M. Tech(Signal Processing and Digital Design)

Mayank Singh

2021

Text Extraction from Palm Leaf Manuscripts

A DISSERATATION

SUBMITTED IN PARTIAL FULFILLMENT O F THE REQUIREMENTS

FOR THE AWARD OF THE DEGREE

OF

MASTER OF TECHNOLOGY

IN

Signal Processing and Digital Design

Submitted by

Mayank Singh

(2K19/SPD/12)

Under the supervision of

Prof. S. Indu (Prof., ECE Dept.)

DEPARTMENT OF ELECTRONICS AND COMMUNICATION

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi -110042

JUNE 2021

CONTENTS

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

CANDIDATE'S DECLARATION

I **Mayank Singh** student of MTech (Signal Processing and Digital Design), hereby declare that the project Dissertation titled "**Text Extraction from Palm Leaf Manuscripts"** which is submitted by me to the Department of Electronics and Communication Engineering, Delhi Technological University, Delhi in partial fulfilment of the requirement for the award of the degree of Master of Technology, is original and not copied from any source without proper citation. This work has not previously formed the basis for the award of any Degree, Diploma Associateship, Fellowship or other similar title or recognition.

Place: Delhi
Date: 30th July 2021

Mayank Singh

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

CERTIFICATE

I hereby certify that the Project Report titled "**Text Extraction from Palm Leaf Manuscripts**" which is submitted by **Mayank Singh**, **2K19/SPD/12** of Electronics and Communication Department, Delhi Technological University, Delhi in partial fulfilment of the requirement for the award of the degree of Master of Technology, is a record of the project work carried out by the students under my supervision. To the best of my knowledge this work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.

Place: Delhi
Date: 30th July 2021

Prof. S. Indu
SUPERVISOR

ACKNOWLEDGEMENT

Completing M. Tech is the greatest achievement of my life so far. Being a student at DTU has been a really wonderful experience. I have enjoyed a lot in the crests and troughs throughout this journey.

I would like to pay my sincere gratitude to my supervisor, Prof. S. Indu, for her enlightening guidance, encouragement and support throughout my research work. Without her help, this work could not have been completed.

I would also like to thank all of the faculty members, laboratory staff, office staff and other members who are attached to the department of Electronics and Communication.

Date:30th July 2021 **Mayank Singh**

ABSTRACT

Text extraction is a very important step in language processing and pattern recognition. The primary objective of this work is to extract text from the palm leaf manuscripts. The two color spaces used here are very standard ones as in BGR color space and HSV color space which stands for Blue, Green, Red and Hue, Saturation, Value respectively.

The method proposed here first converts the images to the HSV color space and then their Hue, Value, and Saturation is varied manually with the help of the slide-bars introduced. The experimental results show the algorithm used to extract the text of a palm leaf manuscript performs fairly well.

LIST of FIGURES

LIST of TABLES

Chapter 1: INTRODUCTION

1.1. Overview

Text is the origin of data that is implanted into documents or images. Text-based information can be interpreted with ease by computers and humans. Images have become an interactive means of communication in this digital era. Nowadays, people send text embedded images to others. This is growing in trend in social platforms such as Facebook, WhatsApp, Twitter, etc. These images are efficient in describing one's emotions and feelings rather than only text message. Significant details can be drawn out from WWW pictures on the Internet employed in efficient web search. Additionally, images in text aids several image applications which are content based like video indexing, image searching on web, computer and human interaction systems and mobile-based text analysis.

The process of extracting named entities from unstructured text such as: real-world images, some unrecognizable text, manuscripts etc is termed as text extraction. It is composed of several steps: Detection, localisation, orientation, derivation, improvisation and identification.

The three initial steps are usually assumed to be one, but they are quite different from each other. The detection detects text in every single image. The text present in the picture is identified by localization around which the bounding boxes are created. The location is determined and is stored in the memory to decrease the pre-processing time. In the extraction step the image is remodelled into a binary image after which the image is divided into foreground and background where the text is preferred to be on the foreground after which the improvement is done to yield text with a better resolution. Optical character recognition (OCR)[1] or the Natural Language Processing (NLP) [2] technique helps in recognition, which consequently transforms an analog (printed) text to a digital file made of ASCII characters in lieu of analog file (pixels). These text extraction steps are explained in more detail below.

1.1.1. Text Detection

Pixel intensity is used to identify if, whether or not the images consist of text, the text detection step checks for the text present in the given image, because of the lack of prior knowledge. It is assumed that background pixels will have a lower pixel intensity than the text. Therefore, text pixels are those who have a higher value than a fixed threshold and relevant color difference with the neighbourhood pixels. For videos, text detection is done by employing scene change data between two consecutive video frames.[3]

1.1.2. Text Localization

It traces the location of text in a given image. Text can be efficiently located if the shape of text region is circular or rectangular. However, if the text alignment is irregular then it is hard to locate it. Text localization based on similarity in several features such as color, shape, intensity, size, and distance between two text pixels.[3]

1.1.3. Text Tracking

Text tracking is done when text localization cannot locate the text in the image. It preserves the morality of position over adjacent frames and decreases the time for processing text localization. Text localization results are cross checked in this step. Text extraction, detection and localization are often used interchangeably.[3]

1.1.4. Text Extraction and Recognition

It reveals how the text is segregated from the images. The text pixels are separated from other non-text ones. Text can be segmented with the use of features like stroke width, similar font or orientation. One of the most commonly used method for extraction and text recognition is OCR. As there is distortion and noise present in an image therefore OCR cannot be efficiently used for text recognition in an image but it can easily identify text in a text document. Different methods like texture-based and region-based method are used to extract the text more efficiently[3][4][5].

1.1.5. Image Enhancement

Image enhancement is performed so that the image clarity can be improved so that the human eye can perceive more details and also for further image processing operations. It

is done by eliminating blurriness and noise. To remove the noise introduced by the camera and minimize the missing pixels, Image smoothing is performed. Blurriness in an image can be removed by performing Image sharpening. To remove noise from an image different filters like mean filter, median filter etc are used[3][6].

The complete process described above can be illustrated by the block diagram below:

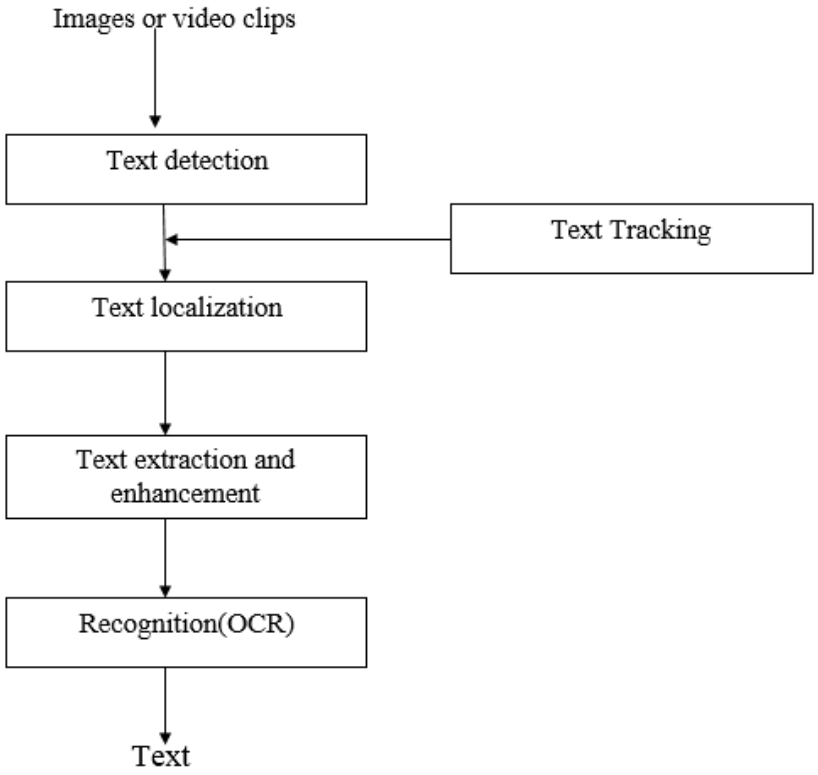


Figure 0: Flow chart of the steps of text extraction

1.2. Problems encountered in text extraction

The efficiency of a specific algorithm is determined by how well it detects the text present in the image. This property is exploited in this paper and the resultant image is a clear evidence of the effectiveness of the proposed method. The difficulty in text extraction relies on different aspects like height, length, depth of the text and the background complexity.

1.2.1. Issues in Text Detection

Most of the methods available for text detection are investigatory method in a top-down fashion and are experimental that's why almost every time it is relied visual features which are manually designed. Because of difficulty in modelling the complex background of the images most of the time these methods detect too many wrong things. The more the text in the image the more wrong things are identified, typically in the ratio of 80%-500%. As the machine learning algorithms obtain text by detecting it using a set of training examples, therefore, they are regarded as improvement over the previously available methods. But building text detection algorithms using machine learning also have its set of problems.

i. Making algorithm more efficient by avoiding working on the whole image and just work in the desired part. Machine learning algorithm's computation cost are associated with their capacities. For example, A neural network with more hidden units will have a more computation cost than the one having less hidden units. Because of the large amount of different backgrounds, a complex model is required to train an ANN for differentiating these backgrounds. Also, performing ANN classification on the whole image is costly when the image is large.

ii. Before training variances of character size and grayscale image has to be reduced. Otherwise, these various variations will become an obstacle in the training of a good text detection model. Further advancement in the field of machine learning is expected to improve the text detection system's performance.

1.2.2. Issues in text recognition

Majority of the previous methods used for recognition in complex videos or images focus on improvising the binarization method [7] prior to OCR module application [1]. Although, an optimal binarization might be tough to achieve when the background is compound and the grayscale distribution evinces various modes. Additionally, the grayscale values of text may be unknown in advance or even variable in a single string of

text. In order to recognize text strings extracted from images or video frames efficiently, the following issues need to be resolved:

i. To enhance different sized text characters: The enhancement of text [6][8] eliminates backgrounds from text images by making use of the features of the substructures of characters. Nevertheless, finite-size filters are incapable in enhancing text characters of varying sizes. Thus, an enhancing technique for variable sizes of characters is needed to achieve an optimal text binarization[7].

ii. To divide and acknowledge the text characters without knowing the grayscale values: Prior literature has assumed that the text is black or white in videos or images. Which is not always correct because even in one word or string the text can have different grayscale values. Besides, the grayscale vales of the extracted text are not always just black or white. As more than two clusters of grayscale values are present in videos and even some text images. The text recognition algorithm is only applied if it has the text recognition capability for different grayscale values.

iii. To incorporate the information present in successive video frames. Previous methods introduced for this purpose are centered on the point of separating the foreground and background of every video frame but this can only be applied on white or black grayscale value text. The text grayscale is repressed to be same throughout frames for which a correct alignment images' text is needed.

