CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION TO RETINA

The retina is the photosensitive (light sensitive) tissue that covers approximately 65% of the interior surface of the eye. The retina is not actually attached to the vascular choroid layer that it lays directly against, but is held in place by the pressure of a jelly-like fluid known as the vitreous humour that fills the chamber of the eye behind the lens.

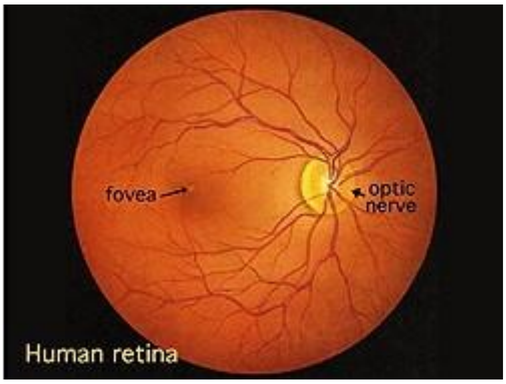

Figure 1.1: Retina

In the centre of the retina is the optic nerve, a circular to oval white area measuring about 2 x 1.5 mm across. From the centre of the optic nerve radiates the major blood vessels of the retina. Approximately 17 degrees (4.5-5 mm), or two and half disc diameters to the left of the disc, can be seen the slightly oval-shaped, blood vessel-free reddish spot, the fovea, which is at the centre of the area known as the macula by ophthalmologists.

A circular field of approximately 6 mm around the fovea is considered the central retina while beyond this is peripheral retina stretching to the ora serrata, 21mm from the centre of the retina (fovea). The total retina is a circular disc of between 30 and 40 mm in diameter. The retina works much like the film of a camera. It takes the visual information transmitted by beams of light reflecting off of objects and converts that information into a neural 'image' that it then transmits to the brain through the bundle of nerve fibers called the optic nerve.

When a beam of light enters the eye, it passes through accessory structures - such as the iris, pupil and lens - all of which serve the singular purpose of enabling that beam of light to reach the photosensitive retinal layer. It houses all of the photoreceptor cells, called rods and cones, which, when triggered by photons of light, result in a cascade of electrochemical events that generate a nerve impulse. It is this nerve impulse that, when received and translated by the brain, allows us the amazing feat of sight!

Rods are the specialized photoreceptor cells that allow us to see in dim lighting. You could think of them as the high-speed black and white film of the eye (back before the days of digital cameras) in that they are ultra-sensitive to dim light (enabling us to see gradations of blacks and greys in low light settings), as well as very sensitive to high-speed movement.

Retina contains about 125 million rod cells, giving you both a very acute awareness for anything trying to sneak up on you, as well as the ability of twilight sight (the ability for your eyes to adjust and see in dim lighting). What's interesting is that rods are completely insensitive to red light frequencies, so if someone were to shine a red frequency light in your eyes after your eyes had adjusted to the dark, your night vision would be undisturbed. This is why navigational instruments use red lights for night illumination.

Cones are the specialized photosensitive cells that allow us to see colour. Cones are like the low speed colour film of the eye because they are great at sensing bright light and detailed colours but are insensitive to low lights and high speed.

1.2. DIABETIC RETINOPATHY

In the recent years, there has been a gradual increase in the number of diabetic patients and it is approximately 65,000 million persons in India. Among the Diabetes related eye disease, Diabetic Retinopathy (DR) is the most chronic disease which affects nearly one out of every ten persons with diabetes, according to point prevalence estimates. DR is one of the most important reasons which make the key cause of vision loss, especially in middle-aged people. India has the highest number of diabetics in world with DR steering it to the 6th biggest cause of vision impairment in the country. In addition, this disease will experience a high growth in the future due to diabetes incidence increase and ageing population in the current society.

The early diagnosis allows, through appropriate treatment, to reduce costs generated when they are in advanced states and may become chronic. This fact justifies screening campaigns However, a screening campaign requires a heavy workload for trained experts in the analysis of anomalous patterns of each disease which, added to the at-risk population increase, makes these campaigns economically infeasible. Therefore, the need for automatic screening systems is highlighted. The working process of an automatic screening system follows an organized procedure in which the system tries to learn the characteristics of disease through the retinal images and then the testing is carried out.

The automatic system requires a clear and distortion free retinal image to learn its characteristics. Hence there is a need to develop an approach which produces a clear and distortion free image even though the image captured through camera is not clear and contains distortions. As a standard image modality, fundus camera is usually used to acquire retinal images, showing structures like optic disc, retinal vessels and several others. The changes detected in these structures indicate the pathological condition associated with diabetic retinopathy. Therefore, the analysis of retinal images is a useful and helpful diagnostic tool.

Diabetic retinopathy (DR), the major cause of poor vision, is an eye disease that is associated with longstanding diabetes. If the disease is detected in its early stages, treatment can slow down the progression of DR. However, this is not an easy task, as DR often has no early warning signs. Earliest signs of DR are damages of the blood vessels and then formation of lesions.

Lesions such as exudates are normally detected and graded manually by clinicians in time consuming and it is susceptible to observer error. Diabetic retinopathy results from the leakage of small vessels in the retina correlated to a prolonged period of hyperglycemia. In the early stages of the disease, known as non-proliferative retinopathy, there may be hemorrhages due to bleeding of the capillaries or exudates resulting from protein deposits in the retina. There is usually no vision loss unless there is a build-up of fluid in the center of the eye. As the disease progresses, new abnormal vessels grow in the retina, known as revascularization. This stage of the disease is called proliferative retinopathy and may cause severe visual problems.

1.3 EXUDATES:

Exudates are common abnormalities in the retina of diabetic patients. Exudates are bright lipids leaked from a blood vessel. The leaked fluid tends to stay close to the lesion, giving a generally well-defined edge suitable for computer analysis. Figure (1) gives an example of exudates, which show up as small, light yellow region.

Figure 1.2: Exudate

The chief cause of exudates is leaking of proteins and lipids from the bloodstream into the retina through damaged blood vessels. In retinal images, exudate exhibits as hard white or yellowish localized regions with varying sizes, shapes and locations. Generally, they materialize near the leaking capillaries within the retina. The hard exudates are formations of lipid that are leaking from these weakened blood vessels. This kind of the retinopathy is termed as nonproliferative diabetic retinopathy. Optic disk is also bright yellow region which have similar appearance of exudates. The optic disk, which can be seen in Figure 1, is also a light-yellow region. Therefore, before searching for exudates based on their yellow color, an algorithm is developed for automatic detection of the optic disk to eliminate this physiologically valid, yet it has similar structure.

The localization of the optic disk as the identification of the center of disk is either by specifying the center of the optic disk or placing a mask within a particular region of the retina. Segmentation of the optic disk usually refers to the subsequent task of determining the contour of the disk. Localization and segmentation of the optic disk are important tasks in retinal image Detection of exudates by computer could offer fast and precise diagnosis to specialist inspection. Also, it assists the clinician to take timely the right treatment decision.

CHAPTER 2

BACKGROUND CONCEPTS

2.1 DIGITAL IMAGE PROCESSING

Digital Image Processing means processing digital image by means of a digital computer. 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 processing mainly include the following steps

a. Importing the image via image acquisition tools.
b. Analyzing and manipulating the image.
c. Output in which result can be altered image or a report, which is based on analyzing that image.

2.1.1 Fundamentals of Digital Image

Image:

An image is a two-dimensional picture, which has a similar appearance to some subject usually a physical object or a person. An image is defined as a two-dimensional function, F(x, y), where x and y are spatial coordinates, and the amplitude of F at any pair of coordinates (x,y) is called the intensity of that image at that point. When x, y, and amplitude values of F are finite, we call it a digital image.

In other words, an image can be defined by a two-dimensional array specifically arranged in rows and columns. Digital Image is composed of a finite number of elements, each of which elements have a particular value at a particular location. These elements are referred to as picture elements, image elements, and pixels. A Pixel is most widely used to denote the elements of a Digital Image.

Classification of Images:

There are three types of images used in Digital Image Processing as follows

A. Binary Image

B. Gray Scale Image

C. Color Image

A. Binary image

A binary image is a digital image that has only two possible values for each pixel. Typically, the two colors used for a binary image are black and white though any two colors can be used. The color used for the object(s) in the image is the foreground color while the rest of the image is the background color.

Binary images are also called bi-level or two-level. This means that each pixel is stored as a single bit (0 or 1). This name black and white, monochrome or monochromatic are often used for this concept, but may also designate any images that have only one sample per pixel, such as grayscale images.

Binary images often arise in digital image processing as masks or as the result of certain operations such as segmentation, thresholding, and dithering. Some input/output devices, such as laser printers, fax machines, and bi-level computer displays, can only handle bi-level images.

B. Gray scale image

A grayscale Image is digital image is an image in which the value of each pixel is a single sample, that is, it carries only intensity information. Images of this sort, also known as black-and-white, are composed exclusively of shades of gray (0-255), varying from black (0) at the weakest intensity to white (255) at the strongest. Grayscale images are distinct from one-bit black-and-white images, which in the context of computer imaging are images with only the two colors, black, and white (also called bi-level or binary images). Grayscale images have many shades of gray in between. Grayscale images are also called monochromatic, denoting the absence of any chromatic variation. Grayscale images are often the result of measuring the intensity of light at each pixel in a single band of the electromagnetic spectrum (e.g. infrared, visible light, ultraviolet, etc.), and in such cases they are monochromatic proper when only a given frequency is captured.

C. Color image

A (digital) color image is a digital image that includes color information for each pixel. Each pixel has a particular value, which determines it is appearing color. Three numbers giving the decomposition of the color in the three primary colors Red, Green and Blue qualify this value. Any color visible to human eye can be represented this way. A number between 0 and 255 quantifies the decomposition of a color in the three primary colors. For example, white will be coded as R = 255, G = 255, B = 255; black will be known as (R, G, B) = (0,0,0); and say, bright pink will be: (255,0,255). In other words, an image is an enormous two-dimensional array of color values, pixels, each of them coded on 3 bytes, representing the three primary colors. This allows the image to contain 256x256x256 = 16.8 million different colors. This technique is also known as RGB encoding and is specifically adapted to human vision.

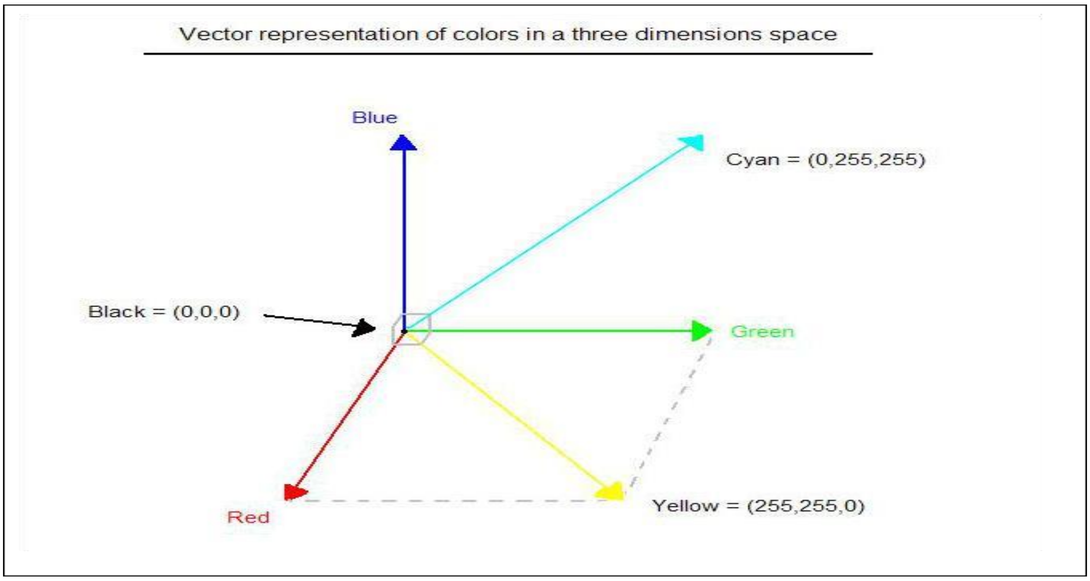

Figure 2.1: Hue Saturation Process of RGB SCALE Image

From the above figure, colors are coded on three bytes representing their decomposition on the three primary colors. It sounds obvious to a mathematician to immediately interpret colors as vectors in a three-dimension space where each axis stands for one of the primary colors. Therefore, we will benefit of most of the geometric mathematical concepts to deal with our colors, such as norms, scalar product, projection, rotation or distance.

Figure 2.2: Colour image to Gray scale Conversion Process

An image is a rectangular grid of pixels. It has a definite height and a definite width counted in pixels. Each pixel is square and has a fixed size on a given display. However, different computer monitors may use different sized pixels. The pixels that constitute an image are ordered as a grid (columns and rows); each pixel consists of numbers representing magnitudes of brightness and color.

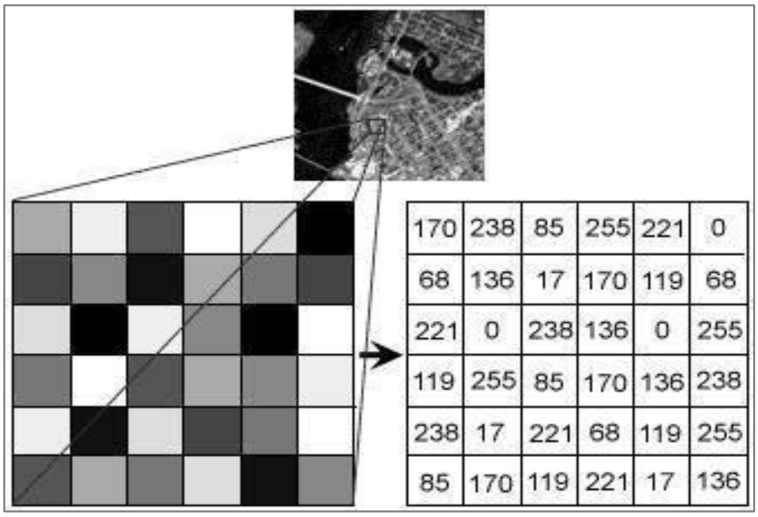

Figure 2.3: Gray Scale Image Pixel Value Analysis

Each pixel has a color. The color is a 32-bit integer. The first eight bits determine the redness of the pixel.

The next eight bits the greenness, the next eight bits the blueness, and the remaining eight bits the transparency of the pixel.

