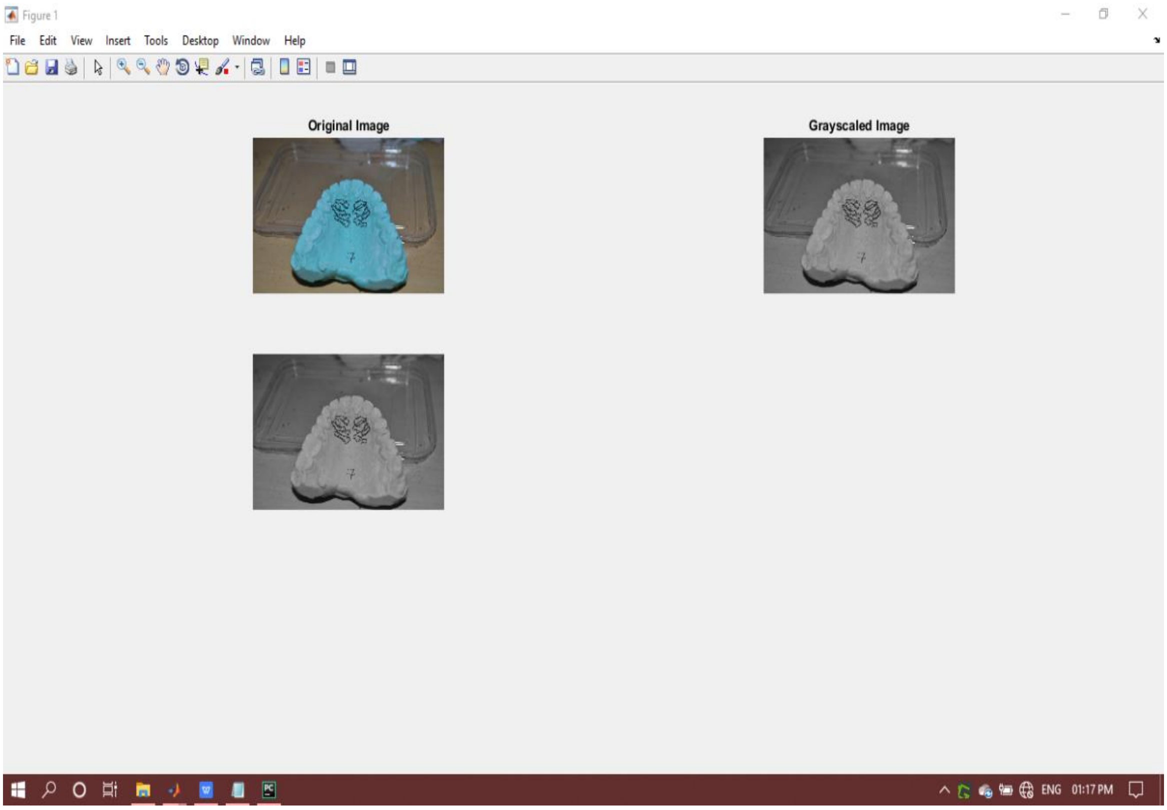

Figure 6.1 Conversion to Grayscale image and Enhancement

The Figure 6.1 screenshot is the MATLAB Command Window used for the displaying the output and provides various tools for different purpose. This screenshot indicates that the Original Image(RGB) is read into the workspace and is converted to the Grayscale Image and then the Grayscaled image is enhanced using the Median Filter which can be seen in the 3rd image.