1.3. Applications of text extraction

1.3.1. Data extraction in digital libraries

The structured data used in digital libraries (DL) to help users to find and operate on images and documents is called "metadata". It also helps the search engines to find the searched document more correctly. There is a downside to metadata, that it takes too long and too much effort for it to be created. There have been many researches in the field of reducing the hard work of producing the metadata and make that process automatic using information extraction. Here, the digital library, Citeseer, is taken an example.

Citeseer is a popular specialty scientific and academic DL developed by NEC Labs. This is hosted on the World Wide Web at the College of Information Sciences and Technology, the Pennsylvania State University. It has more than 700,000 documents, majorly in the fields of computer and information science and engineering[9][10]. Citeseer gathers documents on the web, extracts metadata from them automatically and indexes this metadata to allow querying by metadata.

Citeseer elucidates 15 meta-tags for the document header, including Title, Author, Affiliation, and so on by extending Dublin Core metadata standard. They perceive the task of generation of document metadata automatically as that of labelling the text with the respective meta-tags. Every individual meta-tag correlate to a metadata class. The task of extraction is cast as a classification issue [11] and SVM [12] is used for this purpose. They reveal that categorisation of each text line into one or more classes is efficient for meta-tagging [13] than that of each word. This segregates the metadata extraction issue into two sub-problems: (1) line classification and (2) chunk identification of multi-class lines.

1.3.2. Information Extraction from Emails

One of the most popular means for communication via text is Email. 40 to 50 emails are received by an average PC user every day[14]. Email is provided as an input to many text mining[15] applications like, email routing, email analysis, information extraction from email, newsgroup analysis and email filtering. Alas, not much attention has been gathered by concept of information extraction from emails. Varying amount of information can be included in the email. Especially, it can include signatures, headers, text content and quotations. Additionally, it might also have lists, paragraphs and program codes. Just the header might have additional information like subject, receiver, sender etc. Also, the signature can have some additional information like sender's name, sender's position, sender's address, etc.

A problem is termed as a detection procedure for several kinds of data blocks (it consists of signature, header, program code, quotation, list and paragraph detections) and block-metadata detection (consisting of observation of signature and header). This method extracts email in a 'cascaded' manner. Extraction is done by running various passes of

processing on it. Initially, the entire body of the email is scanned for text (text-block detection), then it is segregated into content (paragraph detection), and lastly it is divided into blocks (signature and header detection). This division is helpful for classification using SVM (Support Vector Machine).

```
1.  From: SY <sandeep....@gmail.com> - Find messages by this author
2.  Date: Mon, 4 Apr 2005 11:29:28 +0530
3.  Subject: Re: ..How to do addition??

4.  Hi Ranger,
5.  Your design of Matrix
6.  class is not      good.
7.  what are you doing with two
8.  matrices in a single class?make class Matrix as follows

9.   import java.io.*;
10.  class Matrix {
11.  public static int AnumberOfRows;
12.  public static int AnumberOfColumns;

13.  public void inputArray() throws IOException
14.  {
15.     InputStreamReader input = new InputStreamReader(System.in);
16.     BufferedReader keyboardInput = new BufferedReader(input)
17.  }

18.  -- Sandeep Yadav
19.  Tel:  011-243600808
20.  Homepage: http://www.it.com/~Sandeep/

21.  On Apr 3, 2005 5:33 PM, ranger <asiri....@gmail.com> wrote:
22.  > Hi... I want to perform the addtion in my Matrix class. I got the program to
23.  > enter 2 Matricx and diaplay them. Hear is the code of the Matrix class and
24.  > TestMatrix class. I'm glad If anyone can let me know how to do the addition.....Tnx
```

Figure 0: Example of email message

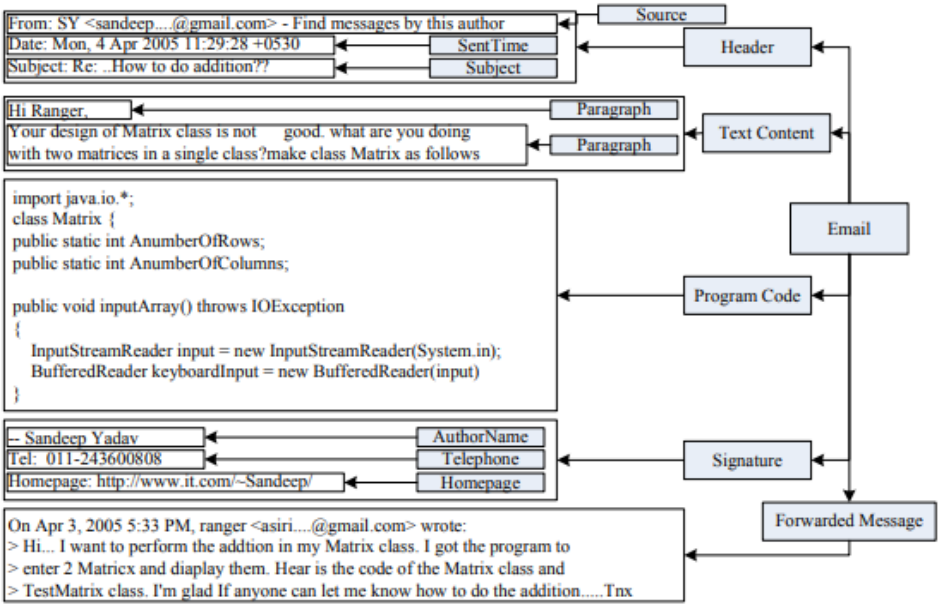


Figure 0: Cascaded Approach of Text Extraction

1.3.3. Person Profile Extraction

How a person's information is managed is a principal subject for both industrial community and research community. An abundant amount of information can be related to a single person such as: profile of that person (including homepage, portrait, affiliation, position, documents and publications), correspondence information (including telephone, address, fax number and email), and social networking info (including professional or person connections between people, like friendship). But these details are mostly concealed in varying and scattered web pages.

After some research it is concluded that most of the information about a person is concealed in that person's homepage, list (e.g., a faculty list), email message (e.g., in signature) and introduction page (web page that introduces the person). A classification method is employed to acquire the information from various sources.

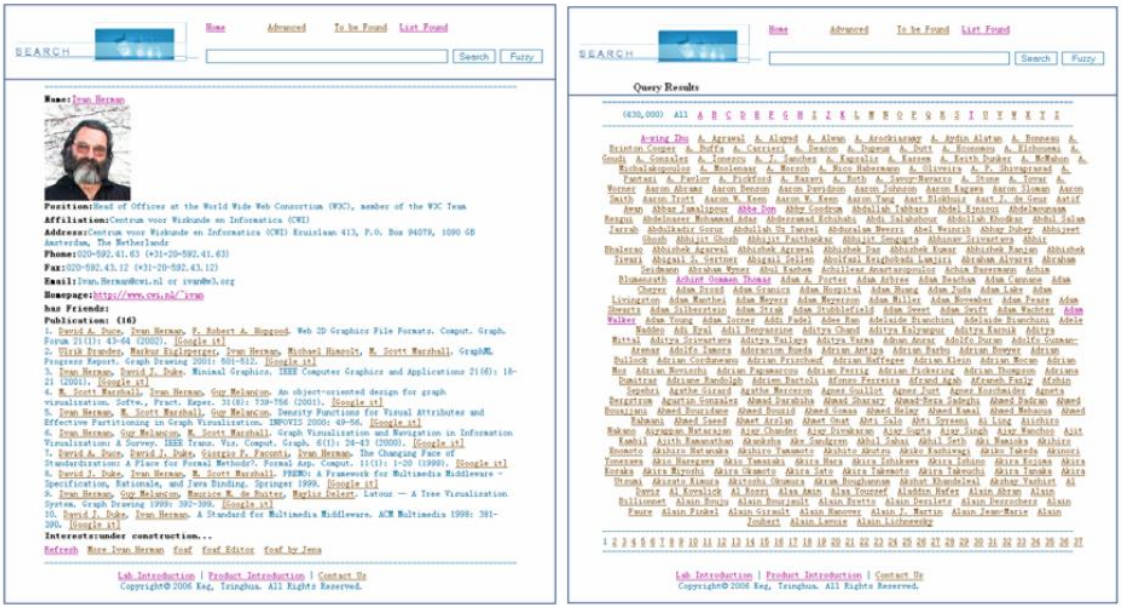


Figure 0: Snapshots of the Personal Network Search (PNS) system

As models, SVM (Support Vector Machine) is used. For every type of information, features are defined in the SVM model. A method called 'Personal Network Search'(PNS) is made which depends on the information extracted. In PNS, if the input given is the user's name, the output will the person's information.

1.4. Related work

1.4.1. Text extraction

Existing methods tried to alleviate some of these problems. An algorithm was proposed by Zhong et al.[16] based on RGB histogram, which was calculated using a finite number of classes. Their distinct peaks were drawn out again and again. All derived portions were then categorised into “non-text” or “text”. Zhou et al.[17] combined RGB color distance with spatial color proximity in the image space and proposed a graduated clustering algorithm. Some interesting results(partially) were yielded by the algorithm based on images obtained from compound internet streamers. A continuous approach was proposed by Sin et al.[18] where each region was transfigured into a single dimension sequence. If the Fourier spectrum’s auto-correlation function contained a relevant amount of peaks the region was affirmed as a text region.

Amid the recent research in text extraction, Sumathi et al. (2012) presented a detailed description of methods already in existence. In 2014 a novel algorithm was proposed by them for text extraction in the image. This technique allowed to omit the image background on the basis of Gamma Correction Method (GCM)[19]. Yet, it is time consuming for text extraction due to tedious calculations of various tasks in various GCM blocks which is disadvantageous for real-time purposes.