2.1.2 Image file sizes

Image file size is expressed as the number of bytes that increases with the number of pixels composing an image, and the color depth of the pixels. The greater the number of rows and columns, the greater the image resolution, and the larger the file. In addition, each pixel of an image increases in size when its color depth increases, an 8bit pixel (1 byte) stores 256 colors, a 24-bit pixel (3 bytes) stores 16 million colors, the latter known as true color. Image compression uses algorithms to decrease the size of a file. High-resolution cameras produce large image files, ranging from hundreds of kilobytes to megabytes, per the camera's resolution and the image-storage format capacity. High-resolution digital cameras record 12-megapixel (1MP = 1,000,000 pixels / 1 million) images, or more, in true color. For example, an image recorded by a 12 MP camera; since each pixel uses 3 bytes to record true color, the uncompressed image would occupy 36,000,000 bytes of memory, a great amount of digital storage for one image, given that cameras must record and store many images to be practical. Faced with large file sizes, both within the camera and a storage disc, image file formats were developed to store such large images.

2.1.3 Image processing

Digital image processing, the manipulation of images by computer, is relatively recent development in terms of man’s ancient fascination with visual stimuli. In its short history, it has been applied to practically every type of images with varying degree of success. The inherent subjective appeal of pictorial displays attracts perhaps a disproportionate amount of attention from the scientists and from the nonprofessional. Digital image processing like other glamour fields, suffers from myths, disconnections, misunderstandings and misinformation. It is vast umbrella under which fall diverse aspect of optics, electronics, mathematics, photography graphics and computer technology.

Several factors combine to indicate a lively future for digital image processing. A major factor is the declining cost of computer equipment. Several new technological trends promise to further promote digital image processing. These include parallel processing mode practical by low cost microprocessors, and the use of charge coupled devices (CCDs) for digitizing, storage during processing and display and large low cost of image storage arrays.

2.2 FUNDAMENTAL STEPS IN DIGITAL IMAGE PROCESSING

2.2.1 Image Acquisition

Image Acquisition is to acquire a digital image. To do so requires an image sensor and the capability to digitize the signal produced by the sensor. The sensor could be monochrome or color TV camera that produces an entire image of the problem domain every 1/30 sec. The image sensor could also be line scan camera that produces a single image line at a time. In this case, the objects motion past the line.

Figure 2.4: Digital camera

Scanner produces a two-dimensional image. If the output of the camera or other imaging sensor is not in digital form, an analog to digital converter digitizes it.

The nature of the sensor and the image it produces are determined by the application.

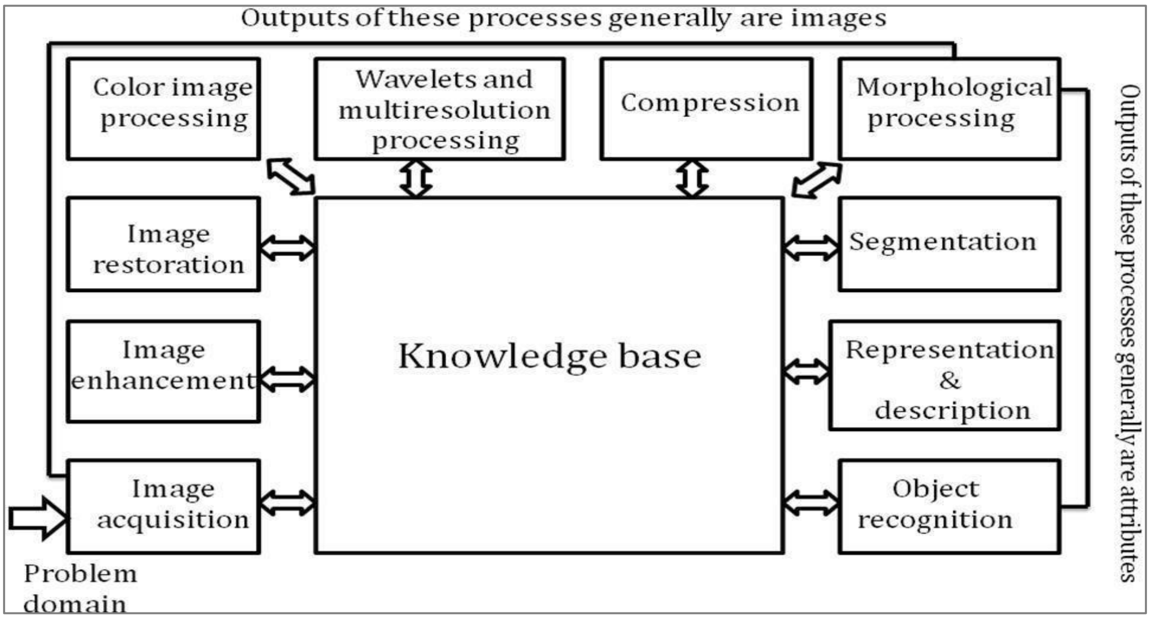

Figure 2.5: Basics steps of image Processing

2.2.2 Image Enhancement

Image enhancement is among the simplest and most appealing areas of digital image processing. The idea behind enhancement techniques is to bring out detail that is obscured, or simply to highlight certain features of interesting an image.

A familiar example of enhancement is when we increase the contrast of an image because "it looks better." It is important to keep in mind that enhancement is a very subjective area of image processing. Image enhancement helps us to create clearer image that means we are going to enhance the picture by using some techniques.

Figure 2.6: Image enhancement process for Gray Scale Image and Colour Image using Histogram Bits

2.2.3 Image restoration

Image restoration is an area that also deals with improving the appearance of an image. However, unlike enhancement, which is subjective, image restoration is objective, in the sense that restoration techniques tend to be based on mathematical or probabilistic models of image degradation.

Figure 2.7: Noise Image to Image Enhancement

2.2.4 Color image processing

The use of colour in image processing is motivated by two principal factors. First, color is a powerful descriptor that often simplifies object identification and extraction from a scene. Second, humans can discern thousands of color shades and intensities, compared to about only two dozen shades of grey. This second factor is particularly important in manual image analysis.

Figure 2.8: Grey scale image to Colour Image

2.2.5 Segmentation

Segmentation procedures partition an image into its constituent parts or objects. In general, autonomous segmentation is one of the most difficult tasks in digital image processing. A rugged segmentation procedure brings the process a long way toward successful solution of imaging problems that require objects to be identified individually.

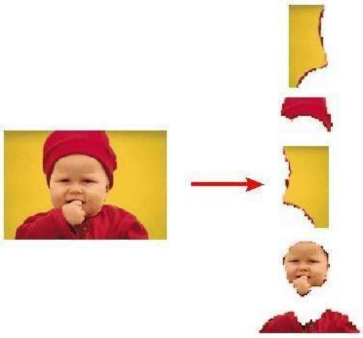

Figure 2.9: Image Segment Process

On the other hand, weak or erratic segmentation algorithms usually guarantee eventual failure. In general, the more accurate the segmentation, the more likely recognition is to succeed. Digital image is defined as a two-dimensional function f (x, y), where x and y are spatial (plane)coordinates, and the amplitude off at any pair of coordinates (x, y) is called intensity or grey level of the image at that point. The field of digital image processing refers to processing digital images by means of a digital computer. The digital image is composed of a finite number of elements, each of which has a particular location and value. The elements are referred to as picture elements, image elements, pels, and pixels. Pixel is the term most widely used.

2.2.6 Image Compression

Digital Image compression addresses the problem of reducing the amount of data required to represent a digital image. The underlying basis of the reduction process is removal of redundant data. From the mathematical viewpoint, this amounts to transforming a 2D pixel array into a statically uncorrelated data set. The data redundancy is not an abstract concept but a mathematically quantifiable entity. If n1 and n2 denote the number of information-carrying units in two data sets that represent the same information, the relative data redundancy R_D of the first data set (the one characterized by n1) can be defined as,

$$R_D = 1-(1/C_R)$$

Where C_R called as compression ratio. It is defined as

$$C_R = \frac{n_1}{n_2}$$

In image compression, three basic data redundancies can be identified and exploited: Coding redundancy, interpixel redundancy, and psychovisual redundancy. Image compression is achieved when one or more of these redundancies are reduced or eliminated. The image compression is mainly used for image transmission and storage. Image transmission applications are in broadcast television; remote sensing via satellite, aircraft, radar, or sonar; teleconferencing; computer communications; and facsimile transmission. Image storage is required most commonly for educational and business documents, medical images that arise in computer tomography (CT), magnetic resonance imaging (MRI) and digital radiology, motion pictures, satellite images, weather maps, geological surveys, and so on.

Image Compression Model

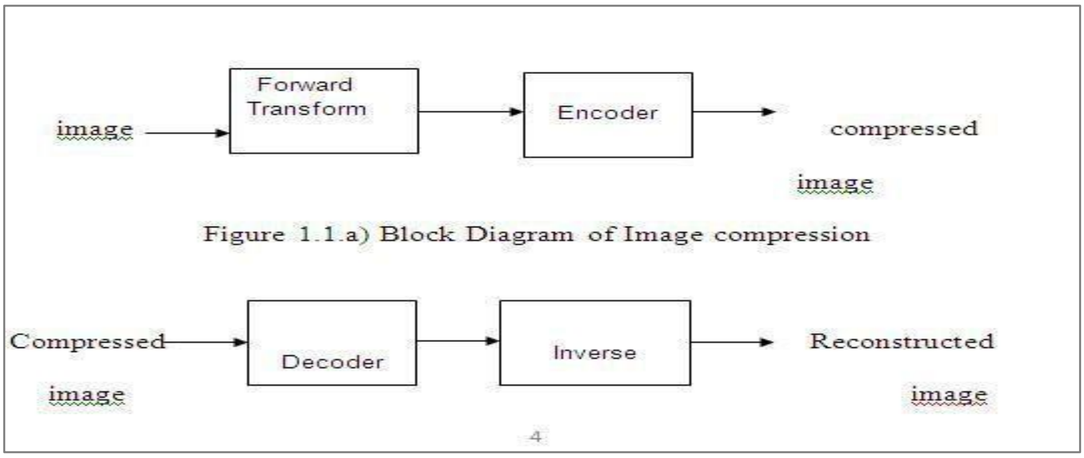

Figure 2.10: Decompression Process for Image

Image Compression Types: There are two types' image compression techniques

a. Lossy Image compression
b. Lossless Image

Compression ratio = B_0/B_1

Bo=number of bits before compression.

B1=number of bits after compression.

a) Lossy Image compression

Lossy compression provides higher levels of data reduction but result in a less than perfect reproduction of the original image. It provides high compression ratio. lossy image compression is useful in applications such as broadcast television, videoconferencing, and facsimile transmission, in which a certain amount of error is an acceptable trade-off for increased compression performance.

Originally, PGF has been designed to quickly and progressively decode lossy compressed aerial images. A lossy compression mode has been preferred, because in an application like a terrain explorer texture data (e.g., aerial orthophotos) is usually mid-mapped filtered and therefore lossy mapped onto the terrain surface. In addition, decoding lossy compressed images is usually faster than decoding lossless compressed images.

In the next test series, we evaluate the lossy compression efficiency of PGF. One of the best competitors in this area is for sure JPEG 2000. Since JPEG 2000 has two different filters, we used the one with the better trade-off between compression efficiency and runtime. On our machine the 5/3 filter set has a better trade-off than the other. However, JPEG 2000 has in both cases a remarkable good compression efficiency for very high compression ratios but also a very poor encoding and decoding speed. The other competitor is JPEG. JPEG is one of the most popular image file formats.

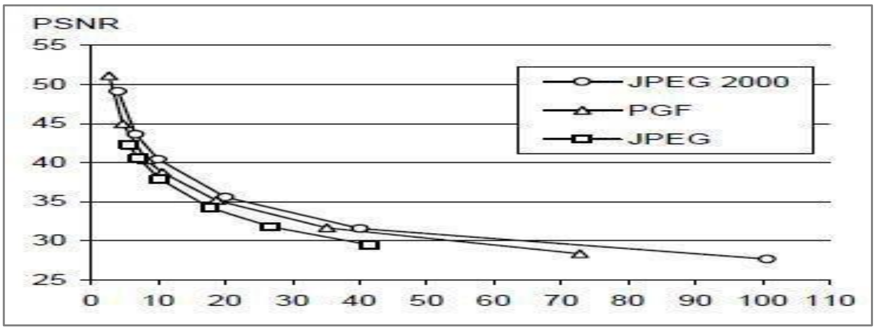

Figure 2.11: PSNR of lossy compression in relation to compression ratio

It is very fast and has a reasonably good compression efficiency for a wide range of compression ratios. The drawbacks of JPEG are the missing lossless compression and the often-missing progressive decoding. Above figure, depicts the average rate-distortion behaviour for the images in the Kodak test set when fixed (i.e., nonprogressive) lossy compression is used. The PSNR of PGF is on average 3% smaller than the PSNR of JPEG 2000, but 3% better than JPEG.

These results are also qualitative valid for our PGF test set and they are characteristic for aerial Orth-photos and natural images. Because of the design of PGF, we already know that PGF does not reach the compression efficiency of JPEG 2000. However, we are interested in the trade-off between compression efficiency and runtime. To report this trade-off, we show JPEG 2000 and PGF test series in above figure, the corresponding average decoding times in relation to compression ratios. Below table contains for seven different compression ratios (mean values over the compression ratios of the eight images of the Kodak test set) the corresponding average encoding and decoding times in relation to the average PSNR values.

In case of PGF, the encoding time is always slightly longer than the corresponding decoding time. The reason for that is that the actual encoding phase (cf. Subsection 2.4.2) takes slightly longer than the corresponding decoding phase. For six of seven ratios the PSNR difference between JPEG 2000 and PGF is within 3% of the PSNR of JPEG 2000. Only in the first row is the difference larger (21%), but because a PSNR of 50 corresponds to an almost perfect image quality, the large PSNR difference corresponds with an almost undiscoverable visual difference. The price they pay in JPEG 2000 for the 3% more PSNR is very high.

The creation of a PGF is five to twenty times faster than the creation of a corresponding JPEG 2000 file, and the decoding of the created PGF is still five to ten times faster than the decoding of the JPEG 2000 file. This gain in speed is remarkable, especially in areas where time is more important than quality, maybe for instance in real-time computation.

We see that the price we pay in PGF for the 3% more PSNR than JPEG is low: for small compression ratios (< 9) decoding in PGF takes two times longer than JPEG and for higher compression ratios (> 30) it takes only ten percent longer than JPEG. These test results are characteristic for both natural images and aerial ortho-photos. Again, in the third test series we only use the 'Lena' image.

