

NEURAL NETWORK ARCHITECTURES FOR ROI SEGMENTATION BASED ON MELANOMA DETECTION

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering Degree in Electronics and Communication Engineering

by

RANISH P (40130194)
SATYA NARAYANAN A (40130204)



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
SCHOOL OF ELECTRICAL AND ELECTRONICS**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

CATEGORY -1 UNIVERSITY BY UGC

Accredited with grade "A++" by NAAC | 12B Status by UGC | Approved by AICTE

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI – 600119

APRIL- 2024



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

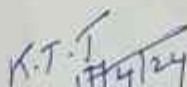
Accredited "A++" Grade by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of SATYA NARAYANAN A (40130204), RANISH P (40130194) who carried out the project entitled " NEURAL NETWORK ARCHITECTURES FOR ROI SEGMENTATION BASED ON MELANOMA DETECTION" under our supervision from November 2023 to April 2024.


Internal Guide

Dr.K.T.ILAYARAJA,M.E.,Ph.D.


Head of the Department
DR.T.RAVI ,M.E.,Ph.D.

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

We, RANISH P (40130194) and SATYA NARAYANAN A (40130204) hereby declare that the Project Report entitled "Neural network architectures for ROI segmentation based on Melanoma detection" done by us under the guidance of Dr.K.T.ILAYARAJAA,M.E.,Ph.D., Assistant Professor, Dept.of Electronics and Communication Engineering, SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY, CHENNAI, is submitted in partial fulfillment of therequirements for the award of Bachelor of Engineering degree in Electronics and Communication Engineering.

DATE:

PLACE: Chennai

SIGNATURE OF CANDIDATES

1.



2.



ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to the **Board of Management of Sathyabama Institute of Science and Technology** for their kind encouragement in doing this project and for completing it successfully. We are grateful to them.

We convey our thanks to **Dr. N. M. NANDHITHA, M.E., Ph.D.** Dean, School of Electrical and Electronics Engineering and **Dr. T. RAVI, M.E., Ph.D.**, Head of the Department, Department of Electronics and Communication Engineering for providing us necessary support and details at the right time during the progressive reviews.

We would like to express my sincere and deep sense of gratitude to our Project Guide **Dr. K.T.ILAYARAJAA, M.E., Ph.D.**, Assistant Professor, Dept. of Electronics and Communication Engineering for his valuable guidance, suggestions and constant encouragement for the successful completion of our project work.

We wish to express my thanks to all Teaching and Non-teaching staff members of the Department of Electronics and Communication Engineering who were helpful in many ways for the completion of the project.

ABSTRACT

Melanoma is the deadliest form of skin cancer that causes around 75% of deaths worldwide. However, most of the skin cancers can be cured, especially if detected and treated early. Existing approaches have employed various feature extraction methods, where different types of features are used individually for skin image classification which may not provide sufficient information to the classification algorithm necessary to discriminate between classes, leading to sub-optimal performance. This study develops a novel skin image classification method using multi-tree genetic programming (GP). To capture local information from gray and color skin images, Local Binary Pattern is used in this work. In addition, for capturing global information, variation in color within the lesion and the skin regions, and domain-specific lesion border shape features are extracted. GP with a multi-tree representation is employed to use multiple types of features. Genetic operators such as crossover and mutation are designed accordingly in order to select a single type of features at terminals in one tree of the GP individual. The performance of the proposed method is assessed using two skin image datasets having images captured from multiple modalities, and compared with six most commonly used classification algorithms as well as the standard (single-tree) wrapper and embedded GP methods. The results show that the proposed method has significantly outperformed all these classification methods. Being interpretable and fast in terms of the computation time, this method can help dermatologist identify prominent skin image.

TABLE OF CONTENTS

CHAPTER No.	TITLE	PAGE No.
	ABSTRACT	V
	LIST OF ABBREVIATIONS	ix
	LIST OF FIGURES	xi
	LIST OF TABLES	xii
1	INTRODUCTION	1
	1.1 Image	3
	1.1.1 Pixels	3
	1.1.2 Image Resolution	4
	1.1.3 Image Brightness	5
	1.1.4 Contrast	6
	1.2 Image Processing	6
	1.2.1 Image Enhancement	9
	1.2.2 Image Restoration	17
	1.2.3 Image Segmentation	18
2	LITERATURE SURVEY	
	2.1 Automated Melanoma Recognition in Dermoscopy Images via Very Deep Residual Networks	20
	2.2 Two system for the detection of Melanoma in Dermoscopy images using Texture and color features	20

2.3 Classification of malignant melanoma and benign skin lesions	21
2.4 Non-invasive realtime automated skin lesion analysis system for melanoma early detection and prevention.	21
2.5 Classification of Malignant melanoma and benign skin lesions	22
2.6 Biologically quad tree color detection using multistage Neural networks	22
2.7 Neural network diagnosis of malignant melanoma from color image	23
3 EXISTING AND PROPOSED SYSTEM	24
3.1 Proposed system	24
3.1.1 Existing system	24
3.1.2 Flow Diagram	25
3.1.3 Feature Extraction	25
3.1.4 Wavelets	26
3.1.5 Genetic Programming	28
3.2 Standards	29
3.3 Constraints	30
3.4 Trade offs	32
4 METHODOLOGY	33
4.1 Flow diagram Description	33
4.2 Block Explanation	33
4.2.1 Grayscale Conversion	34

4.2.2	Median Filter	36
4.2.3	K-Means Clustering	37
4.2.4	Principal Component Analysis	39
4.3	System Requirements for MATLAB	43
4.4	MATLAB	45
5	RESULTS AND DISCUSSION	49
5.1	Results through Image Segmentation	49
5.2	Results through Classification	50
5.3	Final Output	51
6	CONCLUSION AND FUTURE ENHANCEMENT	52
6.1	Conclusion	52
6.2	Future enhancement	52
	REFERENCES	53

LIST OF ABBREVIATIONS

ABBREVIATION	EXPANSION
DNA	- Deoxyribo Nucleic Acid
ABCD	- Asymmetry, Border, Color, Diameter
JPEG	- Joint Photographic Experts Group
PNG	- Portable Network Graphic
IET	- Institution of Engineering and Technology
IEEE	- Institute of Electrical and Electronics Engineering
LBP	- Local Binary Pattern
CNN	- Convolutional Neural Network
SVM	- Super Vector Machine
PCA	- Principal Component Analysis
ROI	- Region Of Interest
DWT	- Discrete Wavelet Transform
GP	- Genetic Programming
MATLAB	- MATrix LABoratory
MDL	- Maximum Description Length
ML	- Maximum Likelihood
PDE	- Partial Differential Equation

LIST OF FIGURES

FIGURE No.	NAME OF THE FIGURE	PAGE No.
1.1	Melanoma detection 1	2
1.2	Melonoma detection 2	2
1.3	a) Gray scale image b)Pixel Representation	4
1.4	Structure of gray scale image	7
1.5	Gray scale image with pixel representation	8
1.6	Different shades of Gray	8
1.7	Color Image	9
1.8	A true color image	9
1.9	Grey value histogram	15
1.10	Lookup Table	15
1.11	Image enhancement (a)original image (b) enhanced image	17
1.12	Image restoration (a) original image (b) partially restored image (c)fully restored image	17
3.1	Flow Diagram	25
3.2	Genetic Programming	30
4.1	a)Normal Image b)converted Grayscale image	35
4.2	Median Filter	39

FIGURE No.	NAME OF FIGURE	PAGE No.
5.1	Result of the melanoma test set segmentation.	49
5.2	20 melanoma image segmentation results.	50
5.3	Results through Image Segmentation	50
5.4	Classification	51
5.5	Final Output	51

LIST OF TABLES

TABLE No.	NAME OF TABLE	PAGE No.
3.1	Algorithm Comparison	25

CHAPTER 1

INTRODUCTION

Skin is the outermost layer of human body and it acts as a protective covering from external agents. Skin is the largest organ in the body. Skin plays a major role in regulating body metabolisms. Skin acts as a primary layer of defensive system against germs and other foreign bodies. Skin has mesodermal cells, melanocytes etc. There are many diseases affecting skin. Among that, most dangerous and life-threatening condition is skin cancer. Skin Cancer is a deadly condition affecting the skin. The number of skin cancer patients is increasing at an alarming rate. Late diagnosis and lack of effective treatment are the main reasons. Cancer is a condition in which there is an abnormal cell growth. Unlike normal cells, cancer cells continue to grow and form new abnormal cells. Like that, skin cancer is the abnormal growth of melanocytic cells in the skin. These melanocytic cells produce the pigment, melanin, which acts as a protective shield from harmful radiations. External agents like Ultraviolet radiations and genetic factors cause damage to the DNA of these cells, resulting in the abnormal production of melanocytic cells. These cells clump together to form cancerous growth. Since this cancer is occurring in the melanocytic cells, skin cancer is also known as Melanoma. Skin cancer initially appears as a flattened lesion on the surface of the skin. It appears as a normal mole. But after a certain stage, its size increases and becomes a malignant tumor. Skin cancer, if not treated properly can lead to death of the patient. When skin cancer reaches deeper into the layers of the skin, then the chance of survival of the patient becomes low. Cancer spreads from one part of the body to another through the blood stream or through lymphatic stream and begins to form new cancerous growth. This is the spreading or evolving property of skin cancer. If it starts, spreading, the chance of curing the patient decreases. Melanoma can be cured in its early stages. Initially melanoma is seen only on the epidermis. At this stage, it is not life-threatening and early detection with immediate surgical excision of the lesion result in complete cure. Early diagnosis is more than 90 % curable and late is less than 50 %. So an early detection is inevitable for the treatment of melanoma. Another challenge in the early skin cancer detection is the resemblance of Malignant melanoma with benign lesions. Benign lesions are simply moles. They are a lump of causes in the creativity. Melanoma look alike in many aspects. So the patients may not give much attention to Melanoma, thinking that the lesion is just a mole. This negligence can

later lead to severe melanoma infliction all over the body. Only an expert dermatologist can distinguish melanoma from nevus. Doctors use ABCD technique to distinguish melanoma from benign lesions. ABCD stands for Asymmetry, Border, Color, Diameter. Melanoma lesions are asymmetrical in shape whereas benign lesions are symmetric. Melanoma is characterized by irregular boundary whereas benign lesions are circular shaped. Melanoma has a unique feature called color variegation but benign is having uniform coloring. The diameter of melanoma is grater than 6mm. This ABCD method alone cannot give any accurate results.

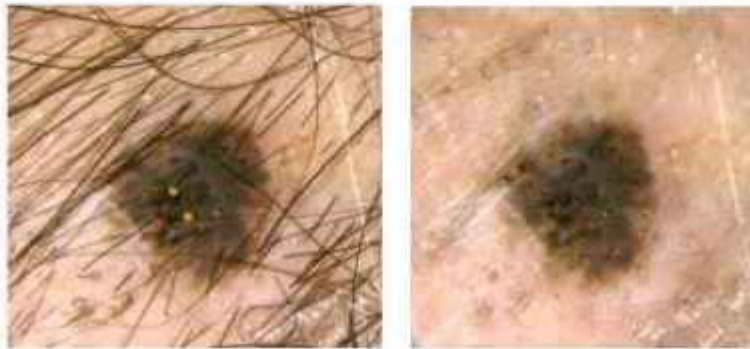


Fig: 1.1 Melanoma 1



Fig: 1.2 Melonoma 2

1.1 IMAGE

An image is an array or a matrix of square pixels (picture elements) arranged in columns and rows. An image (from Latin: imago) is an artifact, for example a two picture, that has a similar appearance to some subject usually a physical object .

1.1.1 Pixels

Image processing is a subset of the electronic domain where in the image is converted to an array of small integers, called *pixels*, representing a physical quantity such as scene radiance, stored in a digital memory and processed by computer or other digital hardware.

Suppose we take an image, a photo, say. For the moment, let's make things easy and suppose the photo is black and white (that is, lots of shades of grey), so no colour. We may consider this image as being a two-dimensional function, where the function values give the brightness of the image at any given point, as shown in figure 1. We may assume that in such an image brightness values can be any real numbers in the range 0.0 (black) to 1.0 (white). The ranges of x and y will clearly depend on the image, but they can take all real values between their minima and maxima.