Another recent algorithm was proposed by Mohamed et al.[20], an effective Gamma Correction Method acceleration referred as New Gamma Correction Method (NGCM) which is the derivative of GCM. NGCM reduces the execution time of GCM while ensuring the same reliability. The gamma values in various portions of the image are predicted and then the corresponding pixels are transformed relative to the predicted gamma value. This method does not incorporate the depth of the text that’s why it cannot be used on manuscript images.

The algorithm proposed here is used for text extraction in palm leaf manuscripts. Text extraction here is performed excluding the text recognition step. The working process of the algorithm introduced here is much simpler than all of the previous algorithm as here

the image is transfigured to HSV color space[21]. Consequently, it's saturation, hue and value estimates are varied manually just as the output the user needs.

1.4.2. Denoising

The existing noise removal algorithms are researched very thoroughly. An algorithm performing binarization over a wide range of parameter value and hence predicting the correct value by repeatedly narrowing the parameters' range. was introduced by Badekas and Papamarkos[22]. A smoothing algorithm was established by Lu et al.[23] based on iterative polynomial where the identified stroke edges in characters decides the local threshold. A technique was invented by Lelore and Bouchara[24] where pixels were classified into different groups like background, foreground etc. A parameter independent algorithm was invented by Gatos et al[25]. Multiple threshold values were combined to obtain a single binarization value based on a cross-section sequence in Dawoud's method[26].A global energy function is optimized drawing on a in Howe's approach[26]. Non local means filtering algorithm is a method in which the filtering parameter depends on the standard deviation of the noise was developed by A. Buades, B. Coll, J. Morel[27]. Baysian Optimization is used to binarize a document image in the algorithm introduced by Vats et al.[28]. Nikolaos Papamarkos, Nikolaos Mitianoudis, implemented Document Image Binarization was performed using Gaussian Mixture Modelling and Local Features[29].

The foreground and background contrast difference in the palm leaves makes it such that most of the algorithm are not suitable for denoising. Thus, algorithms especially suitable this task are introduced by researchers. A technique where the color histogram is used to determine the background color is introduced by Z. Shi, S. Setlur and V. Govindaraju[30]. A binarization method with a cognitive memory network basis was introduced by Kumar et al[31]. Gray-level Histograms were used to develop a method based on threshold selection by N. Otsu[32]. Niblack executed a binarization method that does threshold calculation using a fixed size sliding window which moves across the image[33]. Adaptive document image binarization technique[34] was developed by Sauvola and M. Pietikainen. This method enhances the Niblack one with the addition of dynamic standard deviation.

1.5. Image database

To build up the database we have collected various palm leaf manuscripts images from different sources. Some relevant information on the image database are as follows:

- The database has around 2000 images approximately.
- Most of the images are in RGB and include text in every image as the images are of manuscripts.
- The database contains noisy as well as images of damaged manuscripts.

1.6. Proposed Methodology

1.6.1. BGR to HSV based text extraction from manuscripts using slidebars

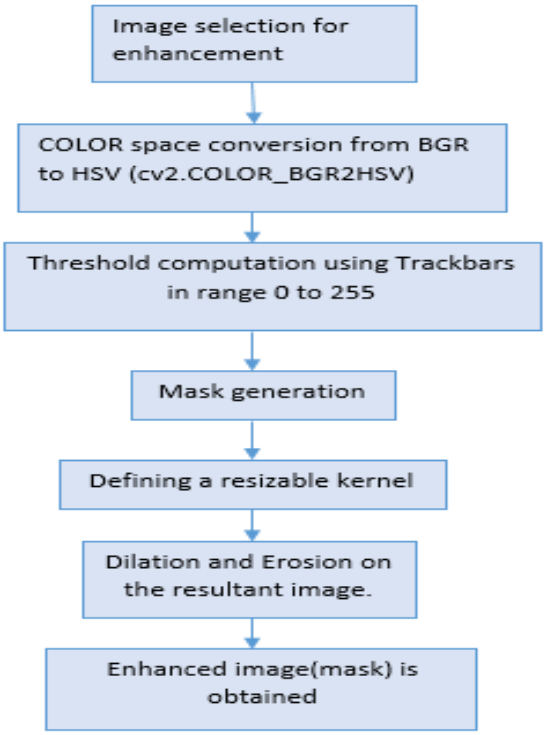


Figure 1.5: Flow chart of obtaining mask

The flow-chart for the proposed methodology is shown here:

- An image is chosen from the numerous images present in the dataset collected. The image is chosen such that some of the text is discernible and some is not which is retrieved through image enhancement[6] done here.

- The color space is being changed from the initial BGR to HSV[35].
- After which the slidebars (Trackbars) are varied according to every image and an ideal value is found on which the best image is obtained.
- Dilation and Erosion[36] is performed on image/mask obtained after the previous step.
- Final image is obtained

1.6.2. Denoising of Palm leaf manuscripts using Gaussian filter and Conservative Smoothing

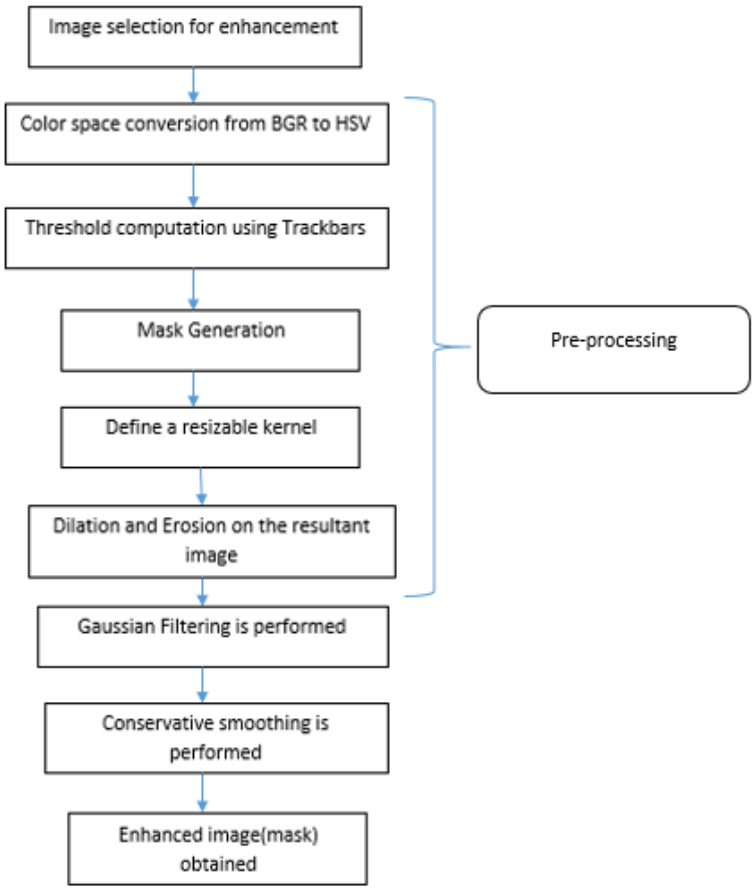


Figure 0: Flow chart for image denoising after the mask is obtained

As indicated by the flow chart shown above, the major steps involved in the method proposed are image selection for enhancement, pre-processing, Gaussian filtering, and finally, conservative smoothing after which the final image is obtained.

The flow chart is explained below:

- First the image is selected among many images present in the dataset.
- Then the pre-processing is started:
    - First the color space[37] of the image is changed from BGR to HSV[35].
    - After which the different thresholds for hue, saturation, and value are calculated using trackbars.
    - When the above step is done, a mask will be obtained.
    - A resizable kernel window is defined[38].
    - Morphological operations like dilation and erosion is performed on the resultant mask[36].
- Gaussian filtering is performed on the image[39][40].
- Conservative smoothing is performed[41].
- Final image is obtained which will have more characters that are recognizable.

After the pre-processing is completed, the image obtained will have some noise which will result in not all of the characters being recognizable, that's why the filtering is performed to reduce the noise in the image.

1.7. Evaluation Metrics

Evaluation metrics normally depends on the problem under consideration as well as the dataset being used to evaluate the algorithm. For example, the correctness of a digital image can be measured using many different parameters like PSNR, MSE, SSIM[42], Entropy[43]. The parameters used here in this paper are PSNR and Entropy, PSNR is calculated using the help of MSE.

Compression of image and video is aimed to yield the optimum visual quality based on the size of the compressed image file or bitrate of the encoded video sequence. For communications purposes, the quality of video may be distorted due to errors in transmission like packet losses. In order to compare the efficiency of several compression and error resilience mechanisms, reliable methods for assessing video quality are of prime importance. However, quality is a subjective feature that cannot be evaluated as a technical parameter, like packet loss rate or bitrate.

The images used here are very detailed so it's not possible to just visually see them and tell which one of the results is better among all that's why the use of some image parameter is encouraged. A basic flow chart of any image analysis algorithm is given in the figure below.

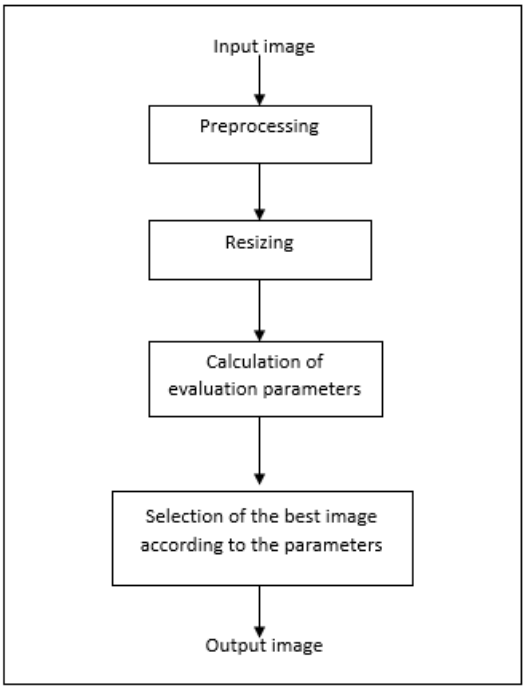


Figure 1.7: General Flow Diagram of Image Analysis

1.7.1. MSE

In the past few decades, the mean squared error (MSE)[44] has been a relevant quantitative performance metric in signal processing applications. It is the typical criterion for assessing the quality and fidelity of signal. It is chosen for comparing several signal processing applications and systems. It is ubiquitous preference of design engineers looking to enhance signal processing algorithms. MSE performs poorly while handling important signals such as images and speech.