Table 2.1: Trade-off between quality and speed for the kodak test set

Ratio	JPEG 2000 5/3			PGF		
	Encoder	Decoder	PSNR	Encoder	Decoder	PSNR
2.7	1.86	1.35	64.07	0.34	0.27	51.1
4.8	1.75	1.14	47.08	0.27	0.21	44.95
8.3	1.68	1.02	41.98	0.22	0.18	40.39
10.7	1.68	0.98	39.95	0.14	0.13	38.73
18.7	1.61	0.92	36.05	0.12	0.11	35.18
35.1	1.57	0.87	32.26	0.1	0.09	31.67
72.9	1.54	0.85	28.86	0.08	0.08	28.37

We run our lossy coder with six different quantization parameters and measure the PSNR in relation to the resulting compression ratios. The results (ratio: PSNR) are

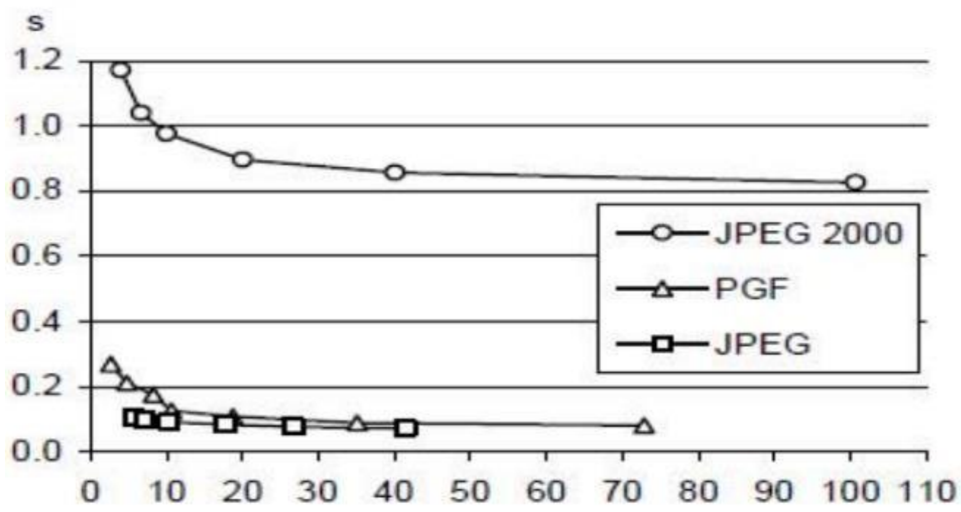

Figure 2.12: Decoding time in relation to compression ratio

b) Lossless Image compression

Lossless Image compression is the only acceptable amount of data reduction. It provides low compression ratio while compared to lossy. In Lossless Image, compression techniques are composed of two relatively independent operations: (1) devising an alternative representation of the image in which its interpixel redundancies are reduced and (2) coding the representation to eliminate coding redundancies.

Lossless Image compression is useful in applications such as medical imaginary, business documents and satellite images. Table2.3 summarizes the lossless compression efficiency and Table 2.4 the coding times of the PGF test set. For WinZip we only provide average runtime values, because of missing source code we have to use an interactive testing procedure with runtimes measured by hand. All other values are measured in batch mode.

Table 2.2: Lossless compression ratios of PGF (Placental Growth Factor) test set.

	WIN ZIP	JPEGLS	JPEG 2000	PNG	PGF
Aerial	1.352	2.073	2.383	1.944	2.314
Compound	12.451	6.802	6.068	13.292	4.885
Hibiscus	1.816	2.2	2.822	2.087	2.538
Houses	1.241	1.518	2.155	1.5	1.965
Logo	47.128	16.28	12.959	50.676	10.302
Red brush	2.433	4.041	4.494	3.546	3.931
woman	1.577	1.92	2.564	1.858	2.556
Average	**9.71**	**4.98**	**4.78**	**10.7**	**4.07**

In Table 2.2, it can be seen that in almost all cases the best compression ratio is obtained by JPEG 2000, followed by PGF, JPEG-LS, and PNG. This result is different to the result in [SEA+00], where the best performance for a similar test set has been reported for JPEG-LS. PGF performs between 0.5% (woman) and 21.3% (logo) worse than JPEG 2000. On average, it is almost 15% worse. The two exceptions to the general trend are the 'compound' and the 'logo' images. Both images contain for the most part black text on a white background. For this type of images, JPEG-LS and in particular WinZip and PNG, provide much larger compression ratios. However, in average PNG performs the best, which is also reported in [SEA+00].

These results show that as far as lossless compression is concerned, PGF performs reasonably well on natural and aerial images. In specific types of images such as ‘compound’ and ‘logo’, PGF is outperformed by far in PNG.

Table 2.3: Runtime of lossless compression of the PGF test set

	JPEG-LS		JPEG 2000		PNG		PGF	
	encoder	decoder	encoder	decoder	encoder	decoder	encoder	decoder
a	1.11	0.8	5.31	4.87	3.7	0.19	0.99	0.77
c	1.61	0.38	3.46	3.06	2.95	0.18	0.95	0.8
hi	0.69	0.3	1.45	1.29	1.77	0.1	0.35	0.27
ho	0.65	0.3	1.62	1.47	0.85	0.1	0.41	0.32
l	0.09	0.02	0.26	0.21	0.16	0.01	0.07	0.06
r	0.65	0.44	4.29	4.01	3.61	0.16	0.66	0.59
w	0.39	0.3	1.76	1.63	1.08	0.08	0.35	0.27
avg	**0.74**	**0.36**	**2.59**	**2.36**	**2.02**	**0.12**	**0.54**	**0.44**

Table 2.3 shows the encoding (enc) and decoding (dec) times (measured in seconds) for the same algorithms and images. JPEG 2000 and PGF are both symmetric algorithms, while WinZip, JPEG-LS and in particular, PNG are asymmetric with a clearly shorter decoding than encoding time. JPEG 2000, the slowest in encoding and decoding, takes more than four times longer than PGF. This speed gain is due to the simpler coding phase of PGF. JPEG-LS is slightly slower than PGF during encoding, but slightly faster in decoding images. WinZip and PNG decode even more faster than JPEG-LS, but their encoding times are also worse. PGF seems to be the best compromise between encoding and decoding times.

Our PGF test set clearly shows that PGF in lossless mode is best suited for natural images and aerial ortho photos. PGF is the only algorithm that encodes the three Mega Byte large aerial ortho photo in less than second without a real loss of compression efficiency. For this particular image the efficiency, loss is less than three percent compared to the best.

These results should be underlined with our second test set, the Kodak test set.

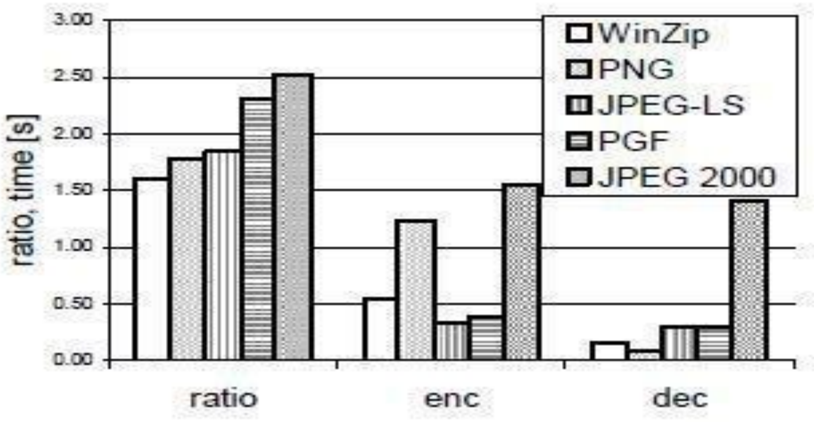

Figure2.13: Lossless compression results of the kodak test set.

Above figure shows the averages of the compression ratios (ratio), encoding (enc), and decoding (dec) times over all eight images. JPEG 2000 shows in this test set the best compression efficiency followed by PGF, JPEG-LS, PNG, and WinZip. In average PGF is eight percent worse than JPEG 2000. The fact that JPEG 2000 has a better lossless compression ratio than PGF does not surprise, because JPEG 2000 is more quality driven than PGF.

However, it is remarkable that PGF is clearly better than JPEG-LS (+21%) and PNG (+23%) for natural images. JPEG-LS shows in the Kodak test set also a symmetric encoding and decoding time behavior. It is encoding and decoding times are almost equal to PGF. Only PNG and WinZip can faster decode than PGF, but they also take longer than PGF to encode.

If both compression efficiency and runtime is important, then PGF is clearly the best of the tested algorithms for lossless compression of natural images and aerial ortho photos. In the third test we perform our lossless coder on the 'Lena' image.

To digitally process an image, it is first necessary to reduce the image to a series of numbers that can be manipulated by the computer.

Each number representing the brightness value of the image at a particular location is called a picture element, or pixel. A typical digitized image may have 512 × 512 or roughly 250,000 pixels, although much larger images are becoming common. Once the image has been digitized, three basic operations can be performed on it in the computer. For a point operation, a pixel value in the output image depends on a single pixel value in the input image. For local operations, several neighbouring pixels in the input image determine the value of an output image pixel. In a global operation, all of the input image pixels contribute to an output image pixel value.

Correspondingly, these combinations attempt to strike a winning trade-off: be flexible and hence bring tolerance toward infraclass variation, while also being discriminative enough to be robust to background clutter and interclass similarity. An important feature of our contour-based recognition approach is that it affords us substantial flexibility to incorporate additional image information. Specifically, we extend the contour-based recognition method and propose a new hybrid recognition method, which exploits shape tokens, and SIFT features as recognition cues. Shape tokens and SIFT features are largely ¨orthogonal, where the former corresponds to shape boundaries and the latter to sparse salient image patches.

Here, each learned combination can comprise features that are either 1) purely shape-tokens, 2) purely SIFT features, or 3) a mixture of shape-tokens and SIFT features. The number and types of features to be combined together are learned automatically from training images and represent the more discriminative ones based on the training set. Consequently, by imparting these two degrees of variability (in both the number and the types of features) to a combination, we empower it with even greater flexibility and discriminative potential.

2.2.7 Wavelets and Multi-Resolution Processing

It is foundation of representing images in various degrees. A scaling function scaling function is used to create a series of approximations of a function or image, each differing by a factor of 2 from its neighbouring approximations. Additional functions called wavelets are then used to encode the difference in information between adjacent approximations.

2.2.8 Morphological Processing

Morphological processing deals with tools for extracting image components that are useful in the representation and description of shape.

2.2.9 Representation and Description

Representation and description usually follow the output of a segmentation stage, which usually is raw pixel data, constituting either the boundary of a region or all the points in the region itself. It follows output of segmentation stage, choosing a representation is only the part of solution for transforming raw data into processed data. Choosing a representation is only part of the solution for transforming raw data into a form suitable for subsequent computer processing. Description deals with extracting attributes that result in some quantitative information of interest or are basic for differentiating one class of objects from another.

2.2.10 Object Recognition

Recognition is the process that assigns a label, such as, “vehicle” to an object based on its descriptors.

2.2.11 Knowledge Base

Knowledge may be as simple as detailing regions of an image where the information of interest is known to be located, thus limiting the search that has to be conducted in seeking that information.

CHAPTER 3
LITERATURE SURVEY

1.Title: Analysis of Fundus Image of Ophthalmoscope for Macula Identification and Detection to Diagnosis of Vision Related Diseases– by Gray Thresholding and Pixel Index Number.

Author: Vaishali B. Shinde Dept. Ramesh R. Manza Dept of Computer Science & IT Dr. B. A. M. University, Aurangabad.

Diabetic disease has also infected the other parts of the human body, like eye, platelets counts etc. In the field of medicines, medical image processing plays a vital role to detect the abnormalities of eye or eye diseases like glaucoma and diabetic retinopathy. The retina is responsible for peripheral vision. Macula is the central area of the retina, temporal to the optic disk. It is responsible to have fine central vision and colour vision with macular degeneration, a scotoma, an isolated retinal area of diminished vision, may appear in the central visual field. The visual system of a person with a central scotoma from a sensory default is thought to choose a preferred eccentric retinal area to perform the visual tasks that the non-functioning fovea used to perform. In ophthalmology, for diagnosis of diabetes, digital colour retinal images are becoming increasingly important. In computer based retinal image analysis system, image processing techniques are used in order to facilitate and improve diagnosis. Manual analysis of the images can be improved and problem of detection of diabetes in the late stage for optimal treatment may be resolved. The Image Ret database was made publicly available in 2008 and is subdivided into two sub-databases, DIARETDB0 and DIARETDB1. DIARETDB0 contains 130 retinal images of which 20 are normal and 110 contain various symptoms of diabetic retinopathy.

2.Title: Fusion of Pixel and Texture Features to Detect Pathological Myopia.

Author: Beng-Hai Lee1, Damon Wing Kee Wong1, Ngan Meng Tan1, Zhuo Zhang1, Joo Hwee Lim1, Huiqi Li1, Jiang Liu1, Weimin Huang1, Seang Mei Saw2, Louis Hak Tien Tong3, Tien Yin Wong31Institute for Infocomm Research, A*STAR, Singapore.

Myopia, or near sightedness, is a huge public health problem with the economic and medical costs of myopia enormous and often under-estimated. In simple refractive myopia, the images are focused in a location in front of the retina, leading to defocused images when projected onto the retina, resulting in blurred vision. In cases where the degree of myopia is extremely high, there is a possibility of the occurrence of pathological myopia. Pathological myopia differs from simple refractive myopia, and is defined as myopia caused by pathologic axial elongation. Also known as “degenerative myopia”, pathological myopia can be accompanied by straightened and stretched vessels, temporal peripapillary atrophic crescent, tilting of the optic disc, posterior staphyloma, lacquer cracks in the Bruch’s membrane, geographic areas of atrophy of the retinal pigment epithelium and choroids, sub retinal haemorrhage, and choroidal neo vascularisation. Myopia-related visual impairment may affect the productivity, mobility, quality of life and activities of daily living of individuals. Potentially blinding pathologic myopia is often irreversible in nature, especially if diagnosed late. In the Shihpai Eye Study of elderly Taiwanese Chinese aged 65 years. or older [2], myopic macular degeneration was the second leading contributing cause of visual impairment (12.5%) after cataract (41.7%). The risks of visual loss in myopia are sufficiently high to warrant measures to prevent pathologic myopia. The current methods for assessment of a patient for pathologic myopia are still largely reliant on manual efforts. This limits the applicability of such methods for screening efforts, though much desired. A fully automatic method for the assessment of retinal images for pathologic myopia has yet to be reported and yet very much in need, due to the large social impact of myopia in many societies today. To our knowledge no other systems for pathological myopia detection has been described. or older [2], myopic macular degeneration was the second leading contributing cause of visual impairment (12.5%) after cataract (41.7%).