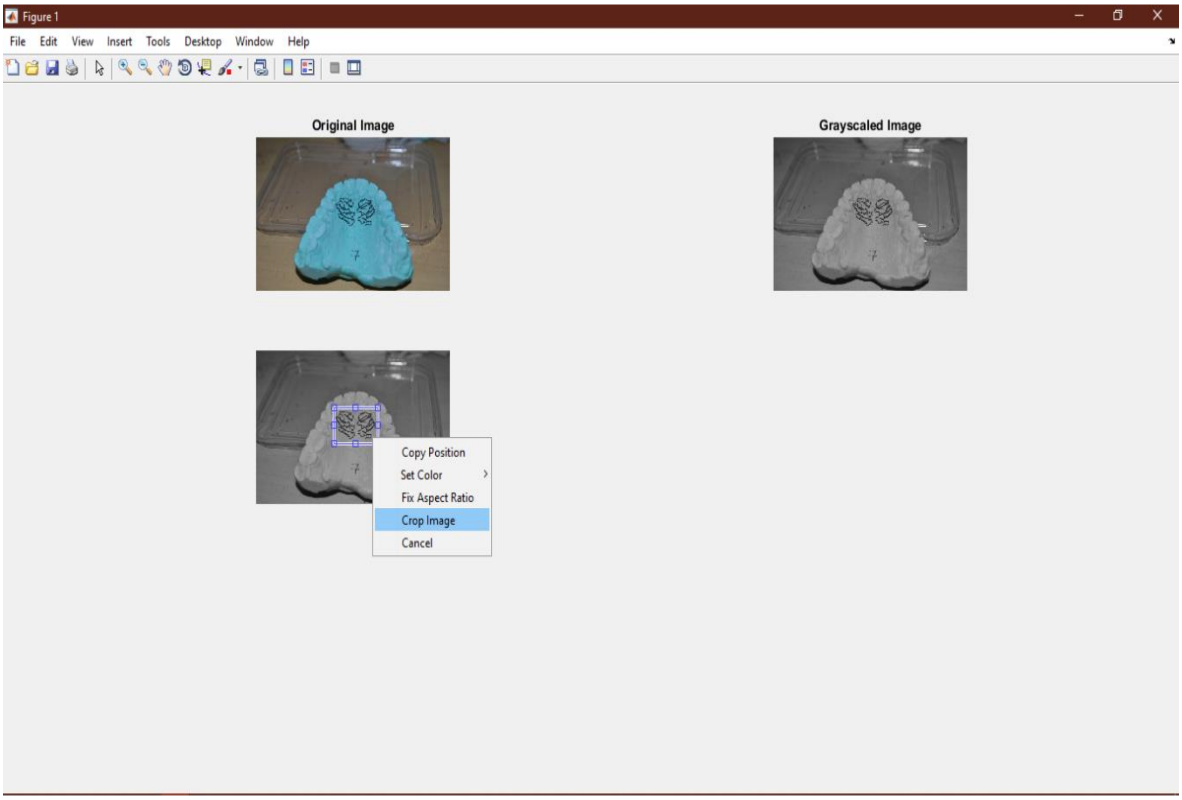

Figure 6.2 Cropping the enhanced image

In Figure 6.2, screenshot is the MATLAB Command Window used for the displaying the output and provides various tools for different purpose. This screenshot shows the working of imcrop function using Crop Interactive Tool. The crop interactive tool provides various functions in the menu such as “copy position”, “Crop Image” and so on. The Crop Image function is selected to crop the image according to the Region of Interest.