MSE is the most preferred measurement metric of image quality. The values tending towards zero are considered to be better. It describes the Root- Mean-Square Deviation (RMSD) or Root-Mean- Square Error (RMSE)[42]. It is frequently termed as variance's

standard deviation. The term Mean Squared Deviation (MSD) is also used for MSE. The process of measuring an unobserved quantity of image is done using estimator. MSD gives an average value for the square of the errors.

The aim of MSE as a signal fidelity metric is to compare two signals by yielding a quantitative score that represents the degree of similarity/fidelity or, inversely, the level of error/distortion between them [45]. It is assumed that one of the signals is original while the other is distorted.

Consider that x = $\{x_i | i = 1, 2, \dots, N\}$ and $y = \{y_i | i = 1,2, \dots, N\}$ are two fixed-length, discrete signals (e.g., visual images), where N is the signal samples (pixels, if the signals are images) and x_i and y_i are the values of the $i\ th$ samples in x and y, respectively. The MSE between the signals is

$$\text{MSE}(\text{x}, \text{y}) = \frac{1}{N}\sum_{i=1}^{N}(x_i - y_i)^2 \qquad (1)$$

The error signal in MSE, $e_i = x_i - y_i$, is the difference between the actual and distorted signals. If one of the signals is an original signal of acceptable (or perhaps pristine) quality, and the other is a distorted version of it whose quality is being evaluated, then the MSE may also be regarded as a measure of signal quality. Of course, a more general form is the l_p norm

$$d_p(x, y) = \left(\sum_{i=1}^{N} |e_i|^p\right)^{1/p} \qquad (2)$$

In image processing applications, MSE is transformed into a PSNR which is computed as shown below:

$$PSNR = 10log_{10}\frac{L^2}{MSE} \qquad (3)$$

Here L is the dynamic range of acceptable pixel intensities of image. Consider an example of images that have allotments of 8 b/pixel of Gray-scale, $L = 2^8 - 1 = 255$. PSNR aids in images comparisons of different dynamic ranges. However, it has no new information relative to the MSE.

1.7.2. PSNR

PSNR [42] is the ratio of the maximum signal power to the distorting noise power. It impacts representation quality (in decibels). It is frequently evaluated using logarithm of decibel as the signals' dynamic range is quite broad. The range lies within the maximum and the minimum achievable values which are varied in accordance with their quality. PSNR is a technique for quality assessment. It measures the reconstruction quality of leaky image compression codecs. The actual data is the signal (assumed) while the noise acts as the error occurrence due to distortion or compression.

For video and image compression quality degradation, the value of PSNR for 16-bit data lies between 60 to 80 dB and for 8-bit data representation, 30 to 50 dB The PSNR is calculated using the equation shown below:

$$PSNR = 10\, log_{10}\,(peakval^2)/MSE \qquad (4)$$

The use of PSNR is not entirely preferred as a complete reference quality metric for images and video processed digitally. This is due to studies showing evidence of a poor correlation between subjective quality scores and their corresponding PSNR values. Jari et al.[46] depicts that the poor efficiency of PSNR is linked to a content based systematic shift of PSNR values. In cases of fixed distortion and content types that are common for visual communications purposes, PSNR may perform well even more efficiently in complex objective quality systems.

Till date, a majority of research focussed on digital image and video processing make use of PSNR as a measuring parameter for visual quality. The main advantage of PSNR is that it is simple and can be derived directly from the mean squared error (MSE) between the pixels in the original and the distorted image or video sequence. However, many studies have shown that PSNR frequently exhibits weak correlation with the mean opinion scores (MOS) yielded from subjective quality assessment [44][47]. PSNR is especially vulnerable to distortion types causing misalignment of pixels, such as spatial shift, rotation or resizing [44]. In order to overcome the drawbacks of PSNR, relevant efforts were made

to construct more reliable quality models based on the functioning of human visual system (HVS). However, objective image and video quality assessment is quite challenging and results in techniques statistically outperforming PSNR were not presented before the second phase evaluation study by video quality experts' group (VQEG) in 2003.

PSNR is widely used even after the presence of more precise metrics of quality evaluation. It approximates the relative perceptual quality efficiently when finite content and codec is concerned across several test cases[48]. The comparison of PSNR values attained from several content types is ambiguous, since the range of meaningful PSNR values varies heavily between contents. However, few studies have been conducted to explore the meaningful scope of use for PSNR.

The researches drawing comparison of various quality models work with integrated data sets covering all the sequences with different contents and distortion types. For such studies, PSNR usually performs poorly as compared to other highly advanced metrics. When the analysis is done separately for different contents, the circumstances are different. The data points for each content tend to form clusters and significantly better correlation between predicted and actual MOS can be obtained within each content separately than with the entire database.

PSNR is used to predict the perceived subjective quality nearly as well as more complex quality models representing the state-of-the-art. PSNR can be used as a quality metric for video, assuming that the analysis is performed separately for each content (source sequence), or the content-based bias in PSNR values is otherwise taken into account.

1.7.3. Entropy

Digital images can have a diverse range of distortions while acquisition, processing, storage, transmission and reproduction processes. This may lead in a reduction of visual quality. For applications in which images are ultimately to be viewed by human beings, the only "correct" method of quantifying visual image quality is through subjective evaluation. For practical applications, subjective evaluation is very inconvenient, time-

taking and costly. The goal of research in *objective* image quality assessment is to develop quantitative measures that can automatically predict perceived image quality. An objective image quality metric can play a variety of roles in image processing applications. First, it can be used to dynamically *monitor* and adjust image quality; second, it can be used to *optimize* algorithms and parameter settings of image processing systems; third, it can be used to *benchmark* image processing systems and algorithms[49].

Image similarity measure is significant for image compression purposes. During development of image compression models, the more efficiently an image similarity measure approximates the human visual system (HVS), the more accurate are the results of evaluation. The most frequently used measures applied to evaluate image similarity are peak signal-to-noise ratio (PSNR) and mean squared error (MSE). However, MSE and PSNR lack a crucial feature: the capability of assessing image similarity across distortion types [49][50]. Digital images distortions result in poor image quality. Image quality and similarity measures are approximations (of varying accuracy) of the human visual system. The opinions of human observers are the best benchmark for a similarity measure hoping to emulate the HVS.

Entropy is also known as average information of an image. It is employed to compute the randomness or disorder in the image and hence can be said to be a measure of uncertainty. It is used to estimate the how similar the original image is to the edited image. It can also be said to be a criterion for assessment for the ciphering algorithms, more entropy means better ciphering.

Image similarity measurement is significant in many applications of image processing. This assessment of similarity is linked to assessment of image quality. Here, the quality depends on the evident differences between a deteriorated image and the actual (unmodified) image. Accurate quality measurement is the key to automated evaluation of image compression systems. Recent algorithms for similarity measure are structural similarity (SSIM), mean squared error (MSE), and peak signal-to-noise ratio (PSNR). They have disadvantages namely, inconsistency, inaccuracy and large computational cost.

It is shown by Eric et al. [51] that the enhancement measure using entropy (EME) can be employed as a similarity measure for the image. Therefore, a measurement image quality is done. EME is employed in measuring the enhancement level making use of a given enhancement algorithm and enhancement parameter.

1.8. Contribution of this thesis

In this work the images are taken out of ancient palm leaf manuscripts. A pictorial representation of the basic steps is shown in the form of a flow diagram in figure 6. The major contributions of the work are summarized as follows:

- A new approach is introduced to extract the text from palm leaf manuscripts.
- A hybrid filtering technique is introduced to diminish the impact of noise in the text extracted manuscripts.
- Effects of hue, saturation and value are explored with respect to palm leaf manuscripts
- Different filtering techniques were explored and the technique giving the best balance for PSNR and Entropy is found out.
- PSNR and Entropy are calculated at different thresholds for every image

1.9. Organization of this thesis

The remaining part of the dissertation is arranged in the following manner: Chapter, describes the different color spaces and their properties and also which one is used in this dissertation and also why. Chapter 3 tells in detail about some of the morphological operations that can be performed on an image and their effects and which one of them conform to the direction in which this dissertation is being written. Chapter 4 discusses more about the different filtering algorithm and the type of noise they are used to remove and how are they used in the research performed. Chapter 5 gives the results of all the research and some discussion about the results. Chapter 6 provides the conclusion of this dissertation and also its Future scope which is then followed by the references in Chapter 7.

Chapter 2: COLOR SPACES

2.1. Introduction

The richness, diversity and significance of a typical color experience are exceptional. The ability of humans to perceive colors plays a major role in their perspective of beauty and aesthetic admiration of objects in daily life. Color is useful in the field of science, business, industry and medicine to code objects and recognise them as well as to communicate information. Additionally, color tends to be linked with several feelings, emotions, affective responses and moods namely likes, dislikes, depression or excitement.

For these reasons, it is not surprising to find that there has been a great deal of scientific work done on color vision and color perception. Despite all the work in this field, there are many fundamental things we still do not know about this subject. As one illustration, there is still much uncertainty about precisely how various wave lengths of radiant energy are transformed into nervous energy and differentially coded for transmission to the higher nerve centres of the brain.

Color can be studied anywhere along a broad spectrum of problems. At one extreme, some scientists are deeply interested in the micro-functioning of the retinal elements and in the photo-chemicals that transform electromagnetic energy into nervous energy. At the other extreme, one can find research workers concerned with responses to color as an aid to understanding the structure and dynamics of personality. [52]

Color is very atilt and discrete. It is very tedious to find the statistics of features of the brain's retaliation to the visual stimulus. Color spaces' makes it easy to for persons and machine codes to relate with colors. Color is defined as the brain's reaction to an exact visual stimulus. The measurement of spectral power allocation can help ascertain color description. This leads to high redundancy as the retina of the eye, makes use of three wide bands (basically associated with RGB luminosity) to sample the colors. There are two types of cells present in the eye's retina, perceptive cells(cones) and rods but these cells send only intensity signals individually to the brain. The combination of these signals

gives various sensations of the color like lightness, hue, brightness, chroma and saturation colourfulness.