A digital image differs from a photo in that the x , y , and $f(x, y)$ values are all discrete. Usually they take on only integer values, so the image shown in figure 1.2 will have x and y ranging from 1 to 256 each, and the brightness values also ranging from 0 (black) to 255 (white). A digital image can be considered as a large array of discrete dots, each of which has a brightness associated with it. These dots are called picture elements, or more simply pixels. The pixels surrounding a given pixel constitute its neighbourhood. A neighbourhood can be characterized by its shape in the same way as a matrix: we can speak of a 4×4 neighbourhood, or of a 5×7 neighbourhood. Except in very special circumstances, neighbourhoods have odd numbers of rows and columns; this ensures that the current pixel is in the center of the neighbourhood. An example of a neighbourhood is given in figure 1.4. If a melanoma detection using thus neighbourhood has an even number of rows or columns (or both), it may be necessary to specify which pixel in the neighbourhood

is the "current pixel".

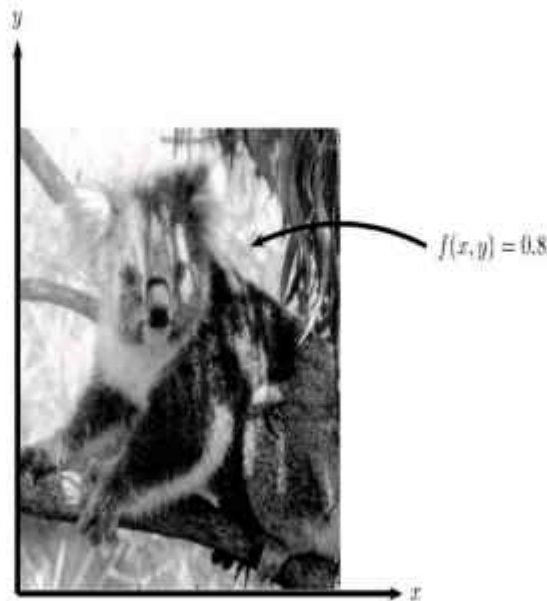


Fig: 1.3 a) Grayscale image

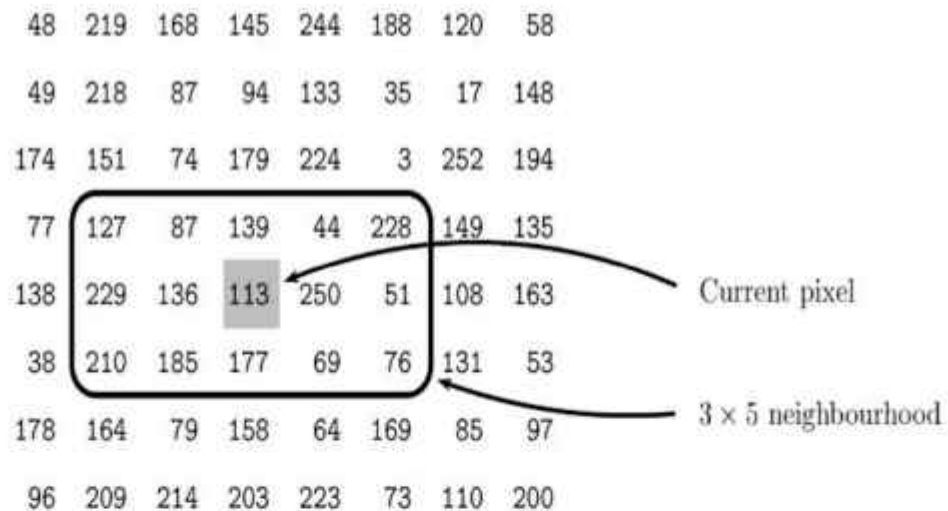


Fig: 1.3 b) Pixel Representation

1.1.2 Image Resolution

Pixels transform into inches through what is called "resolution," -- the number of pixels per square inch on a computer. Resolution allows you to transform pixels into inches and back again.

Two resolution definitions are often used in place of one another. Pixel resolution is the size (in bytes) of your image or its appearance on a computer screen. This number is tied directly to how big your image is on your hard drive. The byte-size of the image file is directly proportional to the pixel count and its size on your computer screen, which simply displays all the pixels in a fixed one-to-one grid. Embedded resolution is different. Embedded resolution tells your printer how far apart to spread the pixels in a printed image. It determines how "fine grained" the printed image will look. It is completely independent of the pixel count (file size) of the image. A high-pixel-count image can have a low embedded resolution or vice versa. Given the same pixel count, a high embedded resolution will result in a smaller printed image (the pixels are packed together more tightly), and a low embedded resolution will result in a larger image (the pixels are more spread out).

1.1.3 Image Brightness

'Brightness' is a problem word, used in most descriptions of image quality. Often, if an image looks bad we will say that it's not 'bright', when it merely lacks contrast. So what does 'brightness' actually mean? Brightness actually describes how we experience light and not 'how it is'. If we are going to describe light and 'brightness' properly, there are two essential terms:

Luminance and illuminance: Luminance is the light we see: reflected or radiating from objects. It is Measured in candela per square metre (cd/m² and the same as 'nit'). Illuminance is actually invisible until it reflects off an object It is Measured in lux (lx or lm/m²). Our eyes are exposed to such dramatic changes in light levels that our optical systems have to protect them. We can read newspaper headlines by the light of a full moon, which might be 0.2 lux of illuminance. Yet we can also read outside in direct , protecting against damage to the retina at the back of the eye. (That reflex to look bright)sunshine, in more than 100,000 lux. So our eyes constantly adjust – our irises contract and expand – to maintain vision consistently, protecting against damage to the retina at the back of the eye. (That reflex to look away from bright lights is important!) Right away we can understand why a screen that looks so bright in a showroom can be barely visible in a shop window!

1.1.4 Contrast

Contrast refers to the difference between black and white levels in images, whether on a flat panel display or a projection screen. Without good contrast, images appear to lack 'brightness', colour and definition. The image contrast ratio refers to the difference between the luminance of the white part of an image, divided by the black part. So if the white part is one hundred times brighter than the black part, it will be 100:1, and so on.

1.1 IMAGE PROCESSING

Image processing is any form of signal processing for which the input is an image, such as a photograph or video frame; the output of image processing may be either an image or a set of characteristics or parameters related to the image. Most image-processing techniques involve treating the image as a dimensional signal and applying standard signal-processing techniques to it.

Image processing usually refers to digital image processing, but optical and analog image processing also are possible. This article is about general techniques that apply to all of them. The acquisition of images (producing the input image in the first place) is referred to as imaging.

Image processing allows one to enhance image features of interest while attenuating detail irrelevant to a given application, and then extract useful information about the scene from the enhanced image. This introduction is a practical guide to the challenges, and the hardware and algorithms used to meet them. An image is digitized to convert it to a form which can be stored in a computer's memory or on some form of storage media such as a hard disk or CD-ROM. This digitization procedure can be done by a scanner, or by a video camera connected to a frame grabber board in a computer. Once the image has been digitized, it can be operated upon by various image processing operations. Image processing operations can be roughly divided into three major categories, Image Compression, Image Enhancement and Restoration, and Measurement Extraction. Image compression is familiar to most people. Image defects which could be caused by the digitization process or by faults in the imaging set-up (for example, bad lighting) can be corrected using Image Enhancement techniques. Once the image

is in good condition, the Measurement Extraction operations can be used to obtain useful information from the image. The example in fig 1.5 operate on 256 gray-scale images. This means that each pixel in the image is stored as a number between 0 to 255, where 0s.

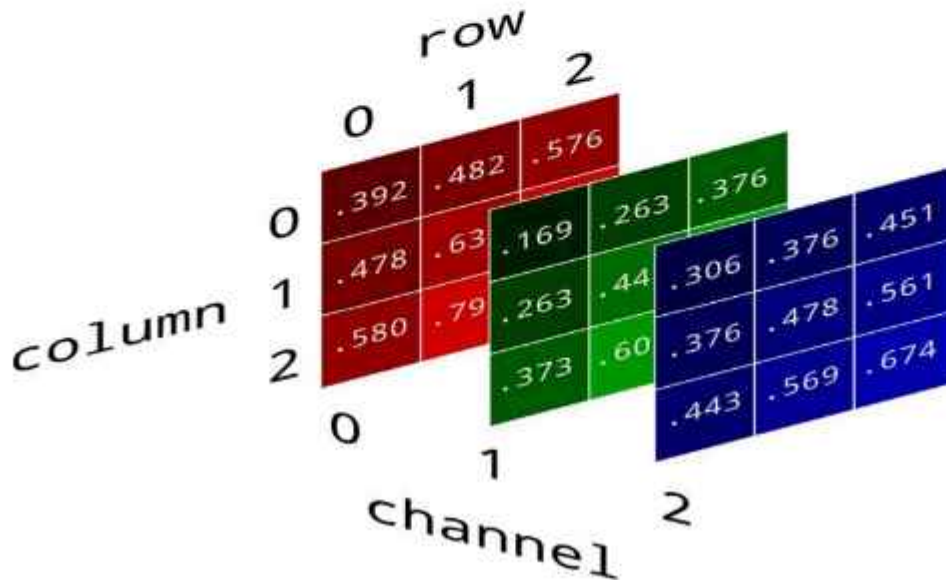


Fig: 1.4 Structure of Gray Scale image

In fig 1.4 each pixel represents a value from 0 to 255 verifying the level of gray. These operations can be extended to colour images too.

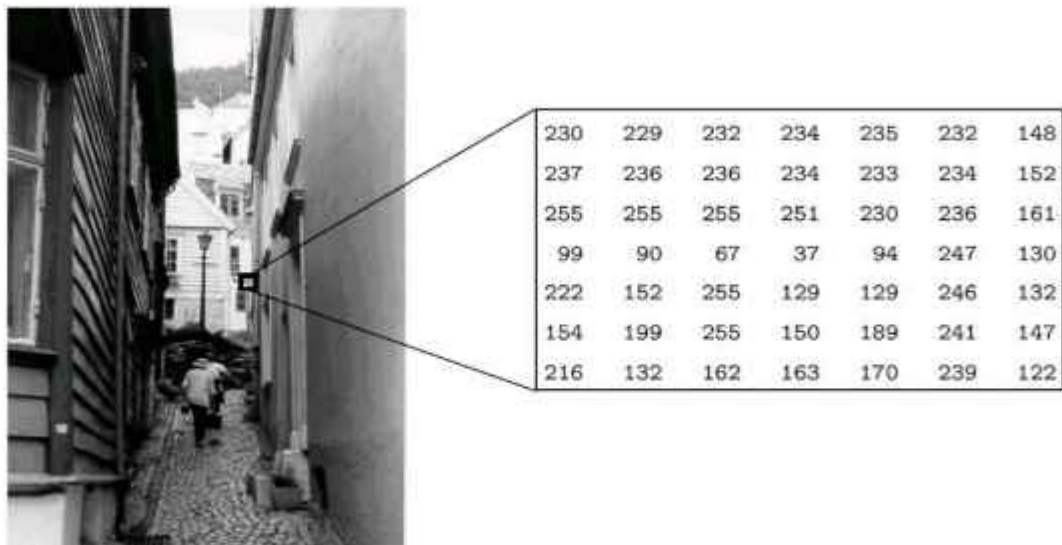


Fig: 1.5 Gray scale image with PIXEL representation

In a (8-bit) greyscale image each picture element has an assigned intensity that ranges from 0 to 255. A grey scale image is what people normally call a black and white image, but the name emphasizes that such an image will also include many shades of grey.

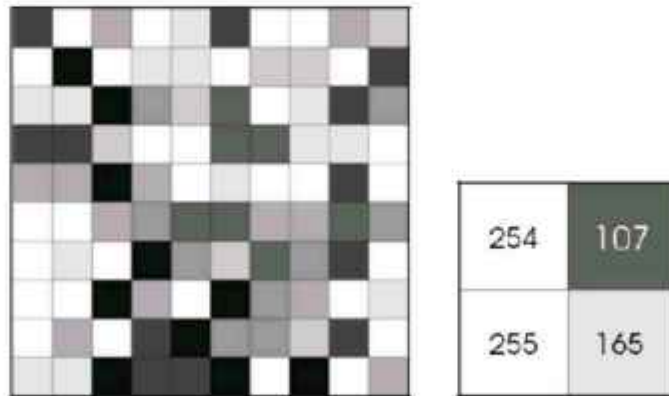


Fig: 1.6 Different shades of grey

A normal greyscale image has 8 bit colour depth = 256 greyscales



Fig: 1.7 Colour image

A "true colour" image has 24 bit colour depth = $8 \times 8 \times 8$ bits = $256 \times 256 \times 256$ colours = ~16 million colours.