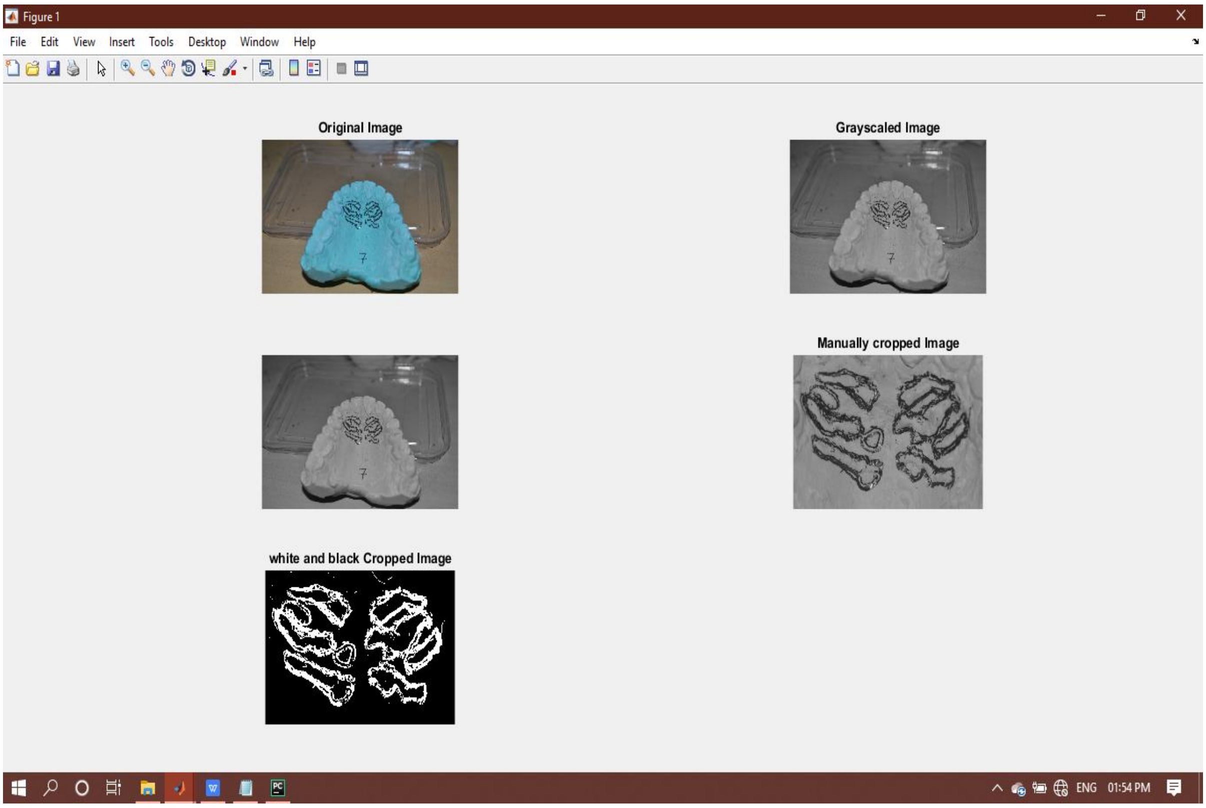

Figure 6.3 Conversion of cropped image to the Binary Image

The Figure 6.3, screenshot is the MATLAB Command Window used for the displaying the output and provides various tools for different purpose. This screenshot indicates the next step of the cropping. In this process the cropped image is being displayed and this image is converted into the Binary (Black and White) pixel image.