A generalised scheme of colors is represented by the color space models. It is evident from the existing literature that in a 3D co-ordinate system every color is represented by a different color space transform. Each application has a different color space preference and optimization. Cameras and scanners can be used to collect relevant information from the RGB data for color spaces. The disadvantage of RGB color space lies in the representation of shading results and rapid changes in illumination. To overcome this limitation different color spaces were introduced.

Colours' representation should be numerical in a statistical methodology. This can play a key role in the future. Such statistical representation is acknowledged as the color model capable of grasping the color space. The main issue in color management is the color space conversion.

HSV color space is employed in applications like traffic sign recognition, face recognition, etc because colors are differentiated in HSV color space more easily than the RGB one. The RGB to HSV conversion is more computationally complex. A hardware is made to make up for the increased time consumption of color space conversion. This hardware helps better the performance and saves more power in the case of small-scale embedded devices.

Color space transformation is described as conversion of a color representation from one base to other. It is usually seen in context of an image transformation being represented in one color space to the other color space. The objective is to construct the transformed image as identical as possible to the original image.

2.2. Color Space dimensions

Colors can be identified without knowing the stimuli which instigate them. In the past years, artists, philosophers and scientists have devised several classifications of colors in accordance with their similarities and differences. All such techniques concur that a three-

dimensional model is required to depict a complete gamut of color sensations experienced by any normal person. A 3D diagram for the same is shown below [52].

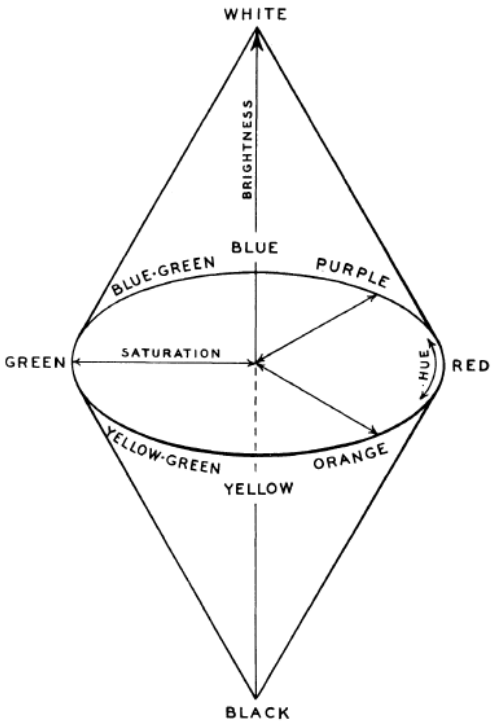


Figure 2.1: Schematic of psychological color space dimensions.

2.2.1. Hue

Hue is the most relevant of the three basic variables of color in terms of a mental phenomenon. It is the main quality factor in color. In other words, hue is the essential element which leads to color referencing by distinctive names like red, yellow, green, blue, violet, and so on. The occurrence of hue sensations is not in distinct groups rather they occur indistinguishable from each other and slowly complete the hue circle (refer to Figure 2.1). For example, beginning with red, one can describe color sensations which become yellower as they progress i.e., red initially forms orange-red, after which its orange, then yellow-orange and lastly yellow. Similarly, yellow has a continuous progression from yellow to green, blue, violet, and back to red.

Mean Hue Selections for the Strong Hues: The mean hues for all the strong colors are shown in Figure 2.2 as arrows radiating out from the center. The scale and the symbols around the inside of the circle show the locations of the 40 principal Munsell hues. The

short arcs on the outside of the circle are ranges for strong hues as defined by color experts of the Inter-Society Color Council and the National Bureau of Standards (Kelly and Judd, 1955).

The Munsell hues are spaced to correspond to equally noticeable color differences, that is, equal distances around the perimeter of the circle correspond to equally discriminable hue differences. Our findings show that we fell far short of this goal. There is a large region between most of the Muriseli greens and all the blue-greens that was not sampled at all by our color names. There is another large empty space between greenish blue and blue and still another between violet and purplish pink. These regions contain a great many colors which the eye can discriminate if it wants to. But apparently, we have not found these colors to be sufficiently interesting.

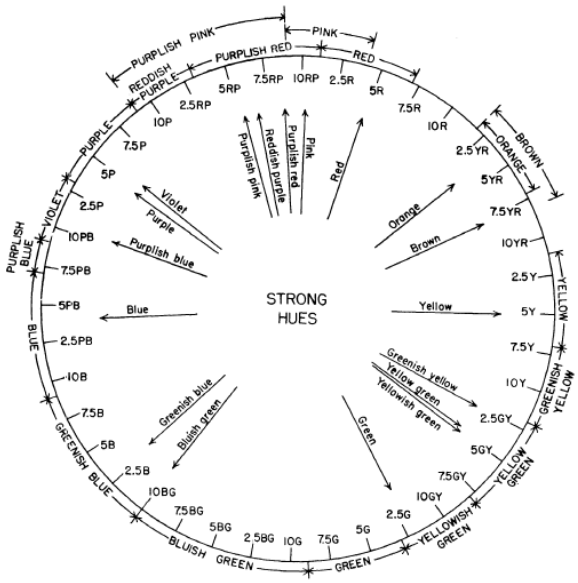


Figure 2.2: Average hue selections for the strong hues (arrows)

2.2.2. Brightness (lightness)

Brightness is the second variable of color sensation (its quantitative aspect). Two colors can have identical hue but differ in brightness. Light and dark is the common terminology used for representing variations in brightness. Similar to hue, the brightness dimension forms a spectrum, shading indistinguishably from very light to very dark hues. Brightness generally points to changes in light intensity.

2.2.3. Saturation

Saturation is the last/third parameter of color sensation and is the toughest to put in words alone without using actual color samples as points of reference. The percentage of pure hue present in a color is called Saturation. This lies approximately parallel to the concept of the purity of a chemical compound. Words like weak or strong, pale or deep reflect changes in saturation. The radii emanating from the centre of the diagram and expanding towards all directions is Saturation shown in figure 2.3. Here, black, white and Gray are the colors with zero saturation and have no hue. Therefore, the white-gray-black continuum changes only in brightness or lightness.

2.2.4. Color Space Defined

Now, when I say color space I mean that three-dimensional space which models color in all possible variations of hue, lightness, and saturation.

2.2.5. The Munsell Color System

Color sensations can be studied without reference to physical stimuli. Nevertheless, it is a fact that color sensations are most consistently and evoked by appropriate stimuli. The most useful general set of surface colors available for scientific and technical work is found in the Munsell Book of Color. This is a large collection of carefully prepared coloured chips, spaced to represent about equally noticeable color differences. Taken together, the Munsell chips cover virtually the full gamut of all colors reproducible by ordinary paints and pigments. The master colors of the Munsell system[52][53] have also been deposited with the National Bureau of Standards, where they have been measured and calibrated so that they can be related to other color measurement systems. In addition to illustrating the dimensions of color space, the Munsell system is extremely useful for specifying colors in business, science, and industry. If a manufacturer specifies the color of one of his products in Munsell terms, anyone else can go to a Munsell atlas and find out precisely what color he means. For all its advantages, the Munsell system is a collection of standardized colors useful primarily as a system of color nomenclature and specification for scientific and technical work. The language of the Munsell system is foreign and strange sounding to the uninitiated. The lightness dimension becomes value in Munsell terms, and saturation is called chroma. Chips bear names like 2.5 YR 4/6,

where the 2.5 YR refers to a particular hue, the 4 to a specific lightness (or value), and the 6 to a particular saturation (or chroma). One can easily find out with great precision exactly what such a color looks like, but the terminology of the Munsell system is hardly one that could be used for ordinary color descriptions.

Professor Albert H. Munsell developed the preliminary color perception organization into color spaces calling it the Munsell color model [54]. This was a basic self-reliant color space i.e., it was not dependent on the device. It was denoted as a three-dimensional cylinder, with the three dimensions equal to hue, saturation (color purity) and value (lightness). This model separated the three color components in regular, distinct and 3D space. Munsell color model was the one that introduces the concept of uniform spacing between the color spaces' components. The hue is depicted in the form of a circular segregated in ten sectors namely; red, yellow-red, yellow, green-yellow, green, blue-green, blue, purple-blue, purple and red-purple. This refers that the range of hue[55] is divided into eleven sections depicting darkness (black) at value zero and lightness (white) at value ten. This means the range is [0, 10] and perpendicular to the Munsell color model. Additionally, the chromate depicts the saturation of the corresponding selected combination of each of hue and value parameters and its range is [0, 12] as shown in figure below.

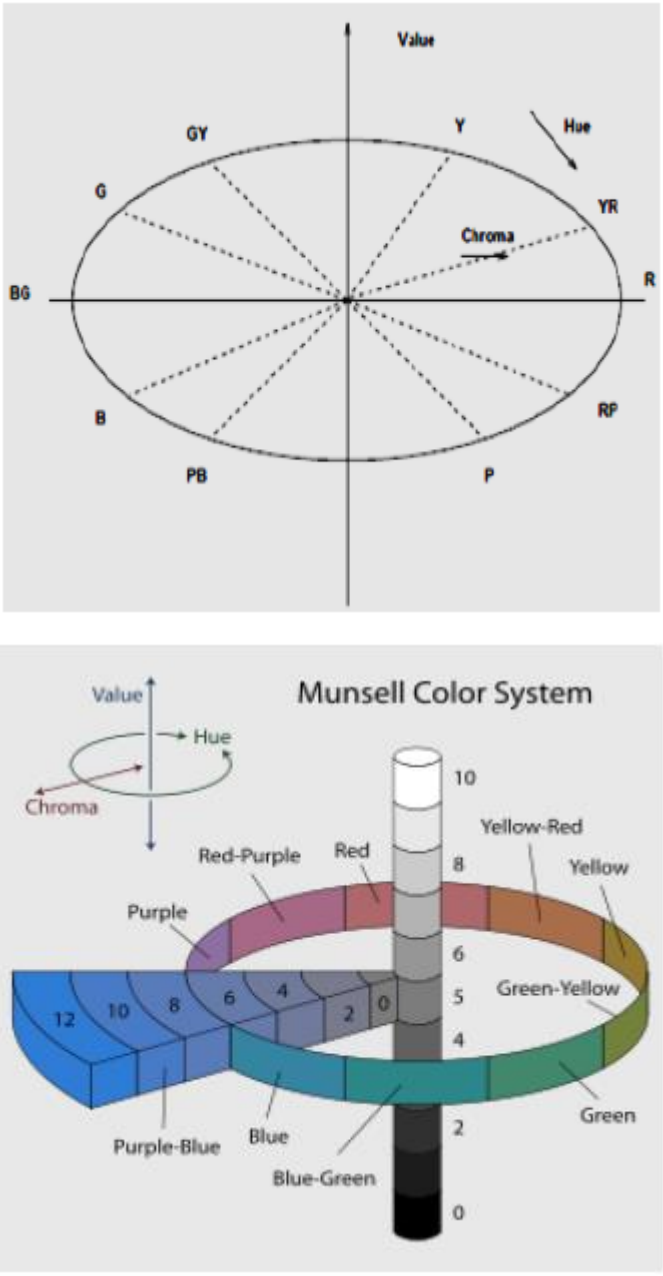


Figure 2.3: Visual representation of Munsell color system. Image at the bottom shows the hues circle at value 5, chroma 6; along the vertical V value from 0 to 10.