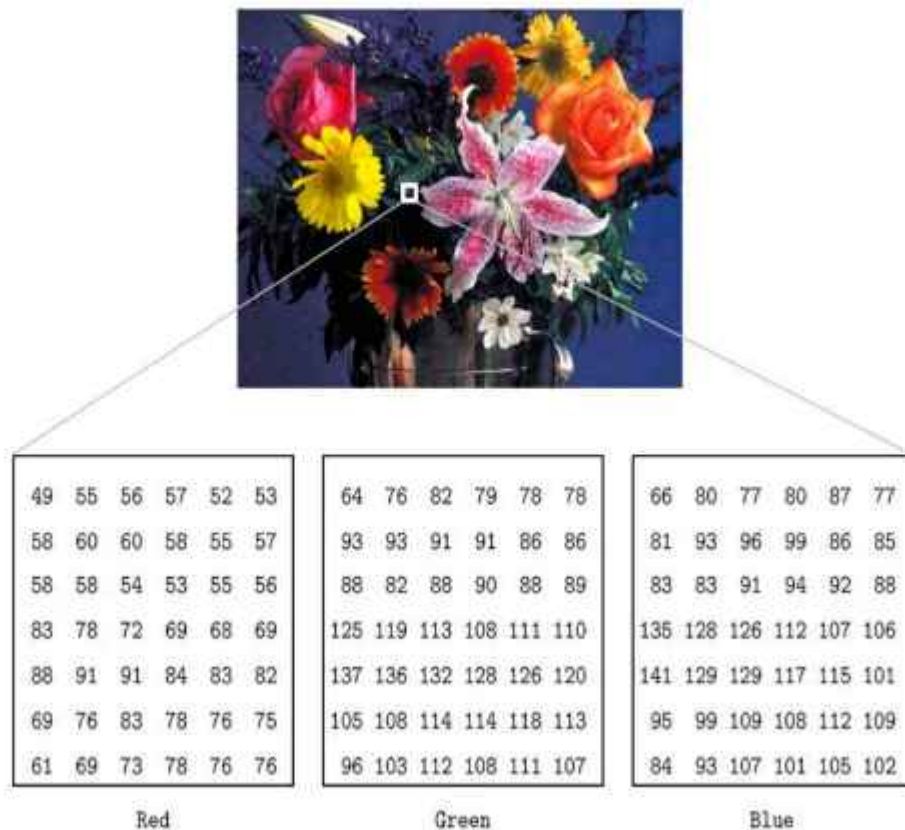


Fig: 1.8 A true colour image

1.2.1 Image Enhancement

Image enhancement is the improvement of digital image quality (wanted e.g. for visual inspection or for machine analysis), without knowledge about the source degradation. If the source of degradation is known, one calls the process image restoration. Both are iconical processes, viz. input and output are images.

Basics of image editing

Raster images are stored in a computer in the form of a grid of picture elements, or pixels. These pixels contain the image's color and brightness information. Image editors can change the pixels to enhance the image in many ways. The pixels can be changed as a group, or individually, by the sophisticated algorithms within the image editors. This article mostly refers to bitmap graphics editors, which are often used to alter photographs and other raster graphics. However, vector graphics software, such as Adobe Illustrator, CorelDRAW, Xara Designer Pro, Pixel Style

Photo Editor, Inkscape or Vector, are used to create and modify vector images, which are stored as descriptions of lines, Bézier curves, and text instead of pixels. It is easier to rasterize a vector image than to vectorize a raster image; how to go about vectorizing a raster image is the focus of much research in the field of computer vision. Vector images can be modified more easily, because they contain descriptions of the shapes for easy rearrangement. They are also scalable, being rasterizable at any resolution.

Automatic image enhancement

Camera or computer image editing programs often offer basic automatic image enhancement features that correct color hue and brightness imbalances as well as other image editing features, such as red eye removal, sharpness adjustments, zoom features and automatic cropping. These are called automatic because generally they happen without user interaction or are offered with one click of a button or mouse button or by selecting an option from a menu. Additionally, some automatic editing features offer a combination of editing actions with little or no user interaction.

Digital data compression

Many image file formats use data compression to reduce file size and save storage space. Digital compression of images may take place in the camera, or can be done in the computer with the image editor. When images are stored in JPEG compression has already taken place. Both cameras and computer programs allow the user to set the level of compression.

Some compression algorithms, such as those used in PNG file format, are lossless, which means no information is lost when the file is saved. By contrast, the JPEG file format uses a lossy compression algorithm by which the greater the compression, the more information is lost, ultimately reducing image quality or detail that can not be restored. JPEG uses knowledge of the way the human brain and eyes perceive color to make this loss of detail less noticeable.

Image editor features

Listed below are some of the most used capabilities of the better graphic

manipulation programs. The list is by no means all inclusive. There are a myriad of choices associated with the application of most of these features.

Selection

One of the prerequisites for many of the applications mentioned below is a method of selecting part(s) of an image, thus applying a change selectively without affecting the entire picture. Most graphics programs have several means of accomplishing this, such as:

- a marquee tool for selecting rectangular or other regular polygon-shaped regions,
- a lasso tool for freehand selection of a region,
- a magic wand tool that selects objects or regions in the image defined by proximity of color or luminance,
- vector-based pen tools, as well as more advanced facilities such as edge detection, masking, alpha compositing, and color and channel-based extraction.

The border of a selected area in an image is often animated with the marching ants effect to help the user .

Layers

Another feature common to many graphics applications is that of Layers, which are analogous to sheets of transparent acetate (each containing separate elements that make up a combined picture), stacked on top of each other, each capable of being individually positioned, altered and blended with the layers below, without affecting any of the elements on the other layers. This is a fundamental workflow which has become the norm for the majority of programs on the market today, and enables maximum flexibility for the user while maintaining non-destructive editing principles and ease of use.

Image size alteration

Image editors can resize images in a process often called image scaling, making them larger, or smaller. High image resolution cameras can produce large images which are often reduced in size for Internet use. Image editor programs use a mathematical process called resampling to calculate new pixel values whose

spacing is larger or smaller than the original pixel values. Images for Internet use are kept small, say 640 x 480 pixels which would equal 0.4 megapixels.

Cropping an image

Digital editors are used to crop images. Cropping creates a new image by selecting a desired rectangular portion from the image being cropped. The unwanted part of the image is discarded. Image cropping does not reduce the resolution of the area cropped. Best results are obtained when the original image has a high resolution. A primary reason for cropping is to improve the image composition in the new image.

Histogram

Image editors have provisions to create an image histogram of the image being edited. The histogram plots the number of pixels in the image (vertical axis) with a particular brightness value (horizontal axis). Algorithms in the digital editor allow user to visually adjust the brightness value of each pixel and to dynamically constant display the results as adjustments are made. Improvements in picture brightness and contrast can thus be obtained

Noise reduction

Image editors may feature a number of algorithms which can add or remove noise in an image. Some JPEG artifacts can be removed; dust and scratches can be removed and an image can be de-speckled. Noise reduction merely estimates the state of the scene without the noise and is not a substitute for obtaining a "cleaner" image. Excessive noise reduction leads to a loss of detail, and its application is hence subject to a trade-off between the undesirability of the noise itself and that of the reduction artifacts. Noise tends to invade images when pictures are taken in low light settings. A new picture can be given an 'antiqued' effect by adding uniform monochrome noise.

Removal of unwanted elements

Most image editors can be used to remove unwanted branches, etc., using a "clone" tool. Removing these distracting elements draws focus to the subject, improving

overall composition.

Image orientation

Image editors are capable of altering an image to be rotated in any direction and to any degree. Mirror images can be created and images can be horizontally flipped or vertically flopped. A small rotation of several degrees is often enough to level the horizon, correct verticality (of a building, for example), or both. Rotated images usually require cropping afterwards, in order to remove the resulting gaps at the image edges.

Perspective control and distortion

Some image editors allow the user to distort (or "transform") the shape of an image. While this might also be useful for special effects, it is the preferred method of correcting the typical perspective distortion which results from photographs being taken at an oblique angle to a rectilinear subject. Care is needed while performing this task, as the image is reprocessed using interpolation of adjacent pixels, which may reduce overall image definition. The effect mimics the use of a perspective control lens, which achieves a similar correction in-camera without loss of definition.

Enhancing images

In computer graphics, the process of improving the quality of a digitally stored image by manipulating the image with software. It is quite easy, for example, to make an image lighter or darker, or to increase or decrease contrast. Advanced photo enhancement software also supports many filters for altering images in various ways. Programs specialized for image enhancement are sometimes called image editors.

Sharpening and softening images

Graphics programs can be used to both sharpen and blur images in a number of ways, such as unsharp masking or deconvolution. Portraits often appear more pleasing when selectively softened (particularly the skin and the background) to better make the subject stand out.[citation needed] This can be achieved with a camera by using a large aperture, or in the image editor by making a selection and

then blurring it. Edge enhancement is an extremely common technique used to make images appear sharper, although purists frown on the result as appearing unnatural.

Another form of image sharpening involves a form of contrast. This is done by finding the average color of the pixels around each pixel in a specified radius, and then contrasting that pixel from that average color. This effect makes the image seem clearer, seemingly adding details. An example of this effect can be seen to the right. It is widely used in the printing and photographic industries for increasing the local contrasts and sharpening the images.

Examples of simple grey level transformations in this domain are

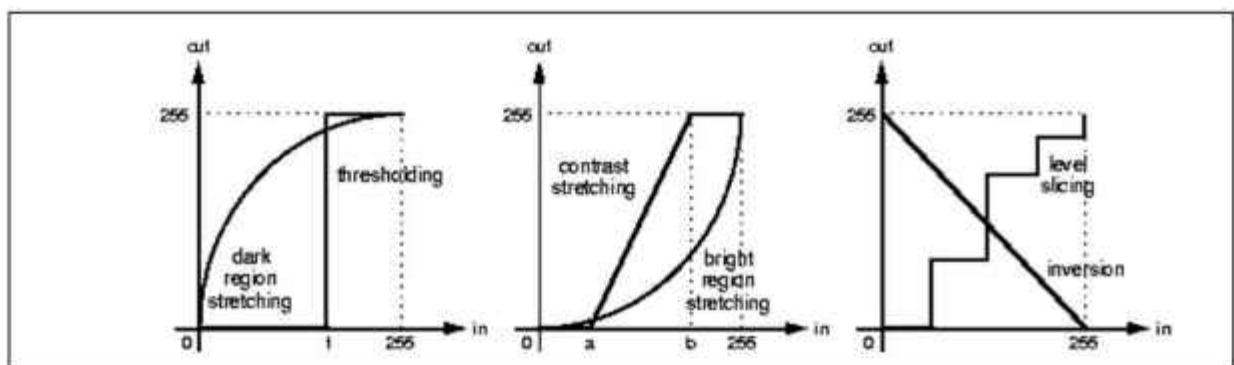


Fig: 1.9 The grey value histogram

Grey values can also be modified such that their histogram has any desired shape, e.g flat (every grey value has the same probability). All examples assume *point processing*, viz. each output pixel is the function of one input pixel; usually, the transformation is implemented with a look-up table:

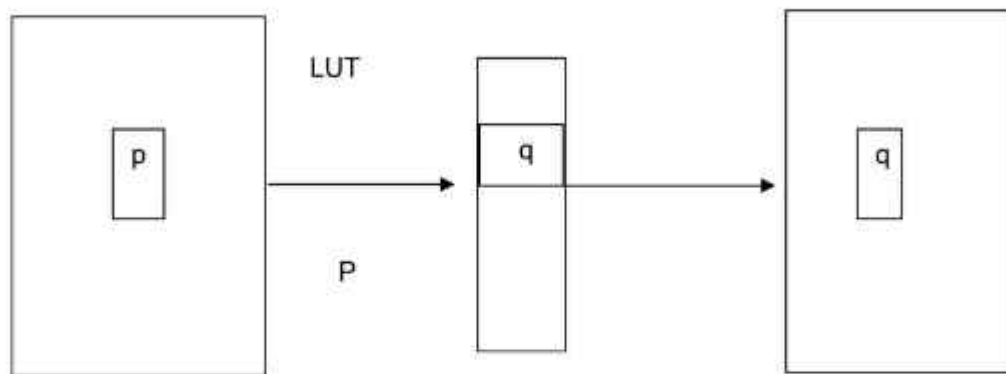


Fig: 1.10 Lookup Table

Physiological experiments have shown that very small changes in luminance are recognized by the human visual system in regions of continuous grey value, and not at all seen in regions of some discontinuities. Therefore, a design goal for image enhancement often is to smooth images in more uniform regions, but to preserve edges. On the other hand, it has also been shown that somehow degraded images with enhancement of certain features, e.g. edges, can simplify image interpretation both for a human observer and for machine recognition.