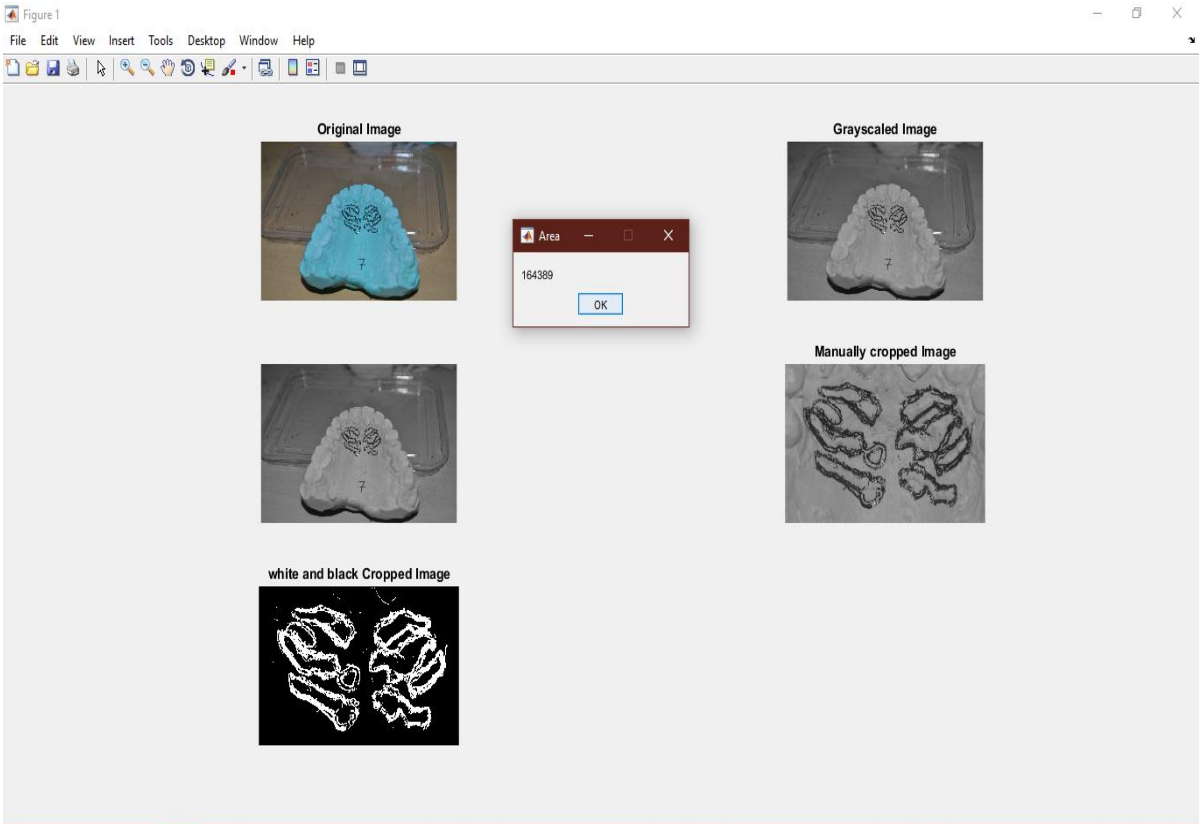

Figure 6.4 Calculation of Area of palatal rugae pattern

The Figure 6.4, screenshot is the MATLAB Command Window used for the displaying the output and provides various tools for different purpose. This screenshot indicates the process of inverting the Black and White image and calculating the white pixel count in the image that is Area of the Palatal rugae pattern.

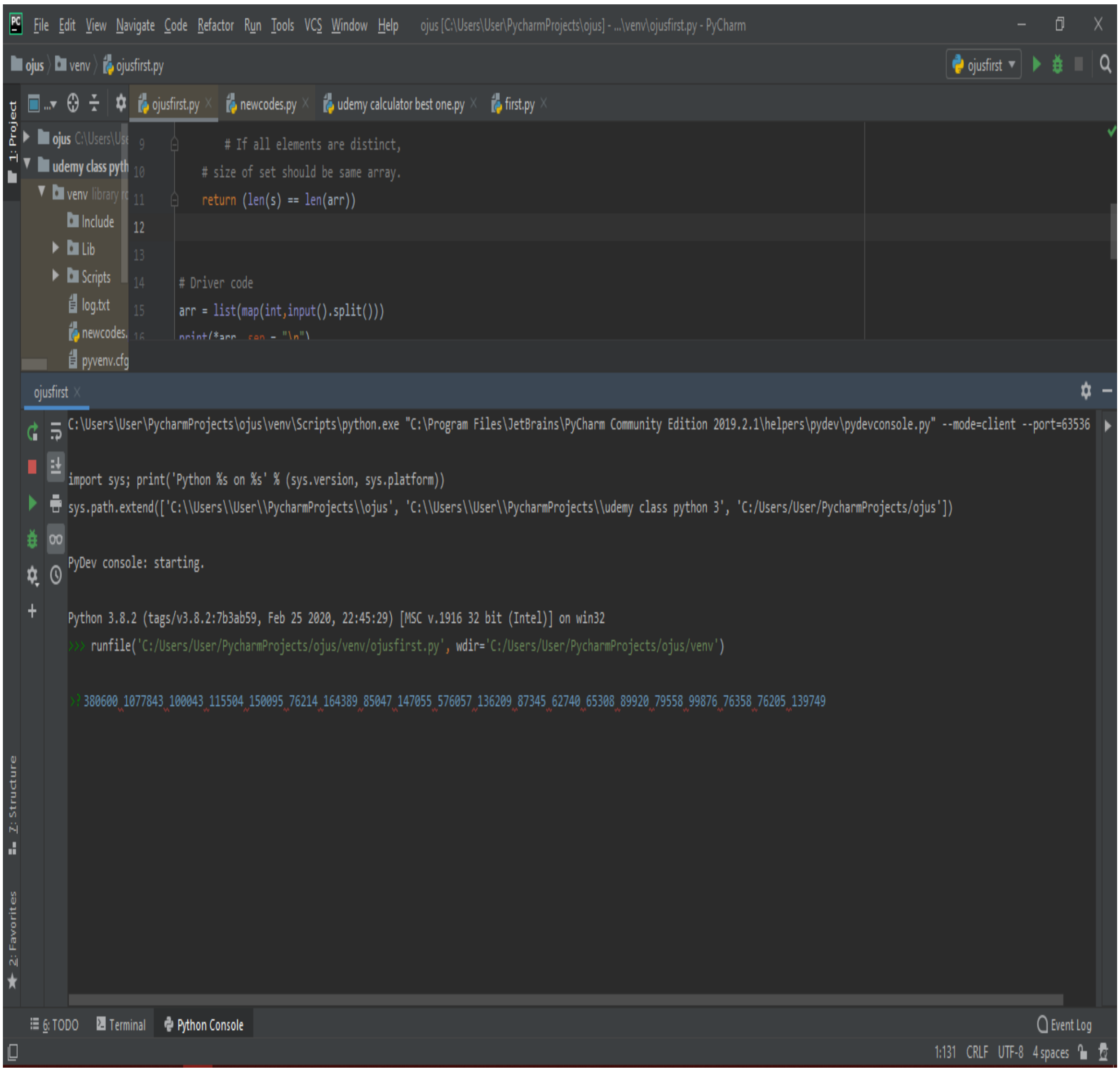

Figure 6.5 Feature Comparison using Python code Execution

The Figure 6.5, screenshot is the Python Console Window used for the displaying the output and provides the option for taking the input values for further Execution. This screenshot indicates the process of taking the input values i.e. derived Area values of 21 samples separated by space. These area values are stored in List[] for the further Execution of the python code.