2.3. BGR Color Space

The default color space used by OpenCV is RGB but the values in it is stored as BGR format[56] where the order of the colors is reversed. Significance of 'Red' and 'Blue' will be reversed.

This model has been named after the three basic colors (red, green and blue). The colors in the light spectrum are the combination of these three colors only just in varying ratios, shown in Figure 2.3 (B)[57].

RGB color space reported in [57] can be described using a cube of normalized RGB color values in the range [0,1] with gray values lying on the primary diagonal of the black values (0,0,0) while the white values lie on the opposite corner (1,1,1) [57]. It is the fundamental color model for major image applications. This is due to the fact that no further transformation is needed for the image obtained before being displayed on the screen [57].

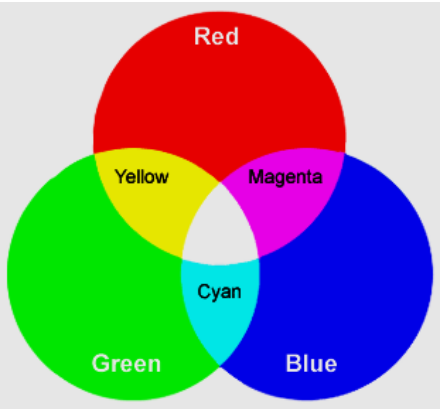
There are two types of RGB color model namely; linear and nonlinear which are explained below:

2.3.1. Linear RGB Color Space

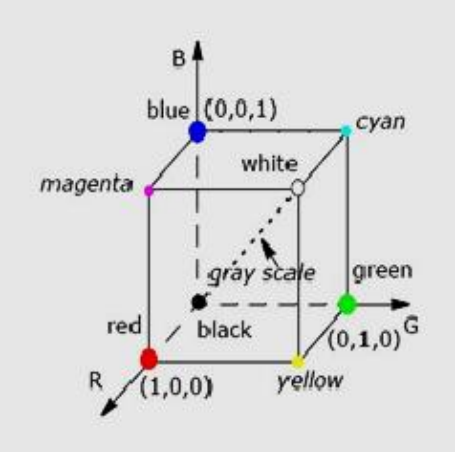
A system for color management is used so that the Linear RGB space can acquire color consistency[54]. It is incapable of performing numeric operations and is rarely utilized to represent an image. Its two main uses are applications using computer graphics and the liner to nonlinear mapping. This is achieved using γ(gamma correction factor) of the camera or any input system in the range [0,1] for both of the models[54].

2.3.2. Non-linear RGB Color Space

The input image captured through a camera or a scanner has information if the form of R'G'B' values in the range 0 to 255. Afterwards, this information is kept aside for application of image processing, MPEJ, JPEG standard. The transformation from linear to non-linear and vice-versa in the range [0,1] is defined in definition 1 and 2 respectively.



a)



b)

Figure 2.4: RGB color Mode. a): RGB representation. b): RGB cube.

Definition 1: Linear to Non-linear

$$R' = \begin{cases} 4.5\,R & if\ R \leq 0.018 \\ 1.099R^{\frac{1}{\gamma C}} - 0.099 & otherwise \end{cases} \quad (5)$$

$$G' = \begin{cases} 4.5\,G & if\ G \leq 0.018 \\ 1.099G^{\frac{1}{\gamma C}} - 0.099 & otherwise \end{cases} \quad (6)$$

$$B' = \begin{cases} 4.5\,B & if\ B \leq 0.018 \\ 1.099R^{\frac{1}{\gamma C}} - 0.099 & otherwise \end{cases} \quad (7)$$

End Definition 1

Definition 2: Non-linear to Linear

$$R = \begin{cases} \frac{R'}{4.5} & if\ R' \leq 0.018 \\ (\frac{R'+0.099}{1.099})^{\gamma D} & otherwise \end{cases} \quad (8)$$

$$G = \begin{cases} \frac{G'}{4.5} & if\ G' \leq 0.018 \\ (\frac{G'+0.099}{1.099})^{\gamma D} & otherwise \end{cases} \quad (9)$$

$$B = \begin{cases} \frac{B'}{4.5} & if\ B' \leq 0.018 \\ (\frac{B'+0.099}{1.099})^{\gamma D} & otherwise \end{cases} \quad (10)$$

2.4. HSV Color Space

The human visual system forms the basis of HSI, HSL, HSV color models [57]. The HSI family makes use of cylindrical coordinates for representing RGB points. HIS color model is significant in two main factors: first, the component I is isolated from the chrominance components saturation S and hue H and second, the dependence of these chrominance components on human perception of the spectrum of color.

HSV and HSL color spaces are alike other than the fact that in HSL high color values are allocated for colors that advance towards white color having a bounded saturation. This raises the model's complexity[58]. A single cone is used to depict a HSV color model, whereas a double is needed for HSV or HSI.

HSV is also known as hue, saturation value. In HSV color space is the color details stored in a cylindrical format of RGB color. HSV and HIS are employed in image analysis and computer vision for segmentation purposes [59].

Algorithm 1: Conversion from RGB to HSV

Input: RGB
Output: HSV
Method:

Step1: *[max and min values]*

$$M = max(R, G, B), m = min(R, G, B) \quad (11)$$

Step2: *[normalization in the range [0, 1]]*

$$r = (M - R)/(M - m) \quad (12)$$

$$g = (M - G)/(M - m) \quad (13)$$

$$b = (M - B)/(M - m) \quad (14)$$

Step3: *[V value]*

$$V = max(R, G, B) \quad (15)$$

Step4: *[S value]*

$$if\ M = 0\ then\ S = 0\ and\ H = 180\ degrees \quad (16)$$

$$if\ M <> 0\ then\ S = (M - m)\ /\ M \quad (17)$$

Step5: *[H value]*

$$if\ R = M\ then\ H = 60(b - g) \quad (18)$$

$$if\ G = M\ then\ H = 60(2 + r - b) \quad (19)$$

$$if\ B = M\ then\ H = 60(4 + g - r) \quad (20)$$

$$if\ H >= 360\ then\ H = H - 360 \quad (21)$$

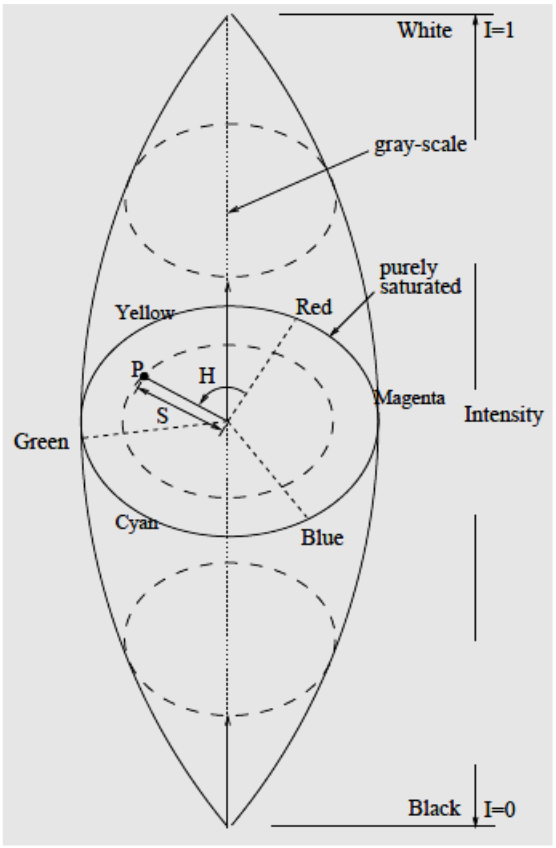
$$if\ H < 0\ then\ H = H + 360 \quad (22)$$

where H in the range [0,360], S and H in the range [0,100]

Step6: *[output]*

H, S, and V computed are the results of the algorithm.

End



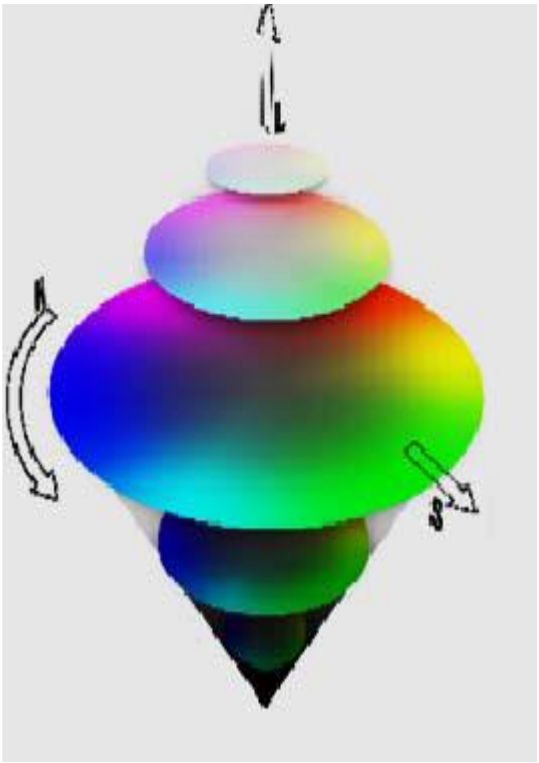


Figure 2.5: HSI color space different representation

Algorithm 2: Conversion from HSV to RGB [7]:

Input: HSV

Output: RGB

Method:

Step1*: [Hue value (H)]*

H value in the range 0 to 360 and divide by 60:

$$Hex = \frac{H}{60} \tag{23}$$

Step2: *[Compute primary and secondary color values]*

Primary and secondary color values namely, a, b, and c are calculated where primary color is the integer component of Hex

$$secondary\ color = Hex - primary\ color \tag{24}$$

$$a = (1 - S)V \tag{25}$$

$$b = \big(1 - (S * secondary\ color)\big)V \tag{26}$$

$$c = (1 - (S * (1 - secondary\ color)))V \tag{27}$$

Step3: *[RGB values calculations]*

When primary color =0 then

$$R = V, G = c, B = a \tag{28}$$

When primary color =1 then

$$R = b, G = V, B = a \tag{29}$$

When primary color =2 then

$$R = a, G = V, B = c \tag{30}$$

When primary color =3 then

$$R = a, G = b, B = V \tag{31}$$

When primary color =4 then

$$R = c, G = a, B = V \tag{32}$$

When primary color =5 then

$$R = V, G = a, B = b \tag{33}$$

Step4: *[output RGB]*

R, G, and B computed are the results of the algorithm.