These operations could be performed using linear operations in either the frequency or the spatial domain. We could, e.g. design, in the frequency domain, one-dimensional low or high pass filters (Filtering), and transform them according to the two-dimensional case.

Unfortunately, linear filter operations do not really satisfy the above two design goals; in this book, we limit ourselves to discussing separately only (and superficially) Smoothing and Sharpening.

Here is a trick that can speed up operations substantially, and serves as an example for both point and neighbourhood processing in a binary image: we number the pixels in a 4x4 neighbourhood like and denote the binary values (0,1) by b_i ($i = 0,8$); we then concatenate the bits into a 9-bit word, like $b_8b_7b_6b_5b_4b_3b_2b_1b_0$. This leaves us with a 9-bit grey value for each pixel, hence a new image (an 8-bit image with b_8 taken from the original binary image will also do). The new image corresponds to the result of a convolution of the binary image, with a 4 x 4 matrix containing as coefficients the powers of two. This neighbour image can then be

passed through a look-up table to perform erosions, dilations, noise cleaning, skeletonization, etc.

Apart from point and neighbourhood processing, there are also global processing techniques, i.e. methods where every pixel depends on all pixels of the whole image. Histogram methods are usually global, but they can also be used in a neighbourhood.



(a)

(b)

Fig: 1.11 Image enhancement (a)original image (b) enhanced image

1.2.2 Image Restoration

The purpose of image restoration is to "compensate for" or "undo" defects which degrade an image. Degradation comes in many forms such as motion blur, noise, and camera misfocus. In cases like motion blur, it is possible to come up with a very good estimate of the actual blurring function and "undo" the blur to restore the original image.

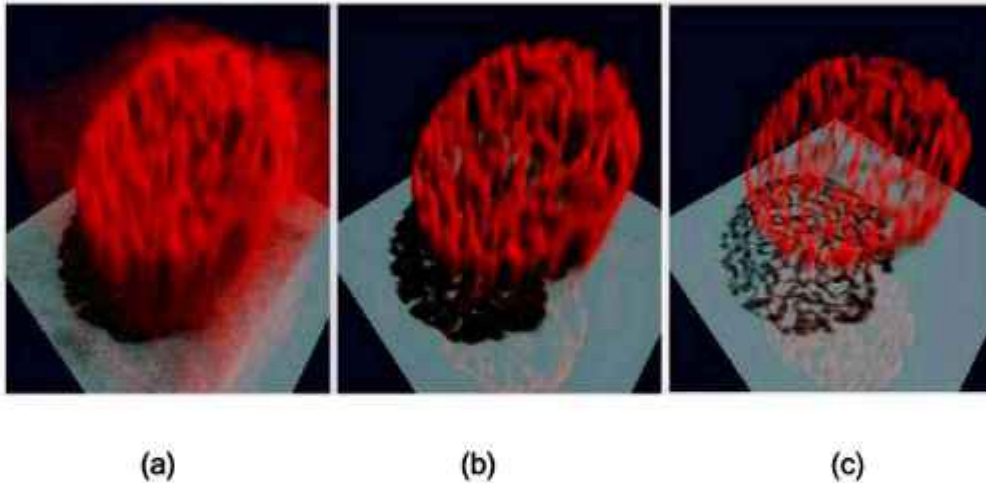


Fig: 1.11 Image restoration (a) original image (b) partially restored image (c) fully restored image

OBJECT RECOGNITION

Object recognition includes the process of determining the object's identity or location in space. The problem of object or target recognition starts with the sensing of data with the help of sensors, such as video cameras and thermal sensors, and then interpreting these data in order to recognize an object or objects. We can divide the object-recognition problem into two categories: the modeling problem and the recognition problem.

1.2.3 Image Segmentation

Image segmentation is the process of partitioning a digital image into multiple segments (sets of pixels, also known as super-pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images. More precisely, image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain characteristics.

The result of image segmentation is a set of segments that collectively cover the entire image, or a set of contours extracted from the image (see edge detection). Each of the pixels in a region are similar with respect to some characteristic or computed property, such as color, intensity, or texture. Adjacent regions are significantly different with respect to the same characteristic(s). When applied to a

stack of images, typical in medical imaging, the resulting contours after image segmentation can be used to create 4D reconstructions with the help of interpolation algorithms like Marching cubes.

Image segmentation is the process of partitioning a digital image into disjointed, meaningful regions. The meaningful regions may represent objects in an image of three-dimensional scene, regions corresponding to industrial, residential, agricultural, or natural terrain in an aerial reconnaissance application, and so on. A region is a connected set of pixels and the objects are considered either four-connected, if only laterally adjacent pixels are considered, or they can be eight-connected, if diagonally adjacent pixels are also considered to be connected. Image segmentation is an efficient and natural process for huma

Segmentation is not an easy task in digital image processing, and it may become a serious problem if the number of objects is large or unknown or if the boundaries between objects are not clear. Three of the most commonly used techniques for digital image segmentation are the gray-level thresholding technique, gradient-based segmentation technique, and the region growing technique. A brief description of the three techniques is given below.

Gray-Level Thresholding

This technique is an example of the regional approach, which implies grouping pixels into distinct regions or objects. When using this technique for image segmentation, all pixels that have gray levels equal to or above a certain threshold are assigned to the object. The rest of the pixels are assigned to the object's background. If the image contains more than one object, then more threshold levels can be used to do the segmentation. The thresholding method involves first locating objects in an image by using the intensity gradient, then noting the levels that correspond to the objects in various areas of the image, and finally using these levels as initial guesses at a threshold. This method is capable of thresholding images that have been produced in the context of variable illumination. The thresholding method, called the local intensity gradient (LIG) method, was implemented in C using a Sun4 host running UNIX. The LIG method was compared against iterative selection (IS), gray level histograms (GLHs) and two correlation based algorithms on a dozen sample images under three different illumination

effects. Overall, the LIG method, while it takes significantly longer, properly thresholds a larger set of images than does any other method examined over the sample images tested.

Many types of automatic thresholding methods exist, the most famous and widely used being Otsu's method. The following list, based on the works of Sezgin et al. (2004) categorizes thresholding methods into broad groups based on the information the algorithm manipulates. Note however that such a categorization is necessarily fuzzy as some methods can fall in several categories (for example, Otsu's method can be both considered a histogram-shape and a clustering algorithm)

- **Histogram shape**-based methods, where, for example, the peaks, valleys and curvatures of the smoothed histogram are analyzed.^[2] Note that these methods, more than others, make certain assumptions about the image intensity probability distribution (i.e., the shape of the histogram),
- **Clustering**-based methods, where the gray-level samples are clustered in two parts as background and foreground,^{[3][4]}
- **Entropy**-based methods result in algorithms that use the entropy of the foreground and background regions, the cross-entropy between the original and binarized image, etc.,^[5]
- **Object Attribute**-based methods search a measure of similarity between the gray-level and the binarized images, such as fuzzy shape similarity, edge coincidence, etc.,
- **Spatial** methods use higher-order probability distribution and/or correlation between pixels.

CHAPTER 2

LITERATURE SURVEY

2.1 Automated Melanoma Recognition in Dermoscopy Images via Very Deep Residual Networks

Journal Name: IEEE

Year of publication: 2016

Pros & Cons

They propose a novel method based on very deep CNNs to meet the challenges of automated melanoma recognition in dermoscopy images, which consists of two steps: segmentation and classification. CNNs can generate features with high discrimination capability, and hence improve the performance of both segmentation and classification tasks.

2.2 Two Systems for the Detection of Melanomas in Dermoscopy Images Using Texture and Color Features

Journal Name : IEEE

Year of Publication : 2014

Methodology:

This paper proposes two systems for the automatic classification of melanocytic skin lesions. The first system uses global methods to classify skin lesions. The second system uses local features, following a recent trend in image analysis and recognition.

Pros & Cons:

This paper has two different strategies for the detection of melanomas in dermoscopy images based on local and global features. The other objective is to compare the role of color and texture features in lesion classification and determine which set of features is more discriminative. It is concluded that color features outperform texture features achieve very good results, i.e., Sensitivity = 96% and Specificity = 80% for global methods against Sensitivity = 100% and Specificity = 75% for local methods.

2.3 Classification Of Malignant melanoma and benign skin lessins.

Journal Name : IET

Year of Publication: 2015

Methodology:

They propose a system in 4 tasks Segmentation. Feature Detection and disease classification.

Pros & Cons:

Analysis of the classification task demonstrates that ensembles of deep learning approaches and additional data led to the highest performance. In addition, collaborative fusions of all participant systems outperformed any single alone. In imaging technology has improved in recent years, encouraging researchers to develop a non-invasive computer-aided diagnosis (CAD) for melanoma detection based on lesion's features. Garnavi *et al.* used border and wavelet based texture methods for computer-aided melanoma diagnosis.

2.4 Noninvasive Real-Time Automated Skin Lesion Analysis System for Melanoma Early Detection and Prevention.

Journal Name: IEEE

Year of Publication: 2015

Methodology:

They propose an accessible mobile health-care solution for melanoma detection, using mobile image analysis.

Pros & Cons:

It could be employed by the general public for preliminary selfscreening or it can assist the general physicians during the diagnosis process. To detect and exclude the hair from the lesion, first, the hair is segmented from the lesion. To accomplish this task, a set of 84 directional filters are used.

2.5 Classification of malignant melanoma and benign skin lesions: implementation of automatic ABCD rule

Journal Name: IET

Year of Publication: 2015

Methodology:

In this paper, image processing techniques were used for an automatic ABCD rule implementation to discriminate malignant melanoma from benign lesions.

Pros & Cons:

The proposed approach may represent a very useful tool for less experienced dermatologists and clinicians. Algorithms are implemented to extract the characteristics of ABCD attributes. Methods used here combine existing methods with novel methods to detect colour asymmetry and dermoscopic structures.

2.6 Biologically Inspired Quad Tree Colour Detection in Dermoscopy Images of Melanoma

Journal Name: IEEE

Year of Publication :2010

Methodology:

In this paper, biologically inspired colour clustering using a QuadTree technique can obtain a high melanoma classification rate in dermoscopy images.

Pros & Cons:

The presence of blue-grey, red, white and black colours have been shown to be a strong indicator of malignancy To explore the influence of colour combinations on classification, we performed an experiment using the SVM classifier on computer-detected colours. This paper presents a QuadTree-based melanoma detection system inspired by dermatologists' color perception. Clinical color assessment in dermoscopy images is challenging because of subtle differences in shades, location-dependent color information, poor color contrast, and wide variation among images of the same class.

2.7 Neural Network Diagnosis of Malignant Melanoma From Color Images

Journal Name: IEEE

Year of Publication: 2000

Methodology:

In this study, we attempted to diagnose melanoma from color skin images using an artificial neural network. For this purpose, a set of features to distinguish melanoma from three other types of benign tumors was defined and methods to measure these features from digitized color slides were described.

Pros & Cons:

Overall, diagnostic results were found to be very promising, and as high as 86% accuracy in detecting malignant melanoma was achieved using training data sets of reasonable size.