6.2 OUTPUTS OF PROJECT FOR DIFFERENT SAMPLE DATA

We have taken 21 data samples that is dental cast images containing palatal rugae patterns and the respective Areas as calculated are given in the following table 6.1.

Data Sample	Calculated Area
Image 1	380600
Image 2	1077843
Image 3	100043
Image 4	115504
Image 5	150095
Image 6	76214
Image 7	164389
Image 8	85047
Image 9	147055
Image 10	576057
Image 11	87193
Image 12	136209
Image 13	87345
Image 14	62740
Image 15	65308
Image 16	89920
Image 17	79558
Image 18	99876
Image 19	76358
Image 20	76205
Image 21	139749

Table 6.1 Calculated Areas for Different Data Samples

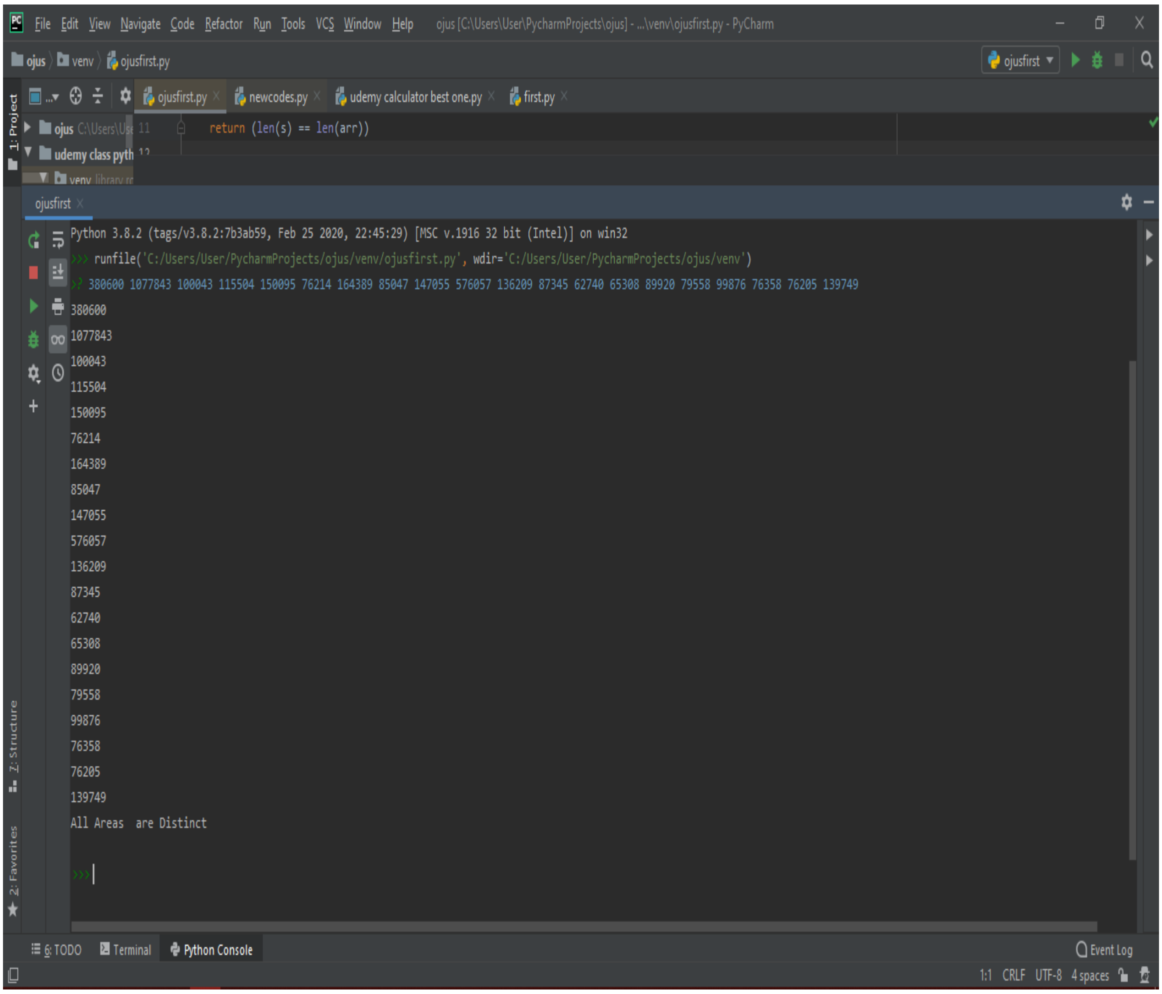

Figure 6.6 Final Output of the Project

The Figure 6.6 screenshot is the Python Console Window used for the displaying the output and provides the option for taking the input values for further Execution. This screenshot indicates that all the calculated Area values are different by providing the output "All Areas are Distinct".

CHAPTER 7

CONCLUSION

The technology advancement has evolved for providing new opportunities to work in different disciplines of science. In the present project, we have taken up one such area of medical science. In precise the process of identifying an individual has been an open-end problem for all the fraternities. The problem of an identifying an individual is important for many reasons since the identity of an individual is related to many issues.

In the present project work, we have palatal rugae which is a unique identity of an individual present in an upper jaw of mouth. The work has been done under the supervision of dental expert to understand the dental issues related to *Palatal Rugae*. The process of sample collection has been done by the dentist (Dr. Rakesh Pande M B, Maruthi Multi-Speciality Dental Care). A total of 200 samples were collected for the process. Out of 200 samples the best 21 samples were selected for computational algorithmic process.

As presented in the methodology chapter, the samples were from the age groups of 18 to 25. The pre-processing stages (Image processing) were implemented for identifying shapes present in palatal rugae. In figure 4.4, the selection of one such sample has been used to work as a case study. The Matlab has been used as a tool for shape recognition. The Histogram Equalization concept presented the Intensity of the pixel present in the palate. Further, the Median Filtering process was taken up to reduce the noise and preserve the edges of the shapes as presented in figure 4.8 and 4.9 respectively. In continuation the **crop** function was used to select the region of interest and **Otsu's** method was used to binarize the image.