The risks of visual loss in myopia are sufficiently high to warrant measures to prevent pathologic myopia. The current methods for assessment of a patient for pathologic myopia are still largely reliant on manual efforts. This limits the applicability of such methods for screening efforts, though much desired. A fully automatic method for the assessment of retinal images for pathologic myopia has yet to be reported and yet very much in need, due to the large social impact of myopia in many societies to day. To our knowledge no other systems for pathological myopia detection has been described.

3.Title: A Thorough Investigation on Automated Diagnosis of Glaucoma.

Author: S.Karthikeyan*1,Dr.N.Rengarajan2*1 Research Scholar/ECE, Principal, K.S.R. College of Engineering, Tiruchengode, TamilNadu,

GLAUCOMA is one of the well-known reasons of blindness with a mean occurrence of 4.2% for the people above 60 years. This disease is featured by transformations in the eye ground (fundus) in the region of the Optic Nerve Head (ONH): (i) enlargement of the excavation, (ii)disc haemorrhage , (iii) thinning of the neuro retinal rim, (iv)asymmetry of the cup between left and right eye, (v)loss of retina nerve fibre's, (vi)Appearance of parapapillary atrophy. The progressive loss of retinal nerve fibres in the parapapillary region provokes the glaucoma. Though, those lost fibers cannot be rejuvenated and there is no probability for healing glaucoma, the development of the disease can be controlled.

A diagnosis of glaucoma needs a clinical triad: elevated intraocular pressure, structural adjustment of the optic disc, and visual field problems. As a psychophysical test of optic nerve function, visual field testing plays a vital role in the evaluation of glaucoma. For the past few decades, white-on-white automated perimetry (W-W) has been taken up for the test of reference for glaucoma diagnosis and monitoring. But, as demonstrable visual field problems occur after structural alterations in the optic disc it is now regarded subordinate to optic nerve head description. In optic nerve fiber size damage due to glaucoma was determined. In the superior and inferior of glaucomatous eye, there was a greater atrophy of fibers of all sizes. The areas that suffer loss of fibers in glaucoma contain a high proportion of larger diameter fibers. Larger fibers were lost in areas of the optic nerve with mild damage, indicating the inherent susceptibility to injury by glaucoma.

Patients with ocular hypertension who had blue and blue-yellow colour vision problems had much higher probability occurrence of glaucomatous visual field loss in later years, compared with the patients with normal colour vision results [8]. By means of special approaches that selectively investigate the sensitivity of short-wavelength-sensitive cones, it is easy to detect glaucomatous visual field problems at an earlier stage. But, still there is a great challenge in knowing and realizing the cause, types and the natural course of glaucoma. But, the analysis of CSLT images is a manual procedure in which a trained person has to manually define the margins of the optic nerve and then categorize whether the optic nerve is normal or glaucomatous.

The existing techniques results in misjudgements /errors in the analysis of the CSLT image, failure to distinguish between actual and noisy images and variance in the diagnostic recommendations over a group of practitioners. Therefore the main challenge is to automate the analysis of diagnosis of glaucoma in a quantifiable manner. This paper investigates the existing techniques for the diagnosis of glaucoma. This paper focuses on bringing out the analysis of the characteristic features of the existing approaches, its advantages and disadvantages.

4.Title: Diagnosis of Diabetic Retinopathy: Automatic Extraction of Optic Disc and Exudates from Retinal Images using Marker-controlled Watershed Transformation.

Author: **Ahmed Wasif Reza& C. Eswaran& Kaharudin Dimyati**

Eye diseases, such as diabetic retinopathy and glaucoma affect the retina and cause blindness. This condition tends to occur in patients who have had diabetes for five or more years. It is reported that more than half of all newly registered blindness is caused by the retinal diseases and diabetic retinopathy is one of the main contributors. Automatic screening for eye disease has been shown to be very effective in preventing loss of sight. Manual analysis and diagnosis requires a great deal of time and energy to review retinal images which are obtained by fundus camera. Therefore, an automated analysis and diagnosis system will save cost and time significantly considering the large number of retinal images that need to be manually reviewed by the medical professionals each year. In summary, computer-aided analysis can be helpful to assist the screening procedure in detecting diabetic retinopathy and to aid the ophthalmologists when they evaluate or diagnose the fundus images. With this motivation in mind, this paper presents an automatic tracing technique for the boundary detection of bright objects, such as optic disc (OD), exudates, and cotton wool spots in fundus images. The extraction of these objects forms an important step in the diagnosis of eye diseases, such as diabetic retinopathy and glaucoma.

CHAPTER 4

SYSTEM ANALYSIS

4.1 EXISTING SYSTEM

- In the existing method the glaucoma detection is performed based on the ellipse fitting and the classification of the dataset is performed.
- The performance of the Glaucoma detection is based on the diameter in ellipse fitting.
- And the disk to cup ratio is calculated by finding the diameter of disk and cup in ellipse fitting.
- Then the featured image is used as test image in the classification and then the diameter of the fuzzy logic output is determined.
- Then the Disk to Cup Ratio is calculated.
- And based on the Disk to Cup Ratio the classification is performed.

Limitations of Existing System:

- The process is not so accurate.
- The classification of data has as error.
- The process is more sensitive and even the cup to disk ratio is not measured.

4.2 PROBLEM STATEMENT

So many people are searching to find the abnormal parts and their glaucoma stages in retina but the exact output we are not getting and the accuracy also. For this problem, we are trying to get the classification results as stages with the help of cup to disk ratio exactly and trying to get the accuracy more.

4.3 PROPOSED SYSTEM

a. In pattern recognition term, detection of blood vessels a segmentation tasks where the object is to separate the structure of interest from the background.
b. To justify the motto of the project and to improve the accuracy level the optical Disk and cup segmentation is performed.
c. And then the image smoothing is performed to trace the outer line of the features.
d. Then to perform the classification the Disk to cup ratio is calculated.
e. Then the classification of dataset into two categories such as Normal eye and Glaucoma eye is performed based on the Disk to cup ratio.
f. The classification will determine normality or abnormality of the disc space in retinal blood cells.
g. In normality classification, the accuracy of vessel segmenting will define normal values.
h. In abnormality classification, the accuracy rate of vessel segmenting will show damage part having blood vessel contours as abnormal values.
i. Finally, after performing all these tasks we are going to detect the glaucoma stages from retinal fundus images and displayed the cup to disk ratio based on that we are going to say the glaucoma stage.

Advantages of Proposed System

- The process is more accurate.
- The error rate is reduced.
- The process is less complex.
- The is less sensitive and even cup to disk ratio is calculated accurately and displayed on the command window.

4.4 SYSTEM ARCHITECTURE

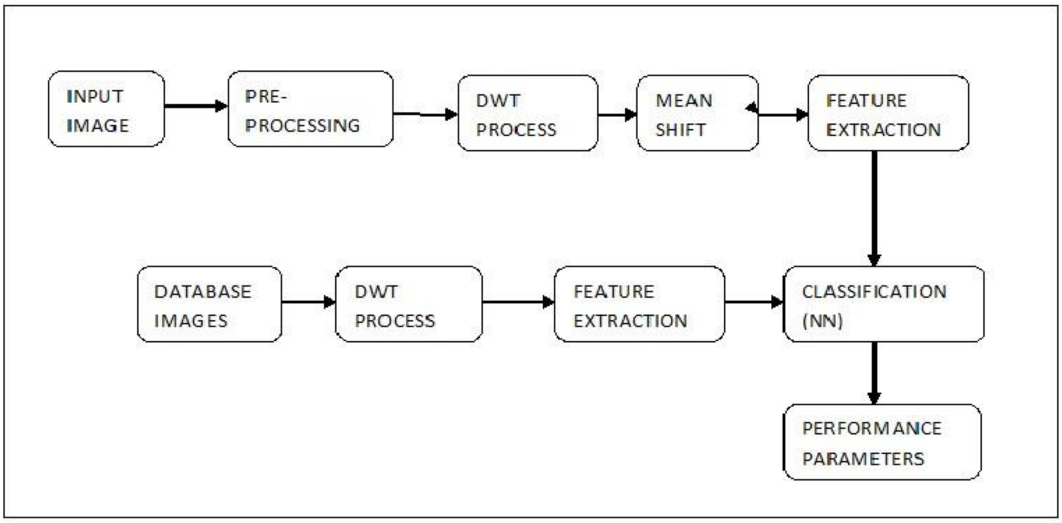

Figure 4.1: System Architecture

4.4.1 Blocks Description

INPUT IMAGE:

We are creating one interface can select any image as a test image.

PRE-PROCESSING:

It is used to improve the quality of image, reducing the complexity in calculation.

DWT PROCESS:

It is used to detect retinal blood vessels, collecting the edge information. The image can be segmented thoroughly and finally obtained the image into segments.

MEAN SHIFT ALGORITHM:

This is a algorithm used for detecting the Exudates.

FEATURE EXTRACTION:

It's a major process in recognition applications and classifications, calculating features and store in one variable.

DATA BASE IMAGE:

All the image features are collecting information are stored in the data sets.

CLASSIFICATION (NN TRAINING):

Normally the classification is used to classify that the image is normal or abnormal. NN is one type of classifier. The classifier compares the given image with in the data base if the abnormality identified while comparing each pixel, it displays the message box after completing the NN training.

4.5 MODULES

- Optical Disk and Cup Segmentation.
- Fuzzy Logic Edge Detection.
- Disk to Cup Ratio.
- Classification.

4.5.1 Flow Diagram

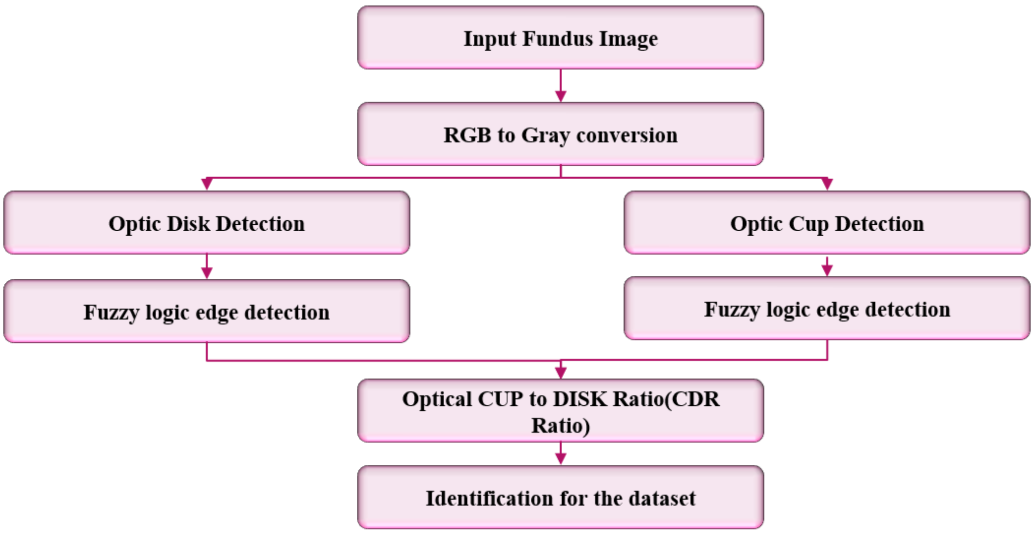

Figure 4.2: Flow Diagram

4.5.2 MODULE DESCRIPTION

OPTICAL DISK SEGMENTATION:

The human visual system does not perceive the world in the same manner as digital detector; hence we need to convert the image into digital image, its nothing but the image processing. Here we are enhancing the contrast of an image for further process. We are also converting the image into grayscale image. From the fundus images, the optic disc is segmented. For the segmentation of optic disc, we compare the fundus image with the threshold value of '0' while a threshold value of our object pixel is '1'. After the comparison we get the optic disc boundary from fundus image. In this paper, we are localizing the optic disc boundary by comparison method. By using edge detection, we localize the optic disc.

OPTICAL CUP SEGMENTATION:

After completing the localization of optic disc boundary, optic cup localization is also mandatory. To localize the optic cup boundary we use the segmentation method same as used for optic disc segmentation. The extraction of the cup boundary we compare the again optic disc with the blank black color image of the same size of the image. From comparing this we get the optic cup boundary.

FUZZY LOGIC EDGE DETECTION:

The processing of captured images involves multiple procedures, so it is necessary to extract important features that will be used to develop an application. In the particular case of a humanoid robot, the system is constantly obtaining images that can be used to carry a set of important tasks, namely a) determine the position of the ball on the field, b) define play spaces and c) locate itself and its opponents into the field. These tasks can be all performed by processing the captured images and are a key aspect to create strategies according to the current state of each element. One of processes that can be applied to image is edge extraction. It represents an important step because it allows an initial distribution into the field. The image preprocessing is to convert the color image to grayscale, and considering that the edges are pixels that have a significant variation in gray level; it is possible to determine a chain of pixels representing the edges. For a reference pixel I_{xy} is the difference with neighboring elements by establishing a matrix

DISK TO CUP RATIO:

The cup-to-disc ratio (often notated CDR) is a measurement used in ophthalmology and optometry to assess the progression of glaucoma. The optic disc is the anatomical location of the eye's "blind spot", the area where the optic nerve and blood vessels enter the retina. The optic disc can be flat or it can have a certain amount of normal cupping. But glaucoma, which is in most cases associated with an increase in intraocular pressure, often produces additional pathological cupping of the optic disc. The pink rim of disc contains nerve fibers. The white cup is a pit with no nerve fibers. As glaucoma advances, the cup enlarges until it occupies most of the disc area. [1] The cup-to-disc ratio compares the diameter of the "cup" portion of the optic disc with the total diameter of the optic disc. A good analogy to better understand the cup-to-disc ratio is the ratio of a donut hole to a donut. The hole represents the cup and the surrounding area the disc. If the cup fills 1/10 of the disc, the ratio will be 0.1. If it fills 7/10 of the disc, the ratio is 0.7. The normal cup-to-disc ratio is 0.3. A large cup-to-disc ratio may imply glaucoma or other pathology. [2] However, cupping by itself is not indicative of glaucoma. Rather, it is an increase in cupping as the patient ages that is an indicator for glaucoma. Deep but stable cupping can occur due to hereditary factors without glaucoma. The cup-to-disc ratio (often notated CDR) is a measurement used in ophthalmology and optometry to assess the progression of glaucoma. The optic disc is the anatomical location of the eye's "blind spot", the area where the optic nerve and blood vessels enter the retina.