The majority of cancer cases lie under the umbrella of nonmelanoma categories, such as basal cell carcinoma (BCC), squamous cell carcinoma (SCC), and sebaceous gland carcinoma (SGC). BCC, SGC, and SCC are formed in the middle and upper layers of the epidermis, respectively.

These cancer cells have a low tendency of spreading to other body parts. Nonmelanoma cancers are easily treated as compared with melanoma cancers.

CHAPTER 3

METHODOLOGY

3.1 PROPOSED SYSTEM

In order to improve the accuracy of feature extraction, eight different types of preprocessing algorithms were used. The algorithms used were converting to grey scale image, sharpening filter, median filter, smooth filter, binary mask, RGB extraction, and histogram and sobel operator. The RGB values of the images are extracted before converting it into a Gray scale image. Sharpening filter is applied to the Gray scale image in order to sharpen the details of the infected region, advanced feature will be added such as Entropy, kwotsis, Skewness etc.. We using Genetic Programming method.

3.1.1 Existing System

In the Existing system, we have focused on developing CAD systems for skin cancer detection. In hospitals, to detect the melanoma tissues, patients generally undergo a skin examination using the skin surface microscopy techniques commonly known as dermo copy. The structural features are obtained from multi resolution analyses which are used to discriminate the structures as borders, dots and streaks. On the other side, the textural features computed by LBP operators are used to thus discriminate the local variation of colours, the pigment network etc. Later, these features are fused in multiple combinations to investigate the influence of each combination in the performance of melanoma detection. They using CNN algorithm and SVM.

The number of deaths because of skin melanoma can be reduced if it is detected at its early stages. The most common method of diagnosing skin cancer is by visual examinations by dermatologists, which has an accuracy of about 60%. The diagnostic accuracy of skin cancers increases to 89% by using dermoscopy. When the input data to an algorithm is too large to be processed and it is suspected to be redundant, then it can be transformed into a reduced set of features (also named a features vector).

3.1.2 Flow Diagram

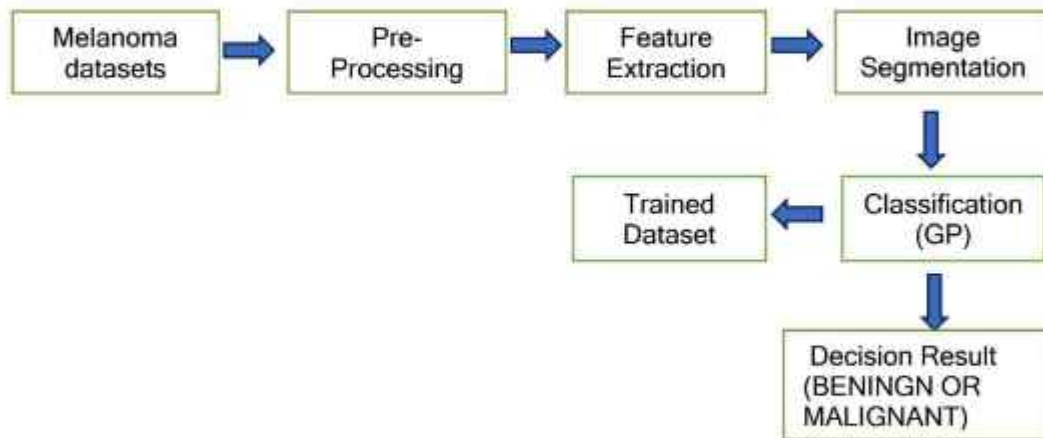


Fig 3.1 Flow diagram

Comparison of Proposed system from Existing system

Tabel: 3.1 Algorithm comparison

Model	Precision	Recall	F1-Score	Inference time(ms)
SSD-Mobilenetv2	0.42	0.56	0.479	7
YOLOv1	0.82	0.69	0.74	340
YOLOv2	0.81	0.76	0.78	33.7
YOLOv3	0.77	0.78	0.78	70.57
Tiny-YOLOv4	0.76	0.75	0.76	4.86
YOLOv4	0.81	0.83	0.82	52.51
YOLOv5	0.93	0.83	0.87	10

3.1.3 Feature Extraction

In image processing, feature extraction starts from an initial set of measured data and builds derived values (features) intended to be informative and non-redundant, facilitating the subsequent learning and generalization steps, and in some cases

leading to better human interpretations. Feature extraction is related to dimensionality reduction. When the input data to an algorithm is too large to be processed and it is suspected to be redundant, then it can be transformed into a reduced set of features (also named a features vector). This process is called feature selection. The selected features are expected to contain the relevant information from the input data, so that the desired task can be performed by using this reduced representation instead of the complete initial data.

General

Feature extraction involves reducing the amount of resources required to describe a large set of data. When performing analysis of complex data one of the major problems stems from the number of variables involved. Analysis with a large number of variables generally requires a large amount of memory and computation power, also cause a classification algorithm to overfit to training samples and generalize poorly to new samples. The best results are achieved when an expert constructs a set of application-dependent features, a process called feature engineering. Nevertheless, if no such expert knowledge is available, general dimensionality reduction techniques may help. These include:

- Independent component analysis
- Isomap
- Kernel PCA
- Latent semantic analysis
- Partial least squares
- Principal component analysis
- Multifactor dimensionality reduction
- Nonlinear dimensionality reduction
- Multilinear Principal Component Analysis
- Multilinear subspace learning
- Semidefinite embedding
- Autoencoder
- Deep feature synthesis

3.1.4 Wavelets

Wavelets are a relatively new area of signal processing and offer a potentially very useful approach to image analysis. Because of the fact that one has local way of control over resolution, many cue features of different size can be extracted from image. Wavelets also show promise as an aid in understanding how early vision can be emulated and further understood.

Wavelet Transforms

It is well known that in signal processing, the more compact your method of representing information the better. In 1822, Jean Joseph Fourier devised a very efficient way to represent the information content of a signal. His idea was to represent a signal as the sum of its frequencies. Transmitted power spectra, carrier frequencies, brain activity, NMR signals - all these global descriptions provide a lot of information in a compact manner. However, most of the power of this kind of representation vanishes when one tries to represent information that changes its nature during the course of signal recording. A good example of this kind of a signal is a musical score. A global analysis of a recording of a musical selection with a Fourier transform (FT) will indicate the specific notes played within the piece of music, but there is no way to recover the timing of the notes. The musical score, on the other hand, indicates each note that was played and the time it was played. Wavelets represent a signal in a way such that local frequency information is available for each position within the signal. Wavelets are able to analyze a signal based on the position-varying spectra. The multiresolution pyramidal decomposition that results is also well matched to fractals and shows great potential for the removal of background noise, such as static in recordings, and for pattern recognition and texture segmentation. In the following section we discuss the conceptual understanding of how wavelets can be used in signal analysis.

3.1.5 Genetic Programming

Genetic programming is a domain-independent method that genetically breeds a population of computer programs to solve a problem. Specifically, genetic programming iteratively transforms a population of computer programs into a new generation of programs by applying analogs of naturally occurring genetic

operations. The genetic operations include crossover (sexual recombination), mutation, reproduction, gene duplication, and gene deletion.

Preparatory Steps of Genetic Programming

The human user communicates the high-level statement of the problem to the genetic programming system by performing certain well-defined preparatory steps.

- The set of terminals for each branch of the to-be-evolved program.
- The set of primitive functions for each branch of the to-be-evolved program
- The termination criterion and method for designating the result of the run.

Genetic programming typically starts with a population of randomly generated computer programs composed of the available programmatic ingredients. Genetic programming iteratively transforms a population of computer programs into a new generation of the population by applying analogs of naturally occurring genetic operations. These operations are applied to individual(s) selected from the population. The individuals are probabilistically selected to participate in the genetic operations based on their fitness (as measured by the fitness measure provided by the human user in the third preparatory step). The iterative transformation of the population is executed inside the main generational loop of the run of genetic programming.

The executional steps of genetic programming (that is, the flowchart of genetic programming) are as follows:

1. Randomly create an initial population (generation 0) of individual computer programs composed of the available functions and terminals.

2. Iteratively perform the following sub-steps (called a generation) on the population until the termination criterion is satisfied:

(a) Execute each program in the population and ascertain its fitness (explicitly or implicitly) using the problem's fitness measure.

(b) Select one or two individual of this program(s) from the population with a probability based on fitness(with reselection)

(c) Create new individual program(s) for the population by applying the following genetic

(i) Reproduction: Copy the selected individual program to the new population.

(ii) Crossover: Create new offspring program(s) for the new population by recombining randomly chosen parts from two selected programs.

(iii) Mutation: Create one new offspring program for the new population by randomly mutating a randomly chosen part of one selected program.

(iv) Architecture-altering operations: Choose an architecture-altering operation from the available repertoire of such operations and create one new offspring program for the new population by applying the chosen architecture-altering operation to one selected program.

3.After the termination criterion is satisfied, the single best program in the population produced during the run (the best-so-far individual) is harvested and designated as the result of the run. If the run is successful, the result may be a solution (or approximate solution) to the problem.

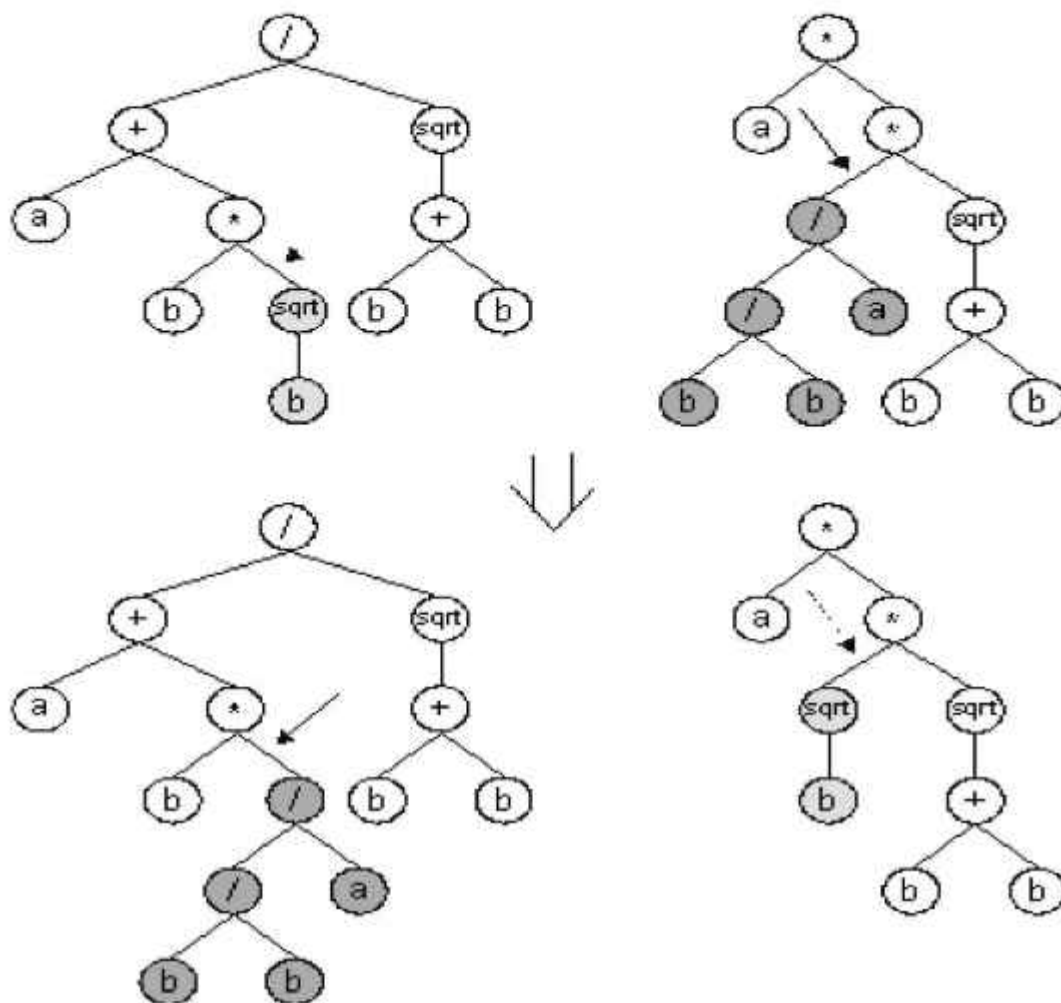


Fig: 3.2 Genetic Programming

3.2 Standards

(i)ISO 13485: Medical devices — Quality management systems — Requirements for regulatory purposes:

ISO 13485 sets out the requirements for a quality management system specific to the medical devices industry. Compliance with this standard is often necessary for regulatory approval of medical.