In the next stage, the total area in which the shapes are present was computed to obtain the region of interest. The algorithmic for computing has been presented in fig 4.1.4. Further, the samples were compared to identify the matching/distinct property between all the samples. This was done using ***python language***.

The entire process of converting a cropped image to binary image is presented in figure 6.3. Similarly, the computation of area is presented in figure 6.4. The overall calculated area for all the 21 samples is presented in figure 6.6. This clearly indicates that the overall area of each sample *(human Identification)* is distinct and clearly separable. This also indicates the uniqueness of every individual. The outcome of all the 21 samples can be grouped based on age, gender which depends on the shapes present in the area.

Future Work:

- The process of identifying requires processing of each shapes present in the region of interest that is area and also its sub-area of each of the shape present in the palate.
- Also, it is important to identifying the number of shapes presented in two different sides of the embodied area that is if a central line is drawn the shape present in the left and right region will give much clarity.

Note:

Since the samples collected were 200 *(two hundred)* but we could use only 21 *(twenty-one)* samples to work on imaging concept, it is very much important to have a good process of capturing the palate from the individual. Hence a good collaborative approach from both dental and technical fraternity is required.

REFERENCES

[1] Harjeet Kaur Sekhon, Keya Sircar1, Sanjeet Singh, Deepti Jawa2, Priyanka Sharma "Determination of the biometric characteristics of palatine rugae patterns in Uttar Pradesh population: A cross-sectional study", Indian Journal of Dental Research, 2014

[2] Namrata Harchandani, Swati Marathe1, Rahul Rochani, Shams Ul Nisa1 "Palatal Rugoscopy: A new era for forensic identification" Journal of Indian Academy of Oral Medicine and Radiology (2015)

[3] Aparna Paliwal, Sangeeta Wanjari, and Rajkumar Parwani "Palatal rugoscopy: Establishing identity" Journal of Forensic Dent Sci. 2010

APPENDIX

- Forensic Odontology
- Rugoscopy
- Edentulous
- Palatal raphe
- MATLAB -matrix laboratory
- ROI – Region of Interest