4.6 SOFTWARE AND HARDWARE REQUIREMENT

The main tools required for this project can be classified into two categories

1. Hardware requirement

2. Software requirement

HARDWARE REQUIREMENT

Processer : Intel

Mother board : Intel

RAM : 512

Hard disk : 80GB

4.7 SOFTWARE REQUIREMENT

MATLAB

MATLAB is a high-performance language for technical computing. It integrates computation visualization and programming in an easy to use environment. MATLAB stands for matrix laboratory. It was written originally to provide easy access to matrix software developed by LINPACK (linear system package) and EISPACK (Eigen system package) projects. MATLAB is therefore built on a foundation of sophisticated matrix in which the basic element in matrix that does not require pre-dimensioning which to solve many technical computing problems especially those with matrix and vector formulations, in a fraction of time.

MATLAB features of applications specific solutions called toolbox. Very important to most users of MATLAB, toolboxes allow learning and applying specialized technology. These are comprehensive collections of MATLAB functions that extend the MATLAB environment to solve particular classes of problems. Areas in which toolboxes are available include signal processing, control system, neural networks, fuzzy logic, wavelets, simulation and many others.

TYPICAL USES OF MATLAB

The typical using areas of MATLAB are

1. Math and computation
2. Algorithm and development
3. Data acquisition
4. Data analysis, exploration and visualization
5. Scientific and engineering graphics
6. Modelling
7. Simulation
8. Prototyping
9. Application development and including graphical user interface building.

MATLAB is an interactive system whose basic data element is an array that does not require dimensioning. This allow you to solve many technical computing problems, especially those with matrix and vector formulations, in a fraction of the time it would take to write a program in a scalar non-interactive language such as C or FORTRON.

MATLAB features a family of add-on application-specific solutions called toolbox. Very important to most users of MATLAB, toolbox allows you to learn and apply specialized technology. Toolbox is comprehensive collections of MATLAB functions that extend the MATLAB environment to solve particular classes of problems.

Areas in which toolboxes are available include signal processing, control systems, neural networks, fuzzy logic, wavelets, simulation and many others.

FEATURES OF MATLAB

1. Advance algorithm for high performance numerical computation, especially in the field matrix algebra.

2. A large collection of predefined mathematical functions and the ability to define one's own functions.

3. Two- and three-dimensional graphics for plotting and displaying data.

4. Powerful, matrix or vector oriented high-level programming language for individual applications.

5. Toolboxes available for solving advanced problems in several application areas.

BASIC BUILDING BLOCKS OF MATLAB

The basic building block of MATLAB is matrix. The fundamental data type is the array. Vectors, scalars, real matrices and complex matrix are handled as specific class of this basic data type. The built-in functions are optimized for vectors operations. No dimension statements are requiring Med for vectors of arrays.

4.8 MATLAB WINDOW

The MATLAB works based on five windows

1. Command window

2. Work space window

3. Current directory window

4. Command history window
5. Editor window
6. Graphics window
7. Online-help window

Command window

The command window is where the user types MATLAB commands and expressions at the prompt (>>) and where the output of those commands is displayed. It is opened when the application program is launched. All commands including user written programs are typed in this window at MATLAB prompt for execution.

Work space window

MATLAB defines the workspace as the set of variables that the user creates in a work session. The workspace browser shows these variables and some information about them. Double clicking on a variable in the work space browser launches the array editor, which can be used to obtain information.

Current Directory Window

The current directory tab shows the contents of the current directory, whose path is shown in the current directory window. For example, in the windows operating system the path might be as follows: c\MATLAB\work, indicating that directory “work” is a sub directory of the main directory “MATLAB”, which is installed in drive c. Clicking on the arrow in the current directory window shows a list of recently used paths. MATLAB uses a search path to find M-files and other MATLAB related files. Any file run in MATLAB must reside in the current directory that is on search path.

Command history window

The command history window contains a record of the commands a user has entered in the command window, including both current and previous MATLAB sessions. Previously entered MATLAB commands can be selected and re-executed from the command history window by right clicking on a command. This is useful to select various options in addition to executing the commands and is useful feature when experimenting with various commands in work sessions.

Editor window

The MATLAB editor is both a text editor specialized for creating Mfiles and a graphical MATLAB debugger. The editor can appear in a window by itself, or it can be a sub window in the desktop. In this window one can write, edit, create and save programs in files called Mfiles.

MATLAB editor window has numerous pull-down menus for tasks such as saving, viewing and debugging files. Because it performs some simple checks and also uses colour to differentiate between various elements of code, this text editor is recommended as the tool of choice for writing and editing M-files.

Graphics or figure window

The output of all graphic commands typed in the command window is seen in this window.

Online help window

MATLAB provides online help for it's built in functions and programming language constructs. The principal way to get help online is to use the MATLAB help browser, opened as a separate window either by clicking on the question mark symbol on the desktop toolbar, or by typing help browser at the prompt in the command window. The help browser is a web browser integrated into the MATLAB desktop that displays a hypertext mark-up language document. The help browser consists of two panes, the help navigator plane, used to find information, and the display plane, used to view this information. Self-explanatory tabs other than navigator plane are used to perform a search.

4.9 MATLAB FILES

MATLAB has two types of files for storing information. They are Mfiles and MAT- files

M-Files

These are standard ASCII text file with 'm' extension to the file name and creating own matrices using m-files which are text files containing MATLAB code. MATLAB editor or another editor is used to create a file containing the same statements which are typed at the MATLAB command line and save the file under a name that ends in m. There are two types of m-files.

Script files

M-files with a set of MATLAB commands in it and is executed by typing name of file on the command line. These files work on global variables currently in that environment.

Function files

A function file is also an M-file except that the variables in a function file are all local. This type of files begins with a function definition line.

Mat-files

These are binary files with. mat extension to that files created by MATLAB when the data is saved. The data written in a special format that only MATLAB can read. These are located into MATLAB with load command.

4.10 MATLAB SYSTEM

The MATLAB system consists of five main parts:

Development environment

This is the set of tools and facilities that help you see use MATLAB functions and files. Many of these tools are graphical user interface. In includes the MATLAB desktop and command window, a command history, an editor and debugger, and browser for viewing help, the work space, files and search path.

MATLAB mathematical function

This is a vast collection of computational algorithms ranging from elementary functions like sum, sine, cosine and complex arithmetic to more many functions like matrix inverse, matrix Eigen values, Bessel functions and fast Fourier transforms.

MATLAB language

This is a high-level matrix or array language with control flow statements, functions, data structures, input or output and object-oriented programming features. It allows both programming in the small to rapidly create quick and dirty throw-away programs, and programming in the large to create complete large and complex application programs.

GUI Construction

MATLAB has extensive facilities for displaying vectors and matrices as graphs, as well as annotating and printing these graphs. It includes high-level functions for two-dimensional and three-dimensional data visualization, image processing, and animation and presentation graphics. It also includes low-level functions that allow you to fully customize the appearance of graphics as well as to build complete graphical user interface on your MATLAB applications.

MATLAB application program interface

It is a library that allows you to write C and FORTRAN programs that interact with MATLAB. It includes facilities for calling routines from MATLAB, calling MATLAB as a computational engine and for reading and writing MAT-files.

4.11 MATLAB WORKING ENVIRONMENT

MATLAB desktop

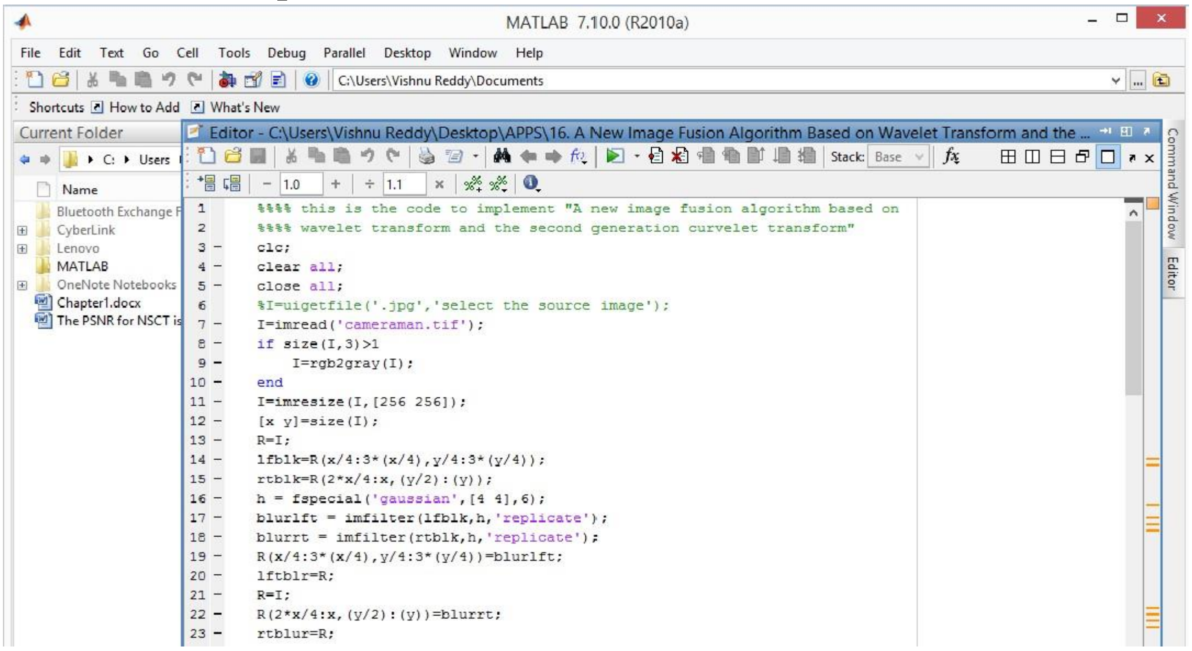

Figure 4.3: Representation of MATLAB window

MATLAB desktop is the main MATLAB application window. The desktop contains five sub windows, the command window, workspace browser, current directory window, command history window, and one or more figure windows which are shown only when the user displays a graphic.

The command window is where the user types MATLAB commands and expressions at the prompt (>>) and where the output of those commands is displayed. MATLAB defines the workspace as the set of variables that the user creates in a work session. The workspace shows these variables and some information about them. Double clicking on a variable in the workspace browser launches the array editor, which can be used to obtain information and income instances edit certain properties of the variable.

The current directory tab above the workspace tab shows the contents of the current directory whose path is shown in the current directory window. For example, in the windows operating system the path might be as follows C:\MATLAB\work, indicating that directory work is a subdirectory of the main directory MATLAB which is installed in drive. Clicking on the arrow in the current directory window shows a list of recently used paths. Clicking on the button to the right of the window allows the user to change the current directory.

MATLAB uses a search path to find M-Files and other related files, which are organize in directories in the computer file system. Any file run in MATLAB must reside in the current directory that is on search path. By default, the files supplied with MATLAB and math works toolboxes are included in the search path. The easiest ways to see which directories is soon the search path or add to modify as search path is to select set path from the file menu the desktop and then use the set path dialog box. It is good practice to add any commonly used directories to the search path to avoid repeatedly having the change the current directory.

The command history window contains a record of the commands a user has entered in the command window including both current and previous MATLAB sessions. Previously entered MATLAB commands can be selected and re-executed from the command history window by right clicking on a command or sequence of commands. This action launches a menu from which to select various options in addition to executing the commands. This is useful to select various options in addition to executing the commands. This is a useful feature when experimenting with various commands in a work session.

Using MATLAB editor to create m-files

The MATLAB editor is both a text editor specialized for creating mfiles and a graphical MATLAB debugger. The editor can appear in window by itself, or it can be a sub window in the desktop. M-files are denoted by the extension.m.

The MATLAB editor window has numerous pull-down menus for tasks such as savings, viewing and debugging files. Because it performs some simple checks and also uses colour to differentiate various elements of code, this text editor is recommended as the tool of choice for writing and editing m-functions.

To open the editor type, edit at the prompt opens the m-file filename.m in an editor window is ready for editing. As noted that the file must be in the current directory or in a directory in the search path.

Getting help

The principal way to get help online is to use the MATLAB help browser, opened as a separate window either by clicking on the questions mark symbol on the desktop toolbar or by typing help browser at the prompt in the command window.

CHAPTER 5

ALGORITHMS AND TECHNIQUES

5.1 INTRODUCTION

Algorithms and techniques provide clear understanding of methods and concepts used in the project. It elaborates the understanding of the proposed system with respect to the existing system. In this chapter, algorithms and techniques used in the project are explained in detail.

5.2 Support Vector Machine Algorithm

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we

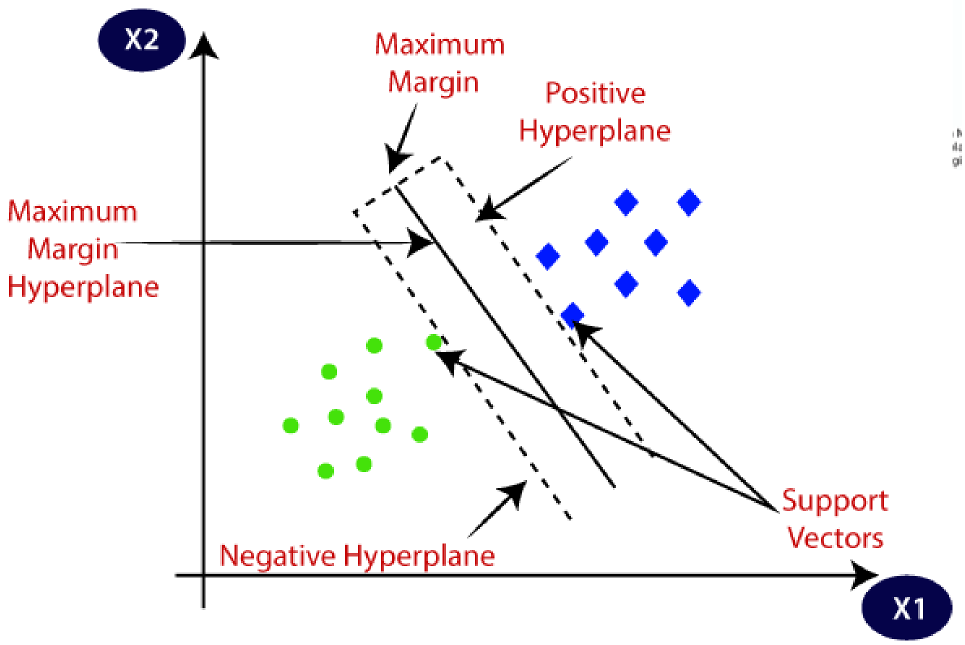

Figure 5.1: By using SVM classifier, two different categories are classified using a decision boundary or hyperplane.

can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.

SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine. Consider the below diagram in which there are two different categories that are classified using a decision boundary or hyperplane.

Example:

SVM can be understood with the example that we have used in the KNN classifier. Suppose we see a strange cat that also has some features of dogs, so if we want a model that can accurately identify whether it is a cat or dog, so such a model can be created by using the SVM algorithm. We will first train our model with lots of images of cats and dogs so that it can learn about different features of cats and dogs, and then we test it with this strange creature. So as support vector creates a decision boundary between these two data (cat and dog) and choose extreme cases (support vectors), it will see the extreme case of cat and dog. On the basis of the support vectors, it will classify it as a cat. Consider the below diagram:

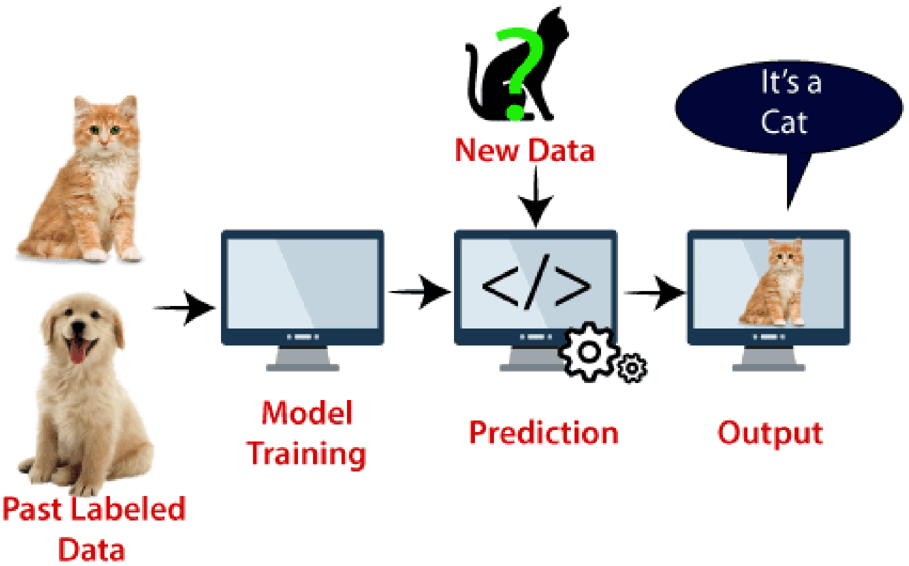

Figure 5.2: Feature extraction using SVM algorithm

SVM algorithm can be used for Face detection, image classification, text categorization, etc.

5.3 Types of SVM

SVM can be of two types:

- **Linear SVM:** Linear SVM is used for linearly separable data, which means if a dataset can be classified into two classes by using a single straight line, then such data is termed as linearly separable data, and classifier is used called as Linear SVM classifier.
- **Non-linear SVM:** Non-Linear SVM is used for non-linearly separated data, which means if a dataset cannot be classified by using a straight line, then such data is termed as non-linear data and classifier used is called as Non-linear SVM classifier.

5.4 Hyperplane and Support Vectors in the SVM algorithm:

Hyperplane: There can be multiple lines/decision boundaries to segregate the classes in n-dimensional space, but we need to find out the best decision boundary that helps to classify the data points. This best boundary is known as the hyperplane of SVM.

The dimensions of the hyperplane depend on the features present in the dataset, which means if there are 2 features (as shown in image), then hyperplane will be a straight line. And if there are 3 features, then hyperplane will be a 2-dimension plane.

We always create a hyperplane that has a maximum margin, which means the maximum distance between the data points.

Support Vectors: The data points or vectors that are the closest to the hyperplane and which affect the position of the hyperplane are termed as support vectors.

How does SVM works?

Linear SVM:

The working of the SVM algorithm can be understood by using an example. Suppose we have a dataset that has two tags (green and blue), and the dataset has two features x1 and x2. We want a classifier that can classify the pair (x1, x2) of coordinates in either green or blue. Consider the below image:

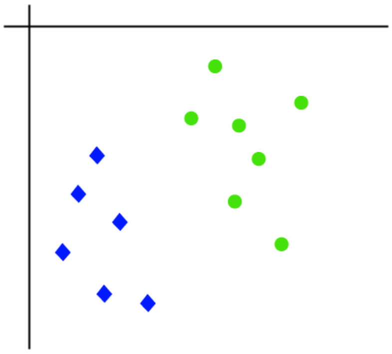

Figure 5.3: Dataset with two different tags.

So, as it is 2-d space so by just using a straight line, we can easily separate these two classes. But there can be multiple lines that can separate these classes. Consider the below image:

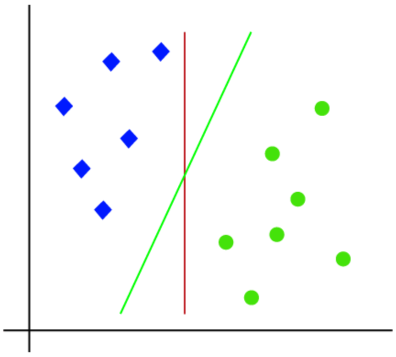

Figure 5.4: 2-D space diagram using one straight line.

Hence, the SVM algorithm helps to find the best line or decision boundary; this best boundary or region is called as a **hyperplane**. SVM algorithm finds the closest point of the lines from both the classes. These points are called support vectors.

And the goal of SVM is to maximize this margin. The **hyperplane** with maximum margin is called the **optimal hyperplane**.

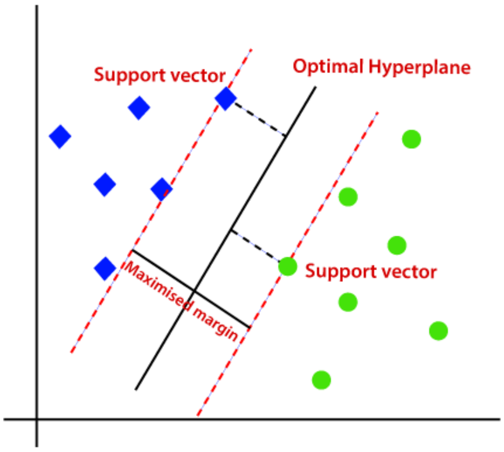

Figure 5.5: 2-D space diagram consists of Optical hyperplane, support vector.

Non-Linear SVM:

If data is linearly arranged, then we can separate it by using a straight line, but for non-linear data, we cannot draw a single straight line. Consider the below image:

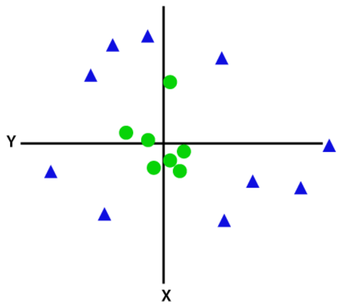

Figure 5.6: Linear data is separated by using a straight line.

So, to separate these data points, we need to add one more dimension. For linear data, we have used two dimensions x and y, so for non-linear data, we will add a third-dimension z. It can be calculated as:

$$Z = x^2 + y^2$$

By adding the third dimension, the sample space will become as below image:

So now, SVM will divide the datasets into classes in the following way. Consider the below image:

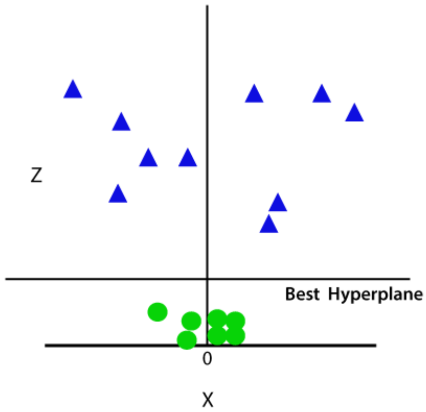

Figure 5.7: Sample space diagram after adding the third dimension.

Since we are in 3-d Space, hence it is looking like a plane parallel to the x-axis. If we convert it in 2d space with $z=1$, then it will become as:

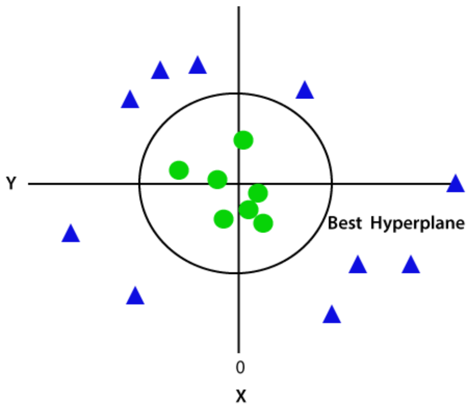

Figure 5.8: 2-D space diagram with non-linear data.

Hence, we get a circumference of radius 1 in case of non-linear data.

CHAPTER 6

FEASIBILITY STUDY

6.1 INTRODUCTION

The feasibility study is carried out to test whether the proposed system is worth being implemented. The proposed system will be selected if it is best enough in meeting the performance requirements.

The feasibility carried out mainly in three sections namely.

• Economic Feasibility

• Technical Feasibility

• Behavioural Feasibility

6.1.1 Economic Feasibility

Economic analysis is the most frequently used method for evaluating effectiveness of the proposed system. More commonly known as cost benefit analysis. This procedure determines the benefits and saving that are expected from the system of the proposed system. The hardware in system department if sufficient for system development.

6.1.2 Technical Feasibility

This study center around the system's department hardware, software and to what extend it can support the proposed system department is having the required hardware and software there is no question of increasing the cost of implementing the proposed system. The criteria, the proposed system is technically feasible and the proposed system can be developed with the existing facility. This technical feasibility helps us in understanding briefly all about the inbuilt procedure.

6.1.3 Behavioural Feasibility

People are inherently resistant to change and need sufficient amount of training, which would result in lot of expenditure for the organization. The proposed system can generate reports with day-to-day information immediately at the user's request, instead of getting a report, which doesn't contain much detail.

6.2 System Implementation

Implementation of software refers to the final installation of the package in its real environment, to the satisfaction of the intended users and the operation of the system. The people are not sure that the software is meant to make their job easier.

- The active user must be aware of the benefits of using the system.
- Their confidence in the software built up.
- Proper guidance is impaired to the user so that he is comfortable in using the application.

Before going ahead and viewing the system, the user must know that for viewing the result, the server program should be running in the server. If the server object is not running on the server, the actual processes will not take place.

6.3 User Training

To achieve the objectives and benefits expected from the proposed system it is essential for the people who will be involved to be confident of their role in the new system. As system becomes more complex, the need for education and training is more and more important. Education is complementary to training. It brings life to formal training by explaining the background to the resources for them. Education involves creating the right atmosphere and motivating user staff. Education information can make training more interesting and more understandable.

6.4 Training on the Application Software

After providing the necessary basic training on the computer awareness, the users will have to be trained on the new application software. This will give the underlying philosophy of the use of the new system such as the screen flow, screen design, type of help on the screen, type of errors while entering the data, the corresponding validation check at each entry and the ways to correct the data entered. This training may be different across different user groups and across different levels of hierarchy.

6.5 Operational Documentation

Once the implementation plan is decided, it is essential that the user of the system is made familiar and comfortable with the environment. A documentation providing the whole operations of the system is being developed. Useful tips and guidance is given inside the application itself to the user. The system is developed user friendly so that the user can work the system from the tips given in the application itself.

6.6 System Maintenance

The maintenance phase of the software cycle is the time in which software performs useful work. After a system is successfully implemented, it should be maintained in a proper manner. System maintenance is an important aspect in the software development life cycle. The need for system maintenance is to make adaptable to the changes in the system environment. There may be social, technical and other environmental changes, which affect a system which is being implemented. Software product enhancements may involve providing new functional capabilities, improving user displays and mode of interaction, upgrading the performance characteristics of the system. So only thru proper system maintenance procedures, the system can be adapted to cope up with these changes. Software maintenance is of course, far more than “finding mistakes”.

6.7 Corrective Maintenance

The first maintenance activity occurs because it is unreasonable to assume that software testing will uncover all latent errors in a large software system. During the use of any large program, errors will occur and be reported to the developer. The process that includes the diagnosis and correction of one or more errors is called Corrective Maintenance.

6.8 Adaptive Maintenance

The second activity that contributes to a definition of maintenance occurs because of the rapid change that is encountered in every aspect of computing. Therefore, Adaptive maintenance termed as an activity that modifies software to properly interfere with a changing environment is both necessary and commonplace.

6.9 Perceptive Maintenance

The third activity that may be applied to a definition of maintenance occurs when a software package is successful. As the software is used, recommendations for new capabilities, modifications to existing functions, and general enhancement are received from users. To satisfy requests in this category, Perceptive maintenance is performed. This activity accounts for the majority of all efforts expended on software maintenance.

6.10 Preventive Maintenance

The fourth maintenance activity occurs when software is changed to improve future maintainability or reliability, or to provide a better basis for future enhancements.

CHAPTER 7

EXPERIMENTAL RESULTS AND EVALUATION

7.1 INTRODUCTION

In this chapter, experimental results and evaluation of analysis are described. The algorithm efficiency and resource utilization are calculated by using MATLAB.