(ii) ISO/IEC 27001: Information technology — Security techniques — Information security management systems — Requirements:

This standard outlines best practices for establishing, implementing, maintaining,

and continually improving an information security management system (ISMS).It's crucial for ensuring the security of patient data and compliance with data protection regulations like HIPAA (Health Insurance Portability and Accountability Act) or GDPR (General Data Protection Regulation).

(iii) IEEE 11073 - Health informatics — Medical / Health Device Communication Standards:

IEEE 11073 standards specify the communication between medical devices and external computer systems.It ensures interoperability between different medical devices and systems, which is crucial in healthcare settings

3.2.1 Constraints

Limited availability of high-quality annotated melanoma images for training and testing algorithms.Variability in image quality due to factors such as lighting and the conditions, camera quality, and patient characteristics.Adherence to regulatory standards such as FDA regulations (if applicable), HIPAA, GDPR, and other data protection laws.Compliance with medical device regulations if the system is intended for clinical use.Ensuring patient privacy and obtaining informed consent for the use of medical images in research or clinical settings.Avoiding bias in algorithms that could disproportionately affect certain demographic groups.Ensuring that the image processing algorithms have high sensitivity and specificity for melanoma detection to minimize false positives and false negatives.Validation of algorithms on diverse datasets to assess performance across different skin types and melanoma subtypes.Limitations in computational resources for real-time processing, especially in resource-constrained environments such as mobile devices or remote clinics.

3.2.2 Trade Off

Developing a melanoma detection system using image processing involves navigating various considerations, including standards, constraints, tradeoffs, and objectives. Standards such as DICOM,

ISO regulations, and HIPAA ensure regulatory compliance, data security, and interoperability. Constraints encompass data quality, regulatory compliance, ethical

considerations, accuracy, computational resources, and integration with clinical workflow, among others. Tradeoffs must be carefully balanced throughout development. These include accuracy vs. computational complexity, sensitivity vs. specificity, cost vs. performance, interpretability vs. complexity, generalization vs. overfitting, usability vs. features, deployment speed vs. model training time, and privacy vs. data utilization

CHAPTER 4 METHODOLOGY

4.1 FLOW DIAGRAM DESCRIPTION

First the input image selected for identification of brain tumor type. The selected image then converted from RGB model to Grayscale model so that the pixel contains only the same weights. Then the image filtered by using median filter which removes the salt and pepper noise in the image. The filtered image then segmented using K-means clustering algorithm which segments the affected part of MELANOMA image. Then the features of the image (i.e., mean, variance, entropy, standard deviation, etc.,) are extracted by using the discrete wavelet transform (DWT) and principal component analysis (PCA). Finally the features of the input image are compared with the trained dataset using GP.

4.2 BLOCK EXPLANATION

4.2.1 *Grayscale Conversion*

A grayscale or greyscale 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, varying from black at the weakest intensity to white at the strongest.

Grayscale images are distinct from one-bit bi-tonal black-and-white images, which in the context of computer imaging are images with only two colors, black and white (also called bilevel or binary images). Grayscale images have many shades of gray in between.

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. But also they can be synthesized from a full color image;

Numerical representations

The intensity of a pixel is expressed within a given range between a minimum and a maximum, inclusive. This range is represented in an abstract way as a range from 0

(total absence, black) and 1 (total presence, white), with any fractional values in between. This notation is used in academic papers, but this does not define what "black" or "white" is in terms of colorimetry.

Another convention is to employ percentages, so the scale is then from 0% to 100%. This is used for a more intuitive approach, but if only integer values are used, the range encompasses a total of only 101 intensities, which are insufficient to represent a broad gradient of grays. Also, the percentile notation is used in printing to denote how much ink is employed in halftoning, but then the scale is reversed, being 0% the paper white (no ink) and 100% a solid black (full ink).

In computing, although the grayscale can be computed through rational numbers, image pixels are stored in binary, quantized form. Some early grayscale monitors can only show up to sixteen (4-bit) different shades, but today grayscale images (as photographs) intended for visual display (both on screen and printed) are commonly stored with 8 bits per sampled pixel, which allows 256 different intensities (i.e., shades of gray) to be recorded, typically on a non-linear scale. The precision provided by this format is barely sufficient to avoid visible banding artifacts, but very convenient for programming because a single pixel then occupies a single byte.

Technical uses (e.g. in medical imaging or remote sensing applications) often require more levels, to make full use of the sensor accuracy (typically 10 or 12 bits per sample) and to guard against roundoff errors in computations. Sixteen bits per sample (65,536 levels) is a convenient choice for such uses, as computers manage 16-bit words efficiently. The TIFF and the PNG (among other) image file formats support 16-bit grayscale natively, although browsers and many imaging programs tend to ignore the low order 8 bits of each pixel.

No matter what pixel depth is used, the binary representations assume that 0 is black and the maximum value (255 at 8 bpp, 65,535 at 16 bpp, etc.) is white, if not otherwise noted.

Converting color to grayscale

Conversion of a color image to grayscale is not unique; different weighting of the color channels effectively represent the effect of shooting black-and-white film with different-colored photographic filters on the cameras.

Syntax

```
I = rgb2gray(RGB)
```

```
newmap = rgb2gray(map)
```

`I = rgb2gray(RGB)` converts the truecolor image `RGB` to the grayscale intensity image `I`. The `rgb2gray` function converts `RGB` images to grayscale by eliminating the hue and saturation information while retaining the luminance. If you have Parallel Computing Toolbox™ installed, `rgb2gray` can perform this conversion on a GPU.

`newmap = rgb2gray(map)` returns a grayscale colormap equivalent to `map`.



Fig: 4.1 a)Normal image



Fig: 4.1 b)Converted

4.2.2 Median Filter

The median filter is a nonlinear digital filtering technique, often used to remove noise. Such noise reduction is a typical pre-processing step to improve the results of later processing (for example, edge detection on an image). Median filtering is very widely used in digital image processing because, under certain conditions, it preserves edges while removing noise (but see discussion below).

Algorithm description

The main idea of the median filter is to run through the signal entry by entry, replacing each entry with the median of neighboring entries. The pattern of neighbors is called the "window", which slides, entry by entry, over the entire signal. For 1D signals, the most obvious window is just the first few preceding and following entries, whereas for 2D signals such as images, more complex window patterns are possible. Note that if the window has an odd number of entries, then the median is simple to define: it is just the middle value after all the entries in the window are sorted numerically. For an even number of entries, there is more than one possible median.

Worked 1D example

To demonstrate, using a window size of three with one entry immediately preceding and following each entry, a median filter will be applied to the following simple 1D signal:

$x = [2 \ 80 \ 6 \ 4]$

So, the median filtered output signal y will be:

$y[1] = \text{Median}[2 \ 2 \ 80] = 2$

$y[2] = \text{Median}[2 \ 80 \ 6] = \text{Median}[2 \ 6 \ 80] = 6$

$y[4] = \text{Median}[80 \ 6 \ 4] = \text{Median}[4 \ 6 \ 80] = 6$

$y[4] = \text{Median}[6 \ 4 \ 4] = \text{Median}[4 \ 4 \ 6] = 4$

i.e. $y = [2 \ 6 \ 6 \ 4]$.

Boundary issues

Note that, in the example above, because there is no entry preceding the first value, the first value is repeated, as with the last value, to obtain enough entries to fill the window. This is one way of handling missing window entries at the boundaries of the signal, but there are other schemes that have different properties that might be preferred in particular circumstances:

- Avoid processing the boundaries, with or without cropping the signal or image boundary afterwards,
- Fetching entries from other places in the signal. With images for example, entries from the far horizontal or vertical boundary might be selected,
- Shrinking the window near the boundaries, so that every window is full.

2D median filter pseudo code

Code for a simple 2D median filter algorithm might look like this:

```
allocate outputPixelValue[image width][image height]
allocate window>window width * window height
edgex := (window width / 2) rounded down
edgey := (window height / 2) rounded down
for x from edgex to image width - edgex
  for y from edgey to image height - edgey
    i=0;
    for fx from 0 to window width
      for fy from 0 to window height
```

```

window[i] := inputPixelValue[x + fx - edgex][y + fy - edgey];
i := i + 1
sort entries in window[]
outputPixelValue[x][y] := window[window width * window height / 2]

```

Note that this algorithm:

- Processes one color channel only,
- Takes the "not processing boundaries" approach

Algorithm implementation issues

Typically, by far the majority of the computational effort and time is spent on calculating the median of each window. Because the filter must process every entry in the signal, for large signals such as images, the efficiency of this median calculation is a critical factor in determining how fast the algorithm can run. The "vanilla" implementation described above sorts every entry in the window to find the median; however, since only the middle value in a list of numbers is required, selection algorithms can be much more efficient. Furthermore, some types of signals use whole number representations: in these cases, histogram medians can be far more efficient because it is simple to update the histogram from window to window, and finding the median of a histogram is not particularly onerous.

Edge preservation properties

Median filtering is one kind of smoothing technique, as is linear Gaussian filtering. All smoothing techniques are effective at removing noise in smooth patches or smooth regions of a signal, but adversely affect edges. Often though, at the same time as reducing the noise in a signal, it is important to preserve the edges. Edges are of critical importance to the visual appearance of images, for example. For small to moderate levels of noise, the median filter is demonstrably better than Gaussian blur at removing noise whilst preserving edges for a given, fixed window size. However, its performance is not that much better than Gaussian blur for high levels of noise, whereas, for speckle noise and salt and pepper noise, it is particularly effective. Because of this, median filtering is very widely used in digital image processing.

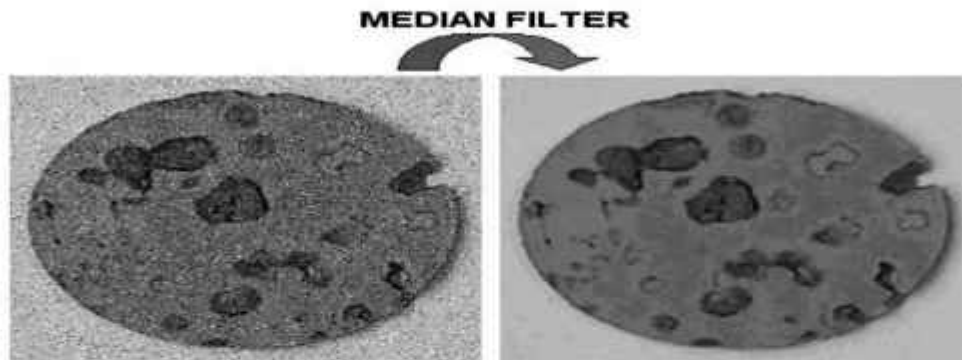


Fig: 4.2 Median filter

4.2.3 K-MEANS CLUSTERING

k-means clustering is a method of vector quantization, originally from signal processing, that is popular for cluster analysis in data mining. k-means clustering aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster. This results in a partitioning of the data space into Voronoi cells.

The problem is computationally difficult (NP-hard); however, there are efficient heuristic algorithms that are commonly employed and converge quickly to a local optimum. These are usually similar to the expectation-maximization algorithm for mixtures of Gaussian distributions via an iterative refinement approach employed by both algorithms. Additionally, they both use cluster centers to model the data; however, k-means clustering tends to find clusters of comparable spatial extent, while the expectation-maximization mechanism allows clusters to have different shapes.

The algorithm has a loose relationship to the k-nearest neighbor classifier, the popular machine learning technique for classification that is often confused with k-means because of the k in the name. One can apply the 1-nearest neighbor classifier on the cluster centers obtained by k-means to classify new data into the existing clusters. This is known as nearest centroid classifier or Rocchio algorithm.