End

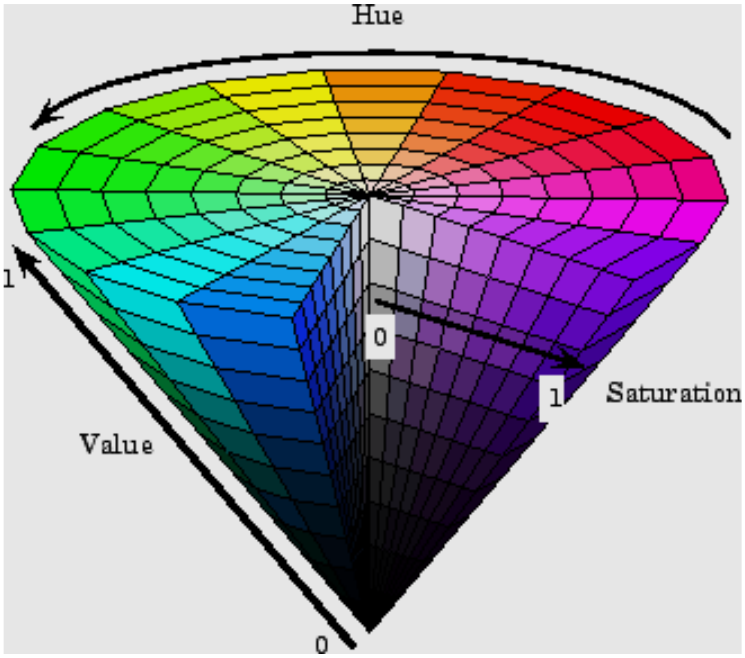


Figure 2.6: HSV color model single hex cone.

BGR to HSV conversion can also be done in the following way:

Calculations have been in the following manner[35]:

$$R' = \frac{R}{255};\ G' = \frac{G}{255};\ B' = \frac{B}{255} \tag{34}$$

$$C_{max} = MAX(R', G', B') \tag{35}$$

$$C_{min} = MIN(R', G', B') \tag{36}$$

$$\Delta = C_{max} - C_{min} \tag{37}$$

$$H = \begin{Bmatrix} 60^o \times \left(\frac{G' - B'}{\Delta} mod\ 6\right), C_{max} = R' \\ 60^o \times \left(\frac{B' - R'}{\Delta} + 2\right), C_{max} = G' \\ 60^o \times \left(\frac{R' - G'}{\Delta} + 4\right), C_{max} = B' \end{Bmatrix} \tag{38}$$

$$S = \begin{Bmatrix} 0, & C_{max} = 0 \\ \frac{\Delta}{C_{max}}, & C_{max} \neq 0 \end{Bmatrix} \tag{39}$$

$$V = C_{max} \tag{40}$$

CHAPTER 3: MORPHOLOGICAL IMAGE PROCESSING

3.1. Introduction

A branch of biology that covers the basics of forms and structures of plants and animals is called Morphology. Morphological image processing[60] outlines a span of image processing methods that handle the shape of features (i.e., morphology) in image. Morphological operations have the major goal of removing imperfections instigated while segmenting the images.

Image enhancement is achieved using mathematical morphological image processing. Generation of a better image and image's interest information are emphasized for the generation of a better image. Mathematical morphology is applied to the image employing several characteristics of the kernel/structuring element (SE) [38] for observation and comparison applications. The resulting output is acquired and proven to be correct for each step of the process in mathematical morphology. On the basis of the result, mathematical morphology can strengthen and improve the image. However, the morphological performance is linked to the properties of the kernel, which is selected according to the image's interest information[61].

The same is utilized here in accordance with mathematical morphology as a tool of extracting image components that are significant for denoting and describing of region shape like boundaries, skeletons etc. The goal of morphological operations include image data simplification, preserving significant shape characteristics and noise removal.[60]

3.2. Fundamental set of operations

Set theory is the base of the language used for mathematical morphology. Sets in mathematical morphology denote image objects. Consider A and B as two sets. When a is the index of a pixel in A, then it is written as $a \in A$. If a does not belong to A, it is written as $a \notin A$. If every element that is in A is also in B, then A is a subset of B, written, $A \subseteq B$. This is equivalent to the statement $a \in A \Rightarrow A \in B$.The union is the set represented by $C = A \cup B$ which is a collection of all elements that are in one or both sets. This can be rewritten as: $C = \{p | p \in A \; or \; p \in B(or \; both)\}$.

3.2.1. Union

The operation is

$$C = A \cup B \tag{41}$$

This operation yields a set containing elements of both A and B as shown in figure 3.1. The presence of some elements (red pixels) in both the sets does not alter the result. It is notable that there are two disjoint sets in just B[60].

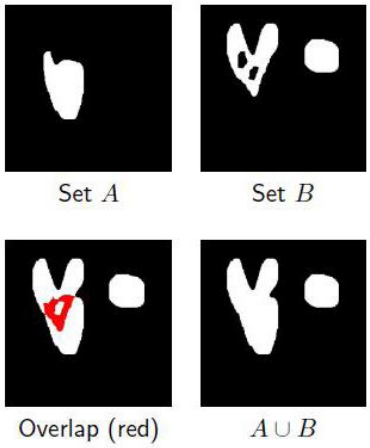


Figure 3.1: Visual representation of union operation

3.2.2. Intersection

The intersection of A and B is denoted by

$$D = A \cap B = \{p | p \in A \; and \; p \in B\} \tag{42}$$

If A and B do not have any common elements then it is called a disjoint set[60].

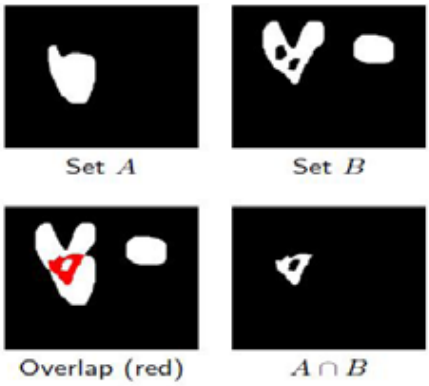


Figure 3.1: Visual representation of intersection operation

3.2.3. Complement

The elements not present in A gives the complement of A which is represented by A^C [60].

$$A^C = A \in Q_0$$

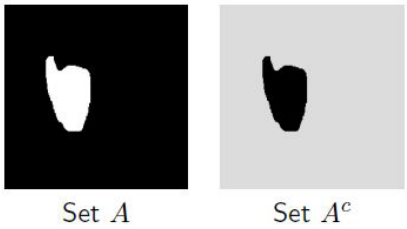


Figure 3.2: Visual representation of Complement operation

3.2.4. Reflection

The reflection of all the points is a standard morphological operation in a set (about the origin of the set). The origin of the base is not necessarily the origin of the set. The figure 3.4 is an image while its reflection about a point is shown in red. The original image is seen in green and the reflected one in white [60].

Figure 3.3: Visual representation of reflection operation

3.3. Morphological Operations

The most basic morphological operations are Dilation and erosion[36][60]. Elementary set operations are used to define these two operations and then they are used as basic component in various algorithms. They are generated when the kernel and the set of pixels of interest in the image interact with each other. The kernel has both a shape and an origin.

3.3.1. Structuring Element (Kernel)

A small numbers' matrix utilised for image convolutions is defined as a kernel[38]. Different dimensions of the matrix of the kernel containing different combinations of numbers results in different outputs under convolution.[61]

For example: A 3 × 3 kernel is shown below

1	1	1
1	1	1
1	1	1

Set of coordinate points =

{ (-1, -1), (0, -1), (1, -1),
(-1, 0), (0, 0), (1, 0),
(-1, 1), (0, 1), (1, 1) }

Figure 3.5: An example of kernel matrix

A 'kernel' is also known as a 'structuring element' which is an indistinguishable object employed in 'mathematical morphology'. However, the structuring element also has a fixed origin.

3.3.2. Erosion

Assume that A is a set of pixels and B is the kernel. Then erosion is given by the equation:

$$A \ominus B = \{s|(B)s \subseteq A\} \qquad (43)$$

Example of Erosion

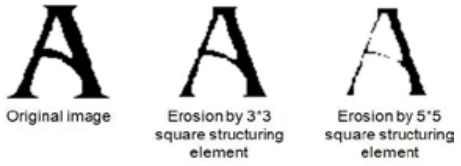


Figure 3.6: Visual representation of erosion

Erosion can separate joint objects and remove extrusions. It is utilized in thinning/shrinking operation while dilation is used for thickening operations and it can also grow boundaries in a binary image.

The main idea behind erosion is to erode the boundaries of the foreground object (foreground should be kept white). This is achieved by the kernel sliding over the boundary of the object and the pixels under it are observed if all the pixels are '1' then it is kept as it is otherwise it is eroded which results in the boundary pixels getting discarded according to the size of the kernel.