7.2 RESULTS

Home Page

The complete graphical user interface (GUI) of blood vessel segmentation process which has axes and handles to execute the complete program. Below screenshot shows home page of MATLAB which consists of code as follows

7.3 OUTPUT 1

Step 1:

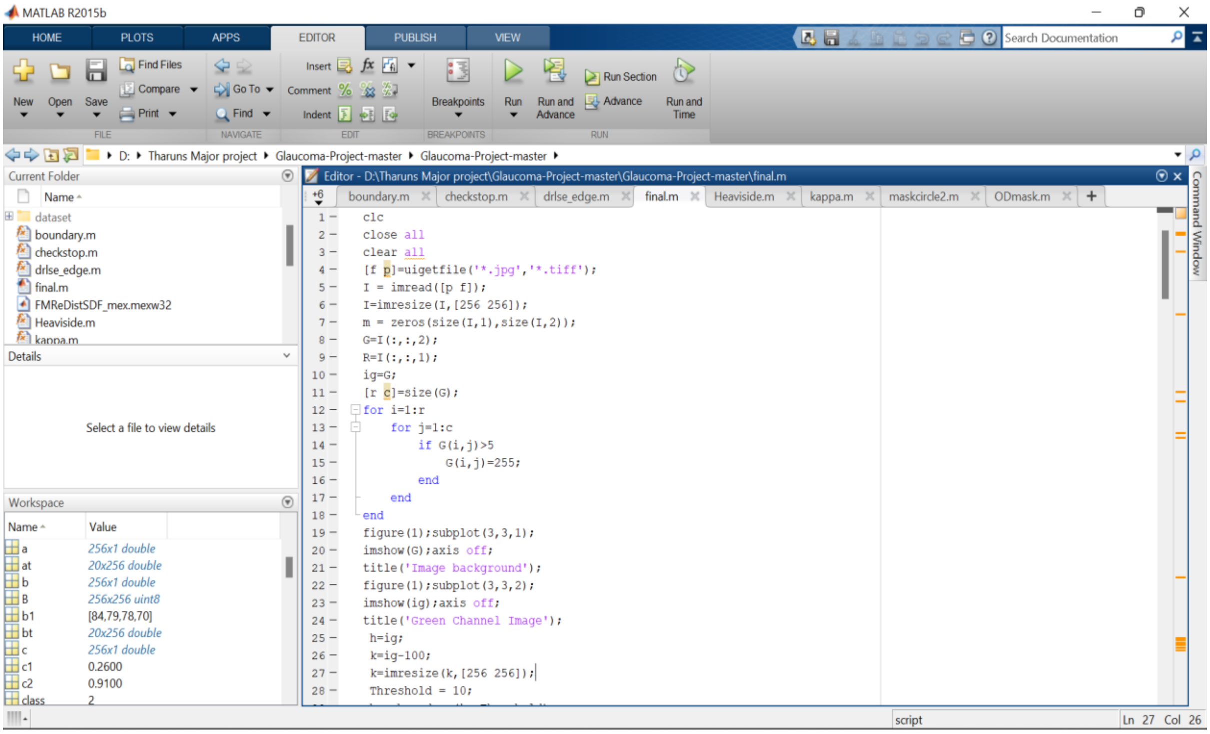

Figure 7.1: MATLAB home page consisting of main code.

Click on **RUN** button and select the image from data set as follows

Step 2:

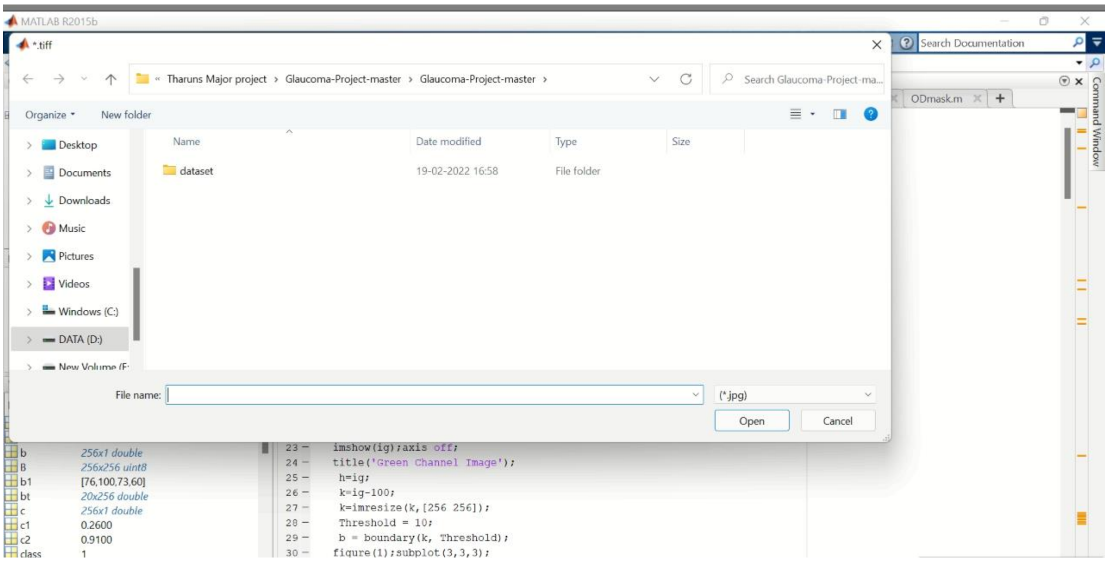

Figure 7.2: opening dialog box for selection of image from dataset.

After opening dialog box select any one image from the dataset and click on insert (or) open button. This process is shown below.,

Step 3:

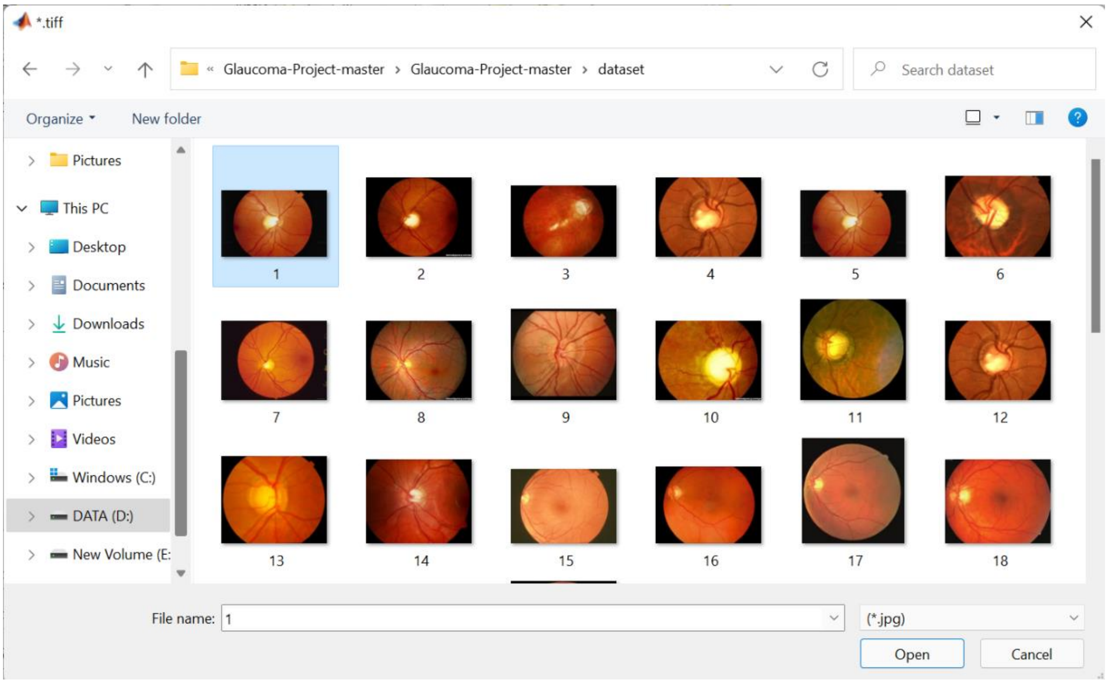

Figure 7.3: Selection of dataset.

After clicking insert (or) open button observe the respected output (Intermediate images) in different stages as shown in below.

Step 4:

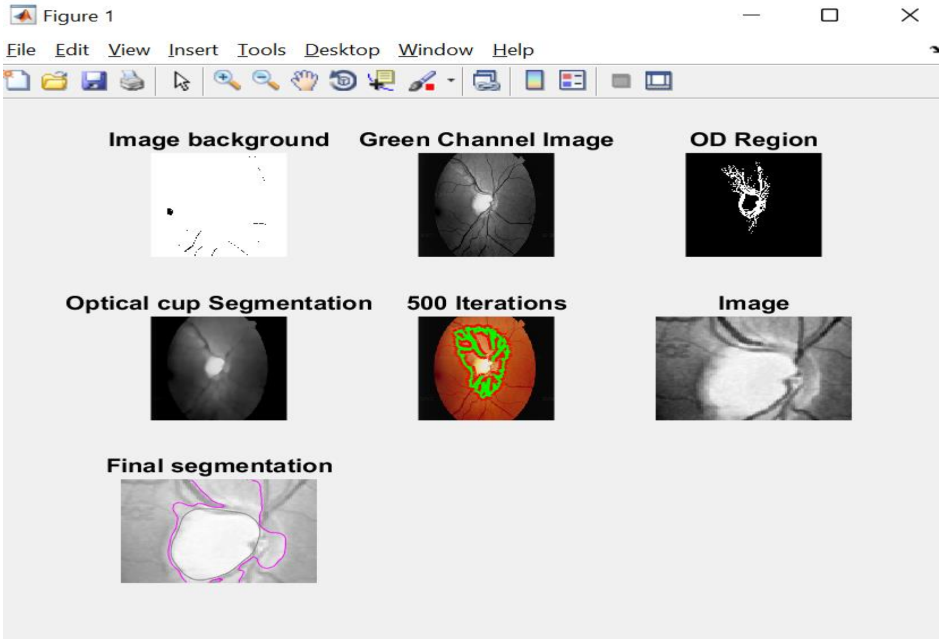

Figure 7.4: Intermediate images of selected dataset.

As we observe above figure 7.4 clearly, we got image background, green channel image, OD (over depletion) region, optical cup segmentation and final segmented image.

Step 5:

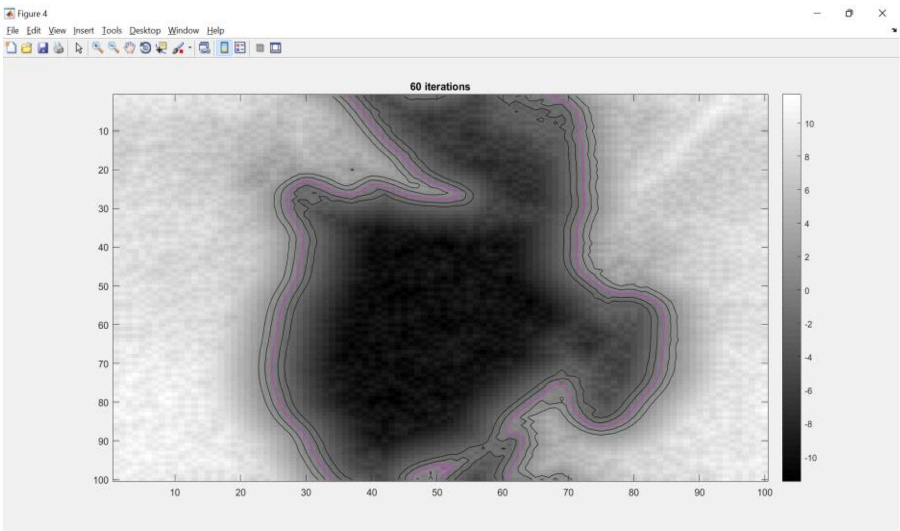

Figure 7.5: epochs

By using ML (Machine learning) Algorithm we train the data set and observe the respected disk to cup ratio output in the command prompt.

Step 6:

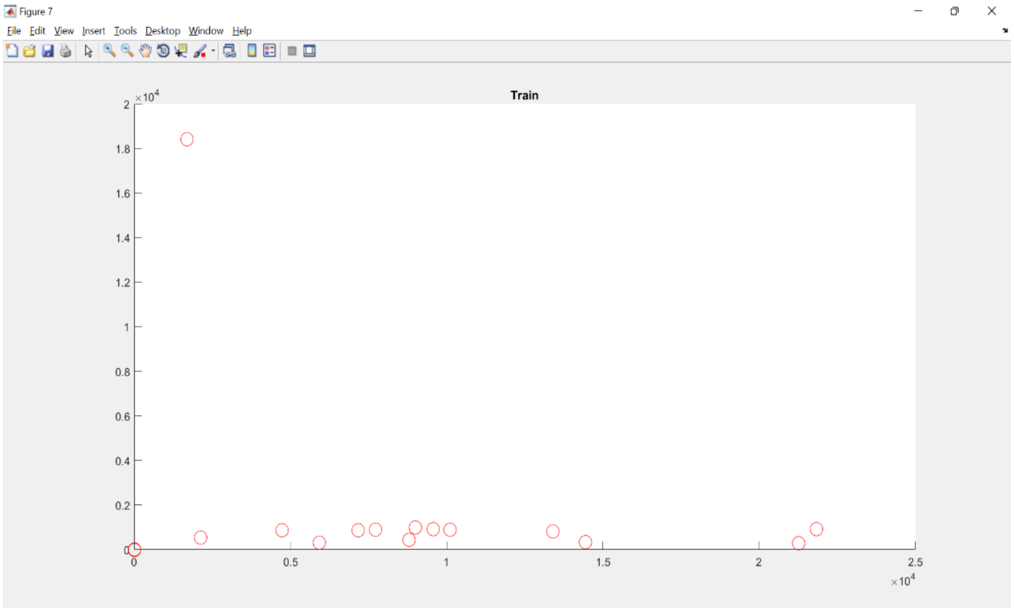

Figure 7.6: training vectors

Step 7:

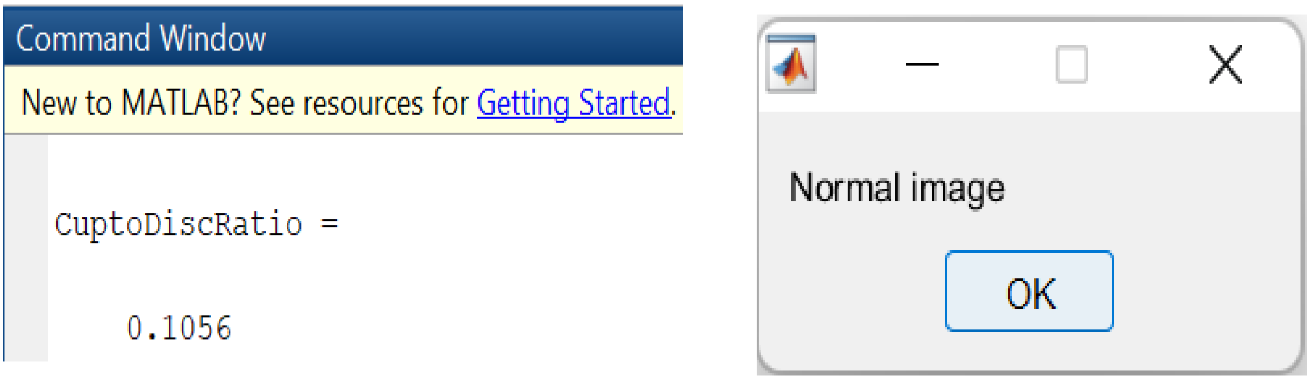

Figure 7.7: Cup to disc ratio and pop up shows that given dataset is a normal image.

In this we have tested one dataset randomly and observed the intermediate process and also got result as normal image which means the given retina image is not affected.