4.2.4 Principal Component Analysis

Principal Component Analysis (PCA) (Rafael and Woods, 2002) is a well-known and one of the most successful techniques used in image recognition and compression for extracting feature and representing data. It is technique

widely used in the area of pattern recognition, computer vision and signal processing. The purpose of PCA is to reduce the large dimensionality of the data space (observed variables) to the smaller intrinsic dimensionality of feature space (independent variables), which are needed to describe the data economically. This is the case when there is a strong correlation between observed variables. By discarding minor components, the PCA effectively reduces the number of features and displays the data set in a low-dimensional subspace. In this study the feature extraction algorithm based on PCA is chosen.

Linear Discriminant Analysis (LDA), Independent Component Analysis and PCA are some of the techniques used for feature extraction, among them PCA is powerful method in image formation, Data patterns, similarities and differences between them are identified efficiently. The other main advantage of PCA is dimension will be reduced by avoiding redundant information, (Daugman, 1994) without much loss. Better understanding of principal component analysis is through statistics and some of the mathematical techniques which are Eigenvalues, Eigen vectors. PCA is a useful statistical and common technique that has found application in fields such as image recognition and compression. Principal Component Analysis (PCA) is a mathematical procedure that uses linear Transformations to map data from high dimensional space to low dimensional space.

The steps involved in PCA include:

- The mean value $\bar{S}$ of the given data set "S" is found
- Subtract the mean value $\bar{S}$ from S. from these values a new matrix is obtained. Let say "A"
- Covariance is obtained from the matrix i.e., $C = A A^T$ Eigen values are obtained from the covariance matrixes that are $V_1 V_2 V_4 V_4 \dots V_N$,
- Finally Eigen vectors are calculated for covariance matrix C.
- Any vector S or S⁻ can be written as linear combination of eigen vectors shown in Equation
- Because covariance matrix is symmetric it form basis $V_1 V_2 V_4 V_4 \dots$

$$V_N S^- = b_1 u_1 + b_2 u_2 + b_4 u_4 + \dots + b_N u_N$$

- Only Largest eigen values are kept to form lower dimension data set (Eq.)

$$\hat{s} - \bar{s} = \sum_{i=1}^L b_i u_i; 1 \leq N$$

5E

One of the difficulties inherent in multivariate statistics is the problem of visualizing data that has many variables. The MATLAB® function `plot` displays a graph of the relationship between two variables. The `plot4` and `surf` commands display different three-dimensional views. But when there are more than three variables, it is more difficult to visualize their relationships.

Fortunately, in data sets with many variables, groups of variables often move together. One reason for this is that more than one variable might be measuring the same driving principle governing the behavior of the system. In many systems there are only a few such driving forces. But an abundance of instrumentation enables you to measure dozens of system variables. When this happens, you can take advantage of this redundancy of information. You can simplify the problem by replacing a group of variables with a single new variable.

Principal component analysis is a quantitatively rigorous method for achieving this simplification. The method generates a new set of variables, called principal components. Each principal component is a linear combination of the original variables. All the principal components are orthogonal to each other, so there is none redundant information. The principal components as a whole form an orthogonal basis. There are an infinite number of ways to construct an orthogonal basis for several columns of data. What is so special about the principal component basis?

The first principal component is a single axis in space. When you project each observation on that axis, the resulting values form a new variable. And the variance of this variable is the maximum among all possible choices of the first axis.

The second principal component is another axis in space, perpendicular to the first. Projecting the observations on this axis generates another new variable. The variance of this variable is the maximum among all possible choices of this second axis.

The full set of principal components is as large as the original set of variables. But it is commonplace for the sum of the variances of the first few principal components to exceed 80% of the total variance of the original data. By examining plots of these few new variables, researchers often develop a deeper understanding of the driving forces that generated the original data.

You can use the function `pca` to find the principal components. To use `pca`, you need to have the actual measured data you want to analyze. However, if you lack the actual data, but have the sample covariance or correlation matrix for the data, you can still use the function `pcacov` to perform a principal components analysis. See the reference page for `pcacov` for a description of its inputs and outputs.

`coeff = pca(X)` returns the principal component coefficients, also known as loadings, for the n -by- p data matrix X . Rows of X correspond to observations and columns correspond to variables. The coefficient matrix is p -by- p . Each column of `coeff` contains coefficients for one principal component, and the columns are in descending order of component variance. By default, `pcacov` centers the data and uses the singular value decomposition (SVD) algorithm.

ACTIVE CONTOUR METHOD

Edge-based active contours are strongly connected to the edge-based segmentation. Most edge based active contour models consist of two parts: which attracts the contour towards the boundaries. Edge-based active contour models have a little disadvantages compared to the region-based active contour models, discussed in the next section.. Therefore, an primary contour must be placed completely inside or outside of ROI, and some level of a previous knowledge is still necessary .Also, edge-based active contours inherit a few disadvantages of the edge-based segmentation methods due to the parallel method used. Since both edge-based segmentation and edge-based active contours rely on the image gradient process, edge-based active contours may omit the blurry boundaries, and they are sensitive to local minima or noise as edge-based segmentation does. Gradient vector flow quick geodesic dynamic contours proposed by Paragios replaced the border detection word with gradient vector field , that refers to a spatial diffusion of the boundary information and guides the propagation to the object boundaries from equally sides, to give extra freedom from the restriction of first contour position. g) Region-based Active Contours Most region-based active contour models consist of two parts: the regularity part, which determines the smooth form of contours, and the energy minimization part, which searches for equality of a preferred feature within a subset. A good characteristic of region-based active contours is that the first contours can be situated anyplace in the image as region-based segmentation relies on the global energy minimization rather than

local energy minimization. Therefore, less previous knowledge is required than edge-based active contours. Although usual region-based active contours partition an image into several sub regions, those several regions belong to only two subsets: both the inside or the outside of contours. Chan and Vese proposed multi-phase active contour model, which increases the amount of subsets that active contours can locate simultaneously. Multiple active contours evolve independently based on the piecewise-constant model or the piecewise-smooth. In order to develop the segmentation performance, the integration of edge- and region based information sources using active contours has been proposed by a few authors. Geodesic active region is a supervised active contour model, proposed by Paragios, integrating edge- and region-based segmentation module in an energy function. A statistical analysis based on the Minimum Description Length (MDL) measure and the Maximum Likelihood (ML) principle for the observed density function, i.e., an image histogram, indicates the number of sub-regions and the statistical PDF within those sub-regions using a mixture of Gaussian elements. Regional probability is estimated from the statistical PDF based on previous knowledge, i.e. training samples. Then, the margin information is resolved by a probabilistic edge detector, expected from the regional probabilities of neighborhood. For example, an image pixel is more likely an edge pixel if the neighbourhoods pixels, located on the opposed sides, have high regional probabilities for a different class.

4.3 SYSTEM REQUIREMENTS FOR MATLAB

Operating Systems

- Windows 10
- Windows 8.1
- Windows 8
- Windows 7 Service Pack 1
- Windows Server 2016
- Windows Server 2012
- Windows Server 2008 R2 Service Pack 1

Processors

- Any Intel or AMD x86-64 processor

- AVX2 instruction set support is recommended
- With Polyspace, 4 cores is recommended

Disk Space

- GB for MATLAB only,
- 4–6 GB for a typical installation.

RAM

- 16 GB
- With Simulink, 4 GB is required
- With Polyspace, 4 GB per core is recommended

Graphics

- No specific graphics card is required.
- Hardware accelerated graphics card supporting OpenGL 4.4 with 1GB GPU memory is recommended.

4.4 MATLAB

What Is MATLAB?

MATLAB is a high-performance language for technical computing. It is used to integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation. Typical uses include:

- Math and computation
- Algorithm development
- Modeling, simulation, and prototyping
- Data analysis, exploration, and visualization
- Scientific and engineering graphics
- Application development, including graphical user interface building

MATLAB is an interactive system whose basic data element is an array that does not require dimensioning. This allows 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 noninteractive language such as C or Fortran.

The name MATLAB stands for *matrix laboratory*. MATLAB was originally written to provide easy access to matrix software developed by the LINPACK and EISPACK projects. Today, MATLAB uses software developed by the LAPACK and ARPACK projects, which together represent the state-of-the-art in software for matrix computation. MATLAB has evolved over a period of years with input from many users. In university environments, it is the standard instructional tool for introductory and advanced courses in mathematics, engineering, and science. In industry, MATLAB is the tool of choice for high-productivity research, development, and analysis.

Toolboxes

MATLAB features a family of application-specific solutions called toolboxes. Very important to most users of MATLAB, toolboxes allow you to learn and apply specialized technology. Toolboxes are comprehensive collections of MATLAB words

functions (M-files) 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.

The MATLAB System

The MATLAB system consists of five main parts:

Development Environment

This is the set of tools and facilities that help you use MATLAB functions and files. Many of these tools are graphical user interfaces. It includes the MATLAB desktop and Command Window, a command history, and browsers for viewing help, the workspace, files, and the search path.

The MATLAB Mathematical Function Library

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

The MATLAB language

This is a high-level matrix/array language with control flow statements, functions, data structures, input/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.

Handle Graphics®

This is the MATLAB graphics system. It includes high-level commands for two-dimensional and three-dimensional data visualization, image processing, animation, and presentation graphics. It also includes low-level commands that allow you to fully customize the appearance of graphics as well as to build complete graphical user interfaces on your MATLAB applications.

The MATLAB Application Program Interface (API)

This is a library that allows you to write C and Fortran programs that interact with MATLAB. It includes facilities for calling routines from MATLAB (dynamic linking), calling MATLAB as a computational engine, and for reading and writing MATLAB files.

Image processing in matlab

Images and pictures. As we mentioned in the preface, human beings are predominantly visual creatures: we rely heavily on our vision to make sense of the world around us. We not only look at things to identify and classify them, but we can scan for differences, and obtain an overall rough *_feeling_* for a scene with a quick glance. Humans have evolved very precise visual skills: we can identify a face in an instant; we can differentiate colours; we can process a large amount of visual information very quickly.

However, the world is in constant motion: stare at something for long enough and it will change in some way. Even a large solid structure, like a building or a mountain, will change its appearance depending on the time of day (day or night); amount of sunlight (clear or cloudy), or various shadows falling upon it. We are concerned with single images: snapshots, if you like, of a visual scene. Although

image processing can deal with changing scenes, we shall not discuss it in any detail in this text.

For our purposes, an image is a single picture which represents something. It may be a picture of a person, of people or animals, or of an outdoor scene, or a microphotograph of an electronic component, or the result of medical imaging. Even if the picture is not immediately recognizable, it will not be just a random blur.

What is image processing?

Image processing involves changing the nature of an image in order to either

1. improve its pictorial information for human interpretation,
2. render it more suitable for autonomous machine perception.

We shall be concerned with digital image processing, which involves using a computer to change the nature of a digital image (see below). It is necessary to realize that these two aspects represent two separate but equally important aspects of image processing. A procedure which satisfies condition,

- (1) A procedure which makes an image look better may be the very worst procedure for satisfying condition
- (2) Humans like their images to be sharp, clear and detailed; machines prefer their images to be simple and uncluttered.

Some applications

Image processing has an enormous range of applications; almost every area of science and technology can make use of image processing methods. Here is a short list just to give some indication of the range of image processing applications.

1. Medicine

Inspection and interpretation of images obtained from X-rays, MRI or CAT scans, analysis of cell images, of chromosome karyotypes.

2. Agriculture

Satellite/aerial views of land, for example to determine how much land is being used for different purposes, or to investigate the suitability of different regions for different crops, inspection of fruit and vegetables_distinguishing good and fresh produce from old.

3. Industry

Automatic inspection of items on a production line, inspection of paper samples.

4. Law enforcement

Fingerprint analysis, sharpening or de-blurring of speed-camera images

CHAPTER 5

RESULTS AND DISCUSSION

5.1 RESULTS THROUGH IMAGE SEGMENTATION

- The objective of the paper is to find better and more efficient ways to automatically detect early malignant melanoma using digital image processing techniques. This paper focuses on the preprocessing stage, the image analysis of the tumor.
- The detection of Melanoma cancer in early stage can be helpful to cure it.
- A threshold limit was imposed on the Euler value heuristically, exceeding which was an indicator of presence of a large number of inflections. This is an important distinguishing feature characteristic for diseases.
- Segmentation separates the suspicious lesion from normal skin. There are some unique features that distinguish the benign from malignant melanoma.