7.4 Output 2

Step 1:

Select one input image from a group of dataset images as shown in below figure., the input retina image is shown below.,

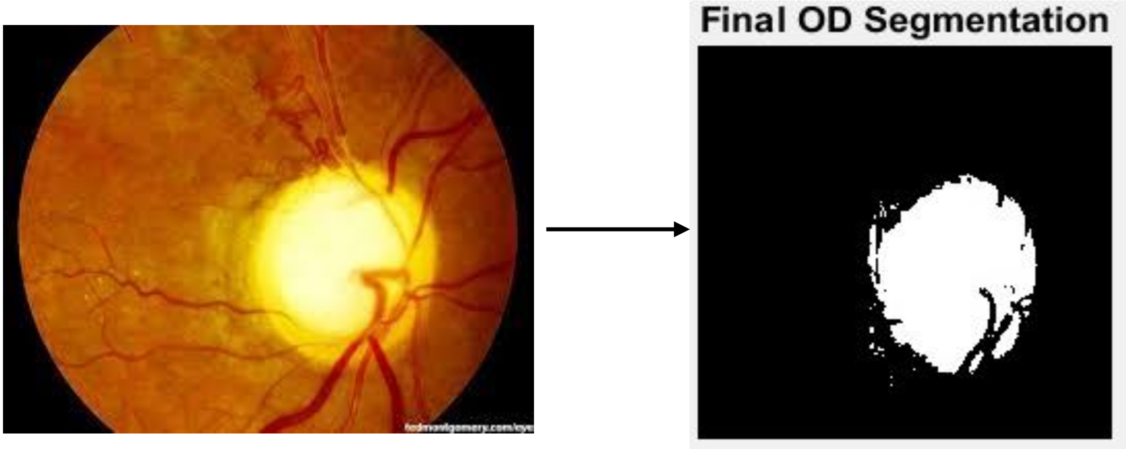

Figure 7.8: Final OD (over depletion) segmented image is taken from input image which is taken from dataset.

Step 2:

Now, observe the Intermediate images of selected dataset.

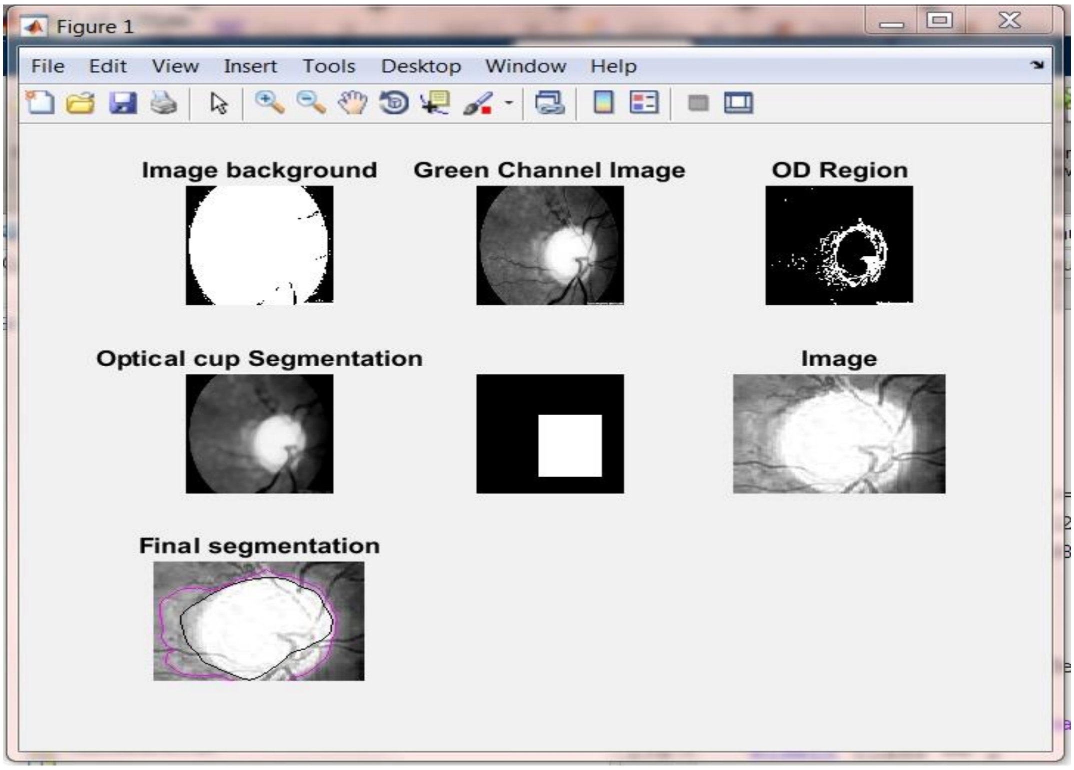

Figure 7.9: Intermediate images of selected dataset.

Step 3:

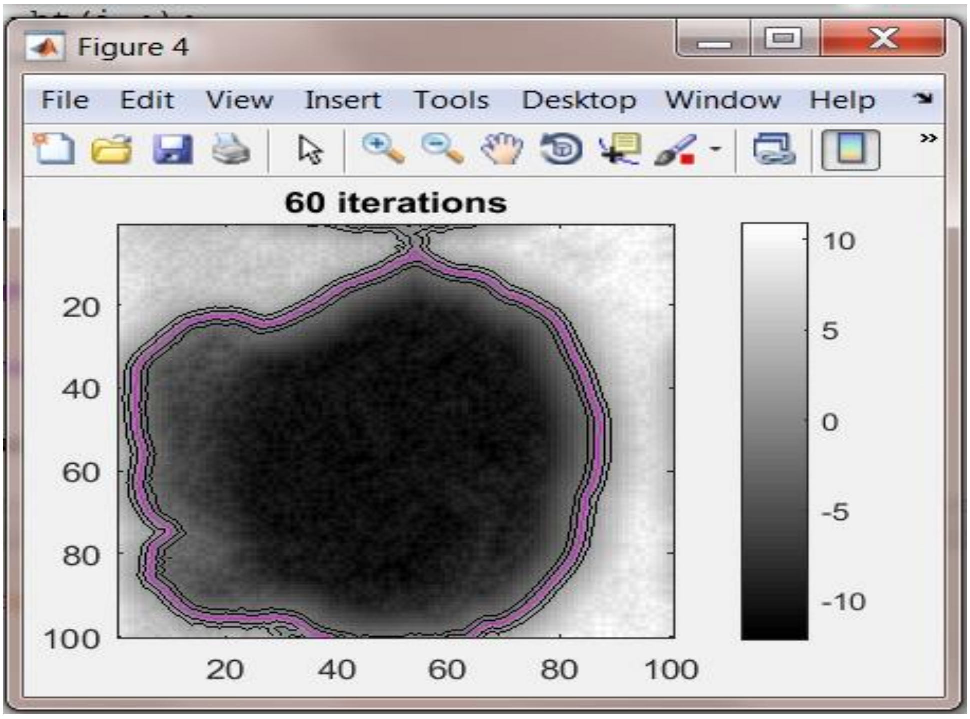

Figure 7.10: epochs

By using ML (Machine learning) Algorithm we train the data set and observe the respected disk to cup ratio output in the command prompt.

Step 4:

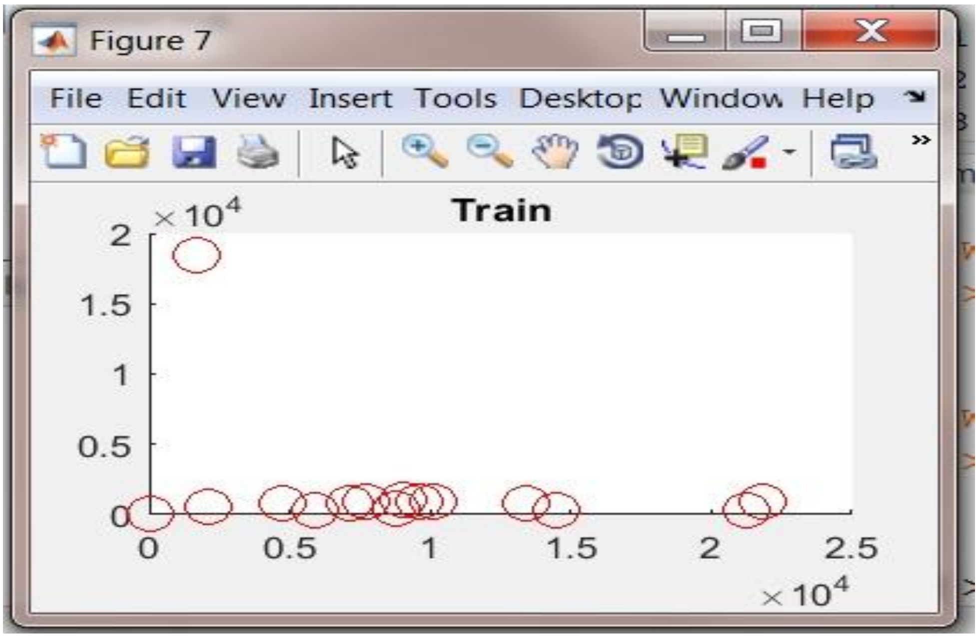

Figure 7.11: training vectors

Step 5:

Figure 7.12: Cup to disc ratio and pop up shows that given dataset is a Glaucoma image.

Here, we have tested another dataset randomly and observed the inter-mediate process and also got result as glaucoma image which means the given retina image is affected because we got cup to disc ratio greater than 0.3 i.e., 0.3458

So only we are concluding that, the tested input image from the dataset is glaucoma image.

CHAPTER 8

CONCLUSION FUTURE SCOPE

8.1 CONCLUSION

In this project, transfer learning is implemented to classify DR (diabetic retinopathy) into 5 classes with a much-reduced training data than other previous DR classification techniques employed. This was done to design a way to train a DL (deep learning) model that performs well on unseen data by efficiently learning from small dataset because training data is limited in healthcare. Our model has reached at an accuracy that is higher than other techniques that challenge dataset for multi-class classification. Our model has reached at a superior performance on account of the selected training algorithm, which is batch gradient descent with ascending learning rate, and the quadratic weighted kappa loss function. Deep learning techniques that can learn from small dataset to categorize medical images should be utilized to classify DR, as this can be transferred to other medical image classification problems facing the challenge of insufficient training data. Experiments should be done to compare performances of other pre-trained deep convolutional Networks

FUTURE WORK

We will use the same model on larger dataset compared to the present dataset. We will apply the feature extraction part from pre-trained model and apply to algorithms such as support vector machines and changing the performance measures such as specificity and sensitivity as it gives healthcare more trust to model usage in real time. we will apply the different image pre-processing techniques to the dataset and compare the performances and will compare the different transfer learning techniques and apply the pre-trained models to complex image classification challenges in real world problems.

REFERENCES

I. J. Liu et al., "Optic cup and disk extraction from retinal fundus images for determination of cup-to-disc ratio," 2008 3rd IEEE Conference on Industrial Electronics and Applications, Singapore, 2008, pp. 1828-1832.

II. F. Mendels, C. Heneghan, and J. P. Thiran, "Identification of the optic disc boundary in retinal images using active contours," Proc. IMVIP, pp. 103–115, 1999.

III. T. Chan and L. Vese, "Active contours without edges," IEEE Trans. Image Processing, vol. 10(2), pp. 266–277,2001.

IV. J. Lowell, A. Hunter, D. Steel, A. Basu, R. Ryder, E. Fletcher, and L.Kennedy, "Optic nerve head segmentation," IEEE Trans. Medical Imaging , vol. 23(2), pp.256–264, 2004.

V. P. Pallawala, W. Hsu, M. Lee, and K. Eong, "Automated optic disc localization and contour detection using ellipse fitting and wavelet transform," Proc. ECCV, pp. 139–151,2004.

VI. H. Li and O. Chutatape, "A model-based approach for automated feature extraction in fundus images," IEEE Trans. Biomedical Engineering, vol. 51(2), pp. 246–254, 2004.

VII. Wong, J. Liu, J. Lim, X. Jia, F. Yin, H. Li, and T. Wong, "Level-set based automatic cup-to-disc ratio de-termination using retinal fundus images in argali," Proc. EMBC, pp. 2266–2269, 2008.

VIII. Gopal Datt Joshi, Jayanthi Sivaswamy, Kundan Karan, S. R. Krishnadas "Optic disc and cup boundary detection using regional information"-IEEE transactions on medical imaging, 2009.

IX. Joshi G D, Sivaswamy J & Krishnadas S.R, 'Optic disk and cup segmentation from monocular color retinal images for glaucoma assessment," IEEE Transactions on Medical Imaging, Vol. 30, No. 6, pp.1192–1205,2011

X. S. Kavita, K. Diraiswamy, "An Efficient Decision Support System For Detection of Glaucoma In Fundus Images Using," International Journal of Advances in Engineering & Technology, Vol. 2, Issue 1, pp. 227-240, Jan 2012.

XI. Chalinee Burana-Anusorn1, Waree Kongprawechnon1, Toshiaki.Kondo1, Sunisa Sintuwong2 and KanokvateTungpimolrut, "Image Processing Techniques for Glaucoma Detection Using the Cup-to-Disc Ratio," Thammasat International Journal of Science and Technology, Vol. 18, No.1, January-March 2013.

XII. Hanmant M. Havagondi, Mahesh S. Kumbhar," Optic cup and disc localization for detection of glaucoma using MATLAB, "International Journal of Electrical, Electronics and Computer Systems (IJEECS), Vol-2, Issue-7, ISSN:.2347-2820, 2014.

XIII. Dr. G.P. Ramesh, M. Malini, "An Efficacious Method of Cup to Disc Ratio Calculation for Glaucoma Diagnosis Using Super pixel," International Journal of Computer Science and Engineering Communications, Vol.2 Issue.3, May 2014.

XIV. Chaitali D. Dhumane, S.B. Patil, "Automated Glaucoma Detection using Cup to Disc Ratio," International Journal of Innovative Research in Science, Engineering and Technology, Vol. 4, Issue 7, July 2016.

XV. Sundari.B, Siva guru, "Early detection of glaucoma from fundus images by using MATLAB GUI for diabetic retinopathy," International Journal of Innovative Research in Computer and Communication Engineering, Vol. 5, Issue 1, January 2018.

XVI. Parashar, Deepak; Agrawal, Dheeraj Kumar. Automated Classification of Glaucoma Stages Using Flexible Analytic Wavelet Transform from Retinal Fundus Images. IEEE Sensors Journal, 1–1. doi:10.1109/JSEN.2020.3001972(March, 2020).