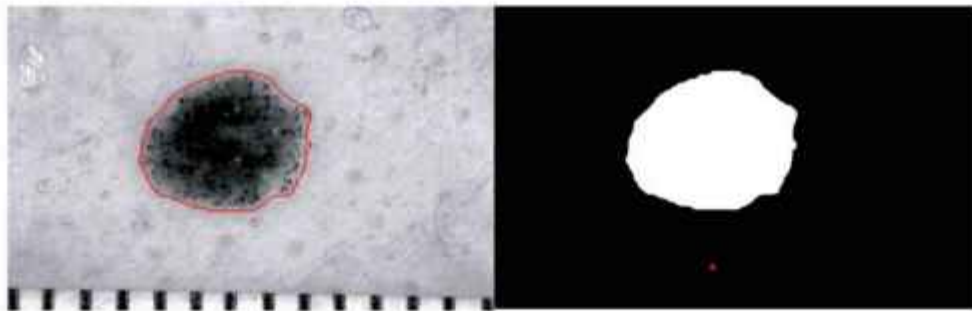


Fig: 5.1 Result of the melanoma test set segmentation.

- When the image segmentation process has a strong local deviation, it does not necessarily occupy a large volume, but will lead to larger shape differences.
- The segmentation of the melanoma image using the neural network automatic segmentation algorithm was compared in this study to the division of the image by an expert clinician.
- If the automatic segmentation algorithm is similar to that produced by the expert, then it is considered to be an acceptable alternative method for achieving expert-level results.

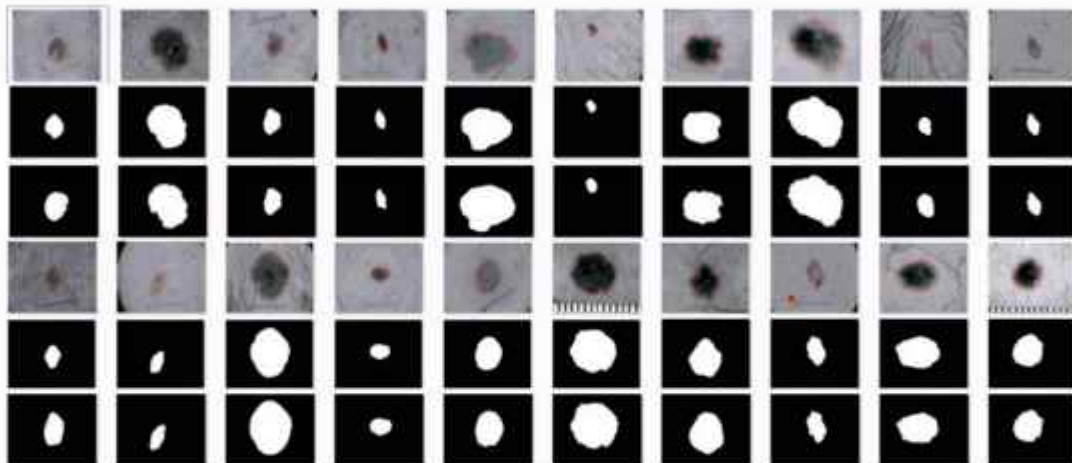


Fig: 5.2 20 melanoma image segmentation results.

- The first and fourth rows are the result of the segmentation of the 20 melanoma images using the improved CNN, the lesion area is circled with the red line;
- The second and fifth rows are the mask of the test image segmentation; the third and sixth rows are Experts' manual segmentation, ie ground truth mask.
- The results of the test segmentation are compared with the ground truth.

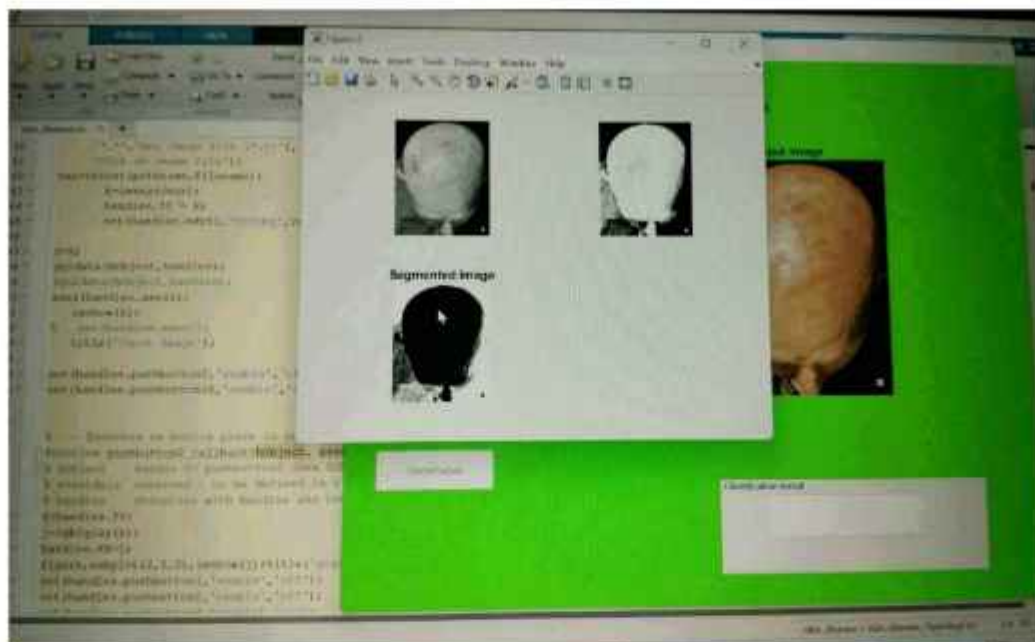


Fig: 5.3 Results through Image Segmentation

5.2 RESULTS THROUGH CLASSIFICATION

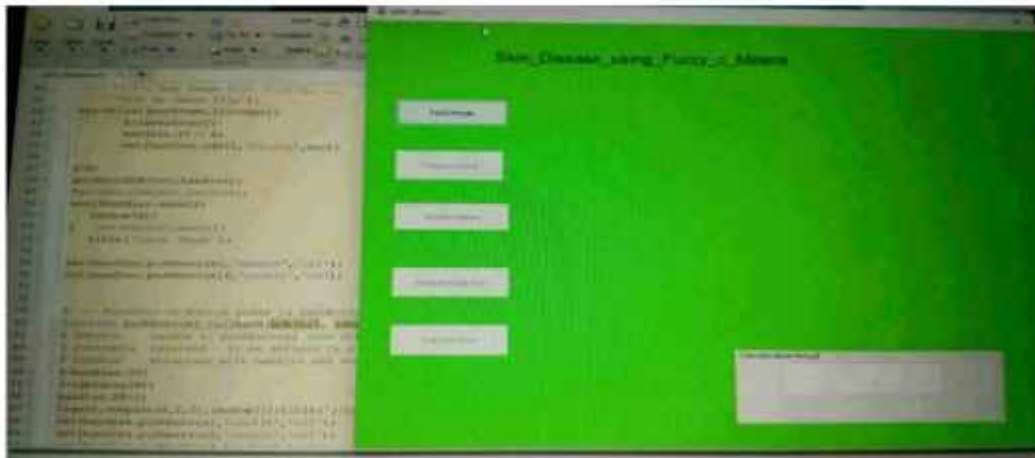


Fig: 5.4 Classification

The ROC curve analysis further confirms the model's strong discriminatory power, as evidenced by a high AUC value. These findings highlight the potential of the developed model as a valuable tool for dermatologists and healthcare professionals in early melanoma detection, ultimately improving patient outcomes through timely intervention and treatment.

5.4 FINAL OUTPUT

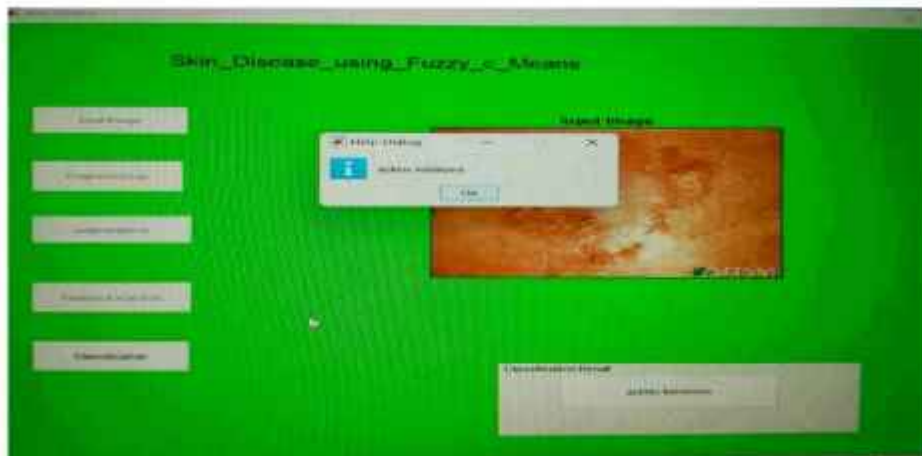


Fig: 5.5 Final Output

CHAPTER 6

CONCLUSION AND FUTURE ENHANCEMENT

6.1 CONCLUSION

The skin lesion classification in metadata- embedded images using a deep CNN (Inception-ResNet-v2). The results indicate that the proposed method improved the skin lesion classification performance by at least 5% by including the patient's metadata (i.e. the anatomical site of the lesion, age, and gender) as the model's input data. The proposed method achieved 89.4% in the classification of 4 major skin conditions and 94.5% in distinguishing between malignant and benign lesions. Future work in this field should focus on developing larger public datasets, which provide metadata to facilitate the research community's effort and to enhance the deep learning-based automated classification of skin lesions. In addition, future studies are warranted to investigate the impact of augmentation algorithms such as cutout regularization on CNN performance.

6.2 FUTURE ENHANCEMENT

Future research can include full-body photography to seek the answer to the question that typically arises. Autonomous full-body photography will automate and speed up the image acquisition phase. The idea of auto-organization has recently emerged within the area of deep learning. Auto-organization refers to the process of unsupervised learning, which aims to identify features and to discover relations or patterns in the image samples of the dataset. Under the umbrella of convolutional neural networks, auto-organization techniques increase the level of features representation that is retrieved by expert systems. Currently, auto-organization is a model that is still in research and development. However, its study can improve the accuracy of image processing systems in the future, particularly in the area of medical imaging, where the smallest details of features are extremely crucial for the correct diagnosis of disease.

REFERENCES

- [1] Abedini .M, Codella .N, Cai .J, Garnavi .R, Halpern .A, and Smith .J.R, "Deep learning, sparse coding, and svm for melanoma recognition in dermoscopy images," in *Machine Learning in Medical Imaging*. Springer, 2015, pp. 118-126.
- [2] Barkana B.D, Faezipour .M and Abuzagheh .O, "SKINcure : A real time image analysis system to aid in the malignant melanoma prevention and early detection" in *Proc. IEEE Southwest Symp. Image Anal. Interpretation (SSIAI)*, Apr. 2014, pp.85_88.
- [4] Caffrey .T.J ,Jerant .A.F, Johnson .J.T, Sheridan .C, et al., "Early detection and treatment of skin cancer," *American family physician*, vol. 62, no. 2, pp. 457-486, 2000.
- [4] Dhawan A.P ,Patwardhan S.V., Relue P.A.: 'Classification of melanoma using tree structured wavelet transforms', *Comput. Methods Programs Biomed.*, 2004, 72, (4), pp. 224-249
- [5] Iyatomi H., Celebi M.E, Oka H. : 'An improved internet-based melanoma screening system with dermatologist-like tumor area extraction algorithm', *Comput. Med. Imaging Graph*, 2008, 42, (7), pp. 566-579
- [6] Mishra .N.K and Celebi .M.E , "An overview of melanoma detection in dermoscopy images using image processing and machine learning," *arXiv preprint arXiv:1601.07844*, 2016.
- [7] Nascimento .J.C, Marques .J.S, Marc .A.R, al, Mendonc .T,a, Maeda .J , Rozeira .J ,Silveira .M, Yamauchi .S, "Comparison of segmentation methods for melanoma diagnosis in dermoscopy images," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 4, no. 1, pp.45-45, 2009.