3.3.3. Dilation

Consider A to be a set of pixels and B as a kernel. Assume $(\hat{B})$s be the reflection of B about its origin and followed by a shift by s. Then, dilation, which is denoted by A $\oplus$ B, is the set of all shifts that satisfy the following:

$$A \oplus B = \{s|(\hat{B})s \cap A\} \quad (44)$$

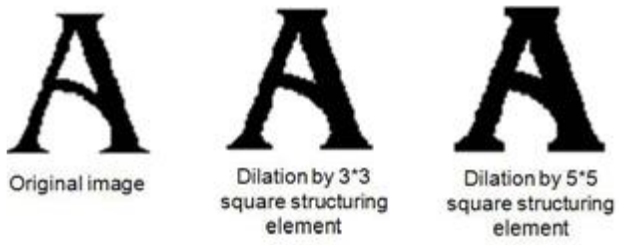


Figure 3.7: Visual representation of dilation

The equation above is built on $(\hat{B})$s. Then dilation between A, B is then a set of all displacement. And it is assumed that B is the kernel and A is the group of pixels to be dilated or we can say that the B is rotating about its origin and slides over A. Dilation is the dual of erosion used for repairing breaks and intrusions. Here if even one pixel under the kernel is '1' then all the pixels are considered as '1' and thus it expands the white region in the image.

3.3.4. Amalgamation of Erosion and Dilation

If one wants to keep just the largest square in an image, then it can be achieved with the help of erosion. By keeping the kernel smaller than the size of the largest square all of the other squares will be eliminated. And if one wants to restore the lost data then it can be done by dilation, using the same kernel as before. [60]

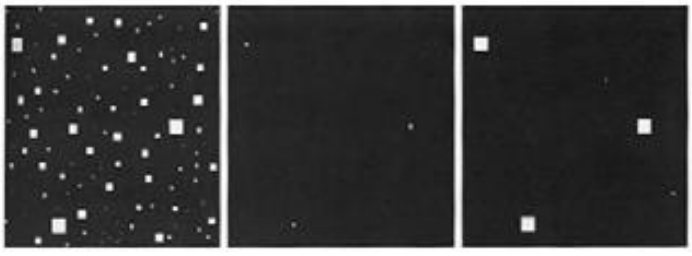


Figure.3.8: This is what happens after dilation and erosion

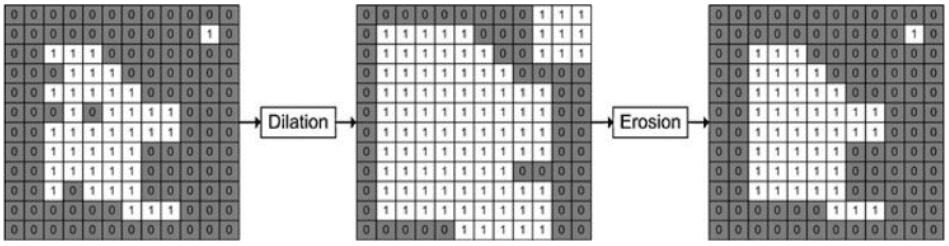


Figure 3.9: Visual representation of erosion after dilation

3.3.5. Opening and Closing

The opening operation smoothens the contour object, eliminates thin protrusions and breaks narrow isthmuses. Closing also seems to smooth sections of contours however, it fuses narrow breaks and long thin gulfs, fills gaps in the contour and eliminates small holes (as opposed to opening)[60][61].

Equation for opening: $A \circ B = (A \ominus B) \oplus B$ (45)

Equation for closing: $A \bullet B = (A \oplus B) \ominus B$ (46)

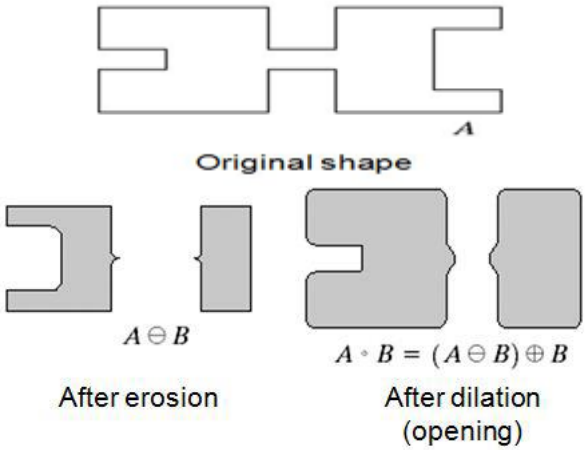


Figure 3.10: Example of opening

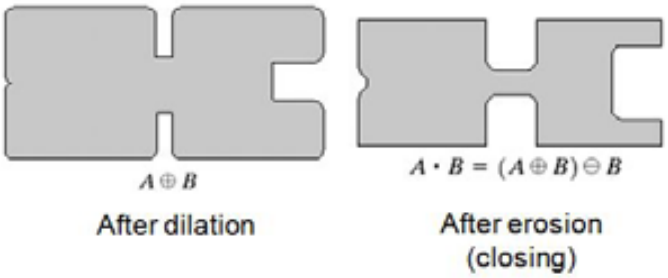


Figure 3.11: Example of closing

3.3.6. The Hit-or-Miss Transformation

This transform is used as a fundamental tool for recognizing shapes. It is used to find the specified shape in the image. It is done using the equation shown below.

$$A \circledast B = (A \ominus X) \cap [A^C \ominus (W - X)] \qquad (47)$$

Here W is a small window which is considered to have a thickness of at least one-pixel than an object. It can also be used to detect patterns in an image where the background does not provide any details to the shape[60].

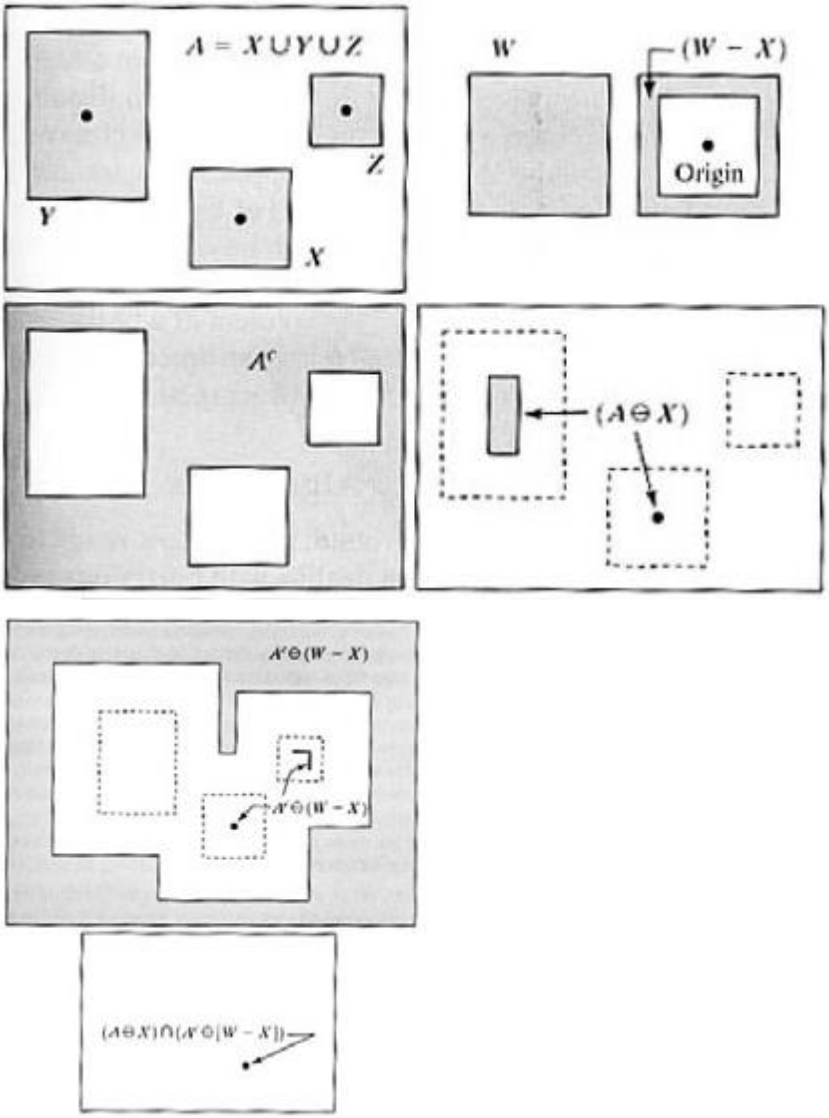


Figure 3.12: Example of Hit-or-mass transformation

CHAPTER 4: FILTERING

4.1. Introduction

The cost of computers and digital cameras are being lowered each year because of the fast pace of technology of today's world which makes them more pocket friendly. Which in turn leads to these digital cameras becoming a household item. The images captured from these digital cameras are much more detailed, giving out more information about the captured subject with having less complexity for extracting that information. That's why digital images are being used more prominently in many researches.

Very often the system for which the image captured by an imaging system or a camera is not able to process that image directly. This can happen due to the corruption of the image because of change in illumination, arbitrary variation of intensity etc. which must be dealt the stage of vision processing. [40]

Usually color-value or a matrix of grey-level is used to encode a digital image. In the case of a video, the matrix becomes three dimensioned, where the third dimension corresponds to time. A pixel or "picture element" is depicted using (i, $u(i)$) pair, where $u(i)$ gives the value at i". For gray level images as they are in two dimensions only so, $u(i)$ will be a real value and i will represent a point on the two-dimensional (2D) grid. For conventional color images, $u(i)$ represents a combination of R, G, B value components. Which will also apply to 3D images, movies etc.

Blur and noise are the two principal shortcomings in image accuracy. Blur will always be present in image acquisition systems because of the Shannon–Nyquist sampling conditions that the digital images have to satisfy. The second main concern for the image is noise.

Value of each pixel $u(i)$ is given by the combination of every measurement done by a CCD matrix with a light focusing system. In each CCD's captor the count with respect to the obturation time is done by squaring the number of incoming photons. According to central limit theorem due to the light source being the same, the amount of photons received by each pixel fluctuates around the mean. It can also be said that all of the

fluctuations will be of the of order $\sqrt{n}$ for n incoming photons. Additionally, if the captor is not properly cooled it will receive fake heat photons. This will result in a disturbance called “obscurity noise”, which can be roughly represented as:

$$v(i) = u(i) + n(i) \qquad (48)$$

where $i \in I$, $v(i)$ is the value observed, $u(i)$ gives the “true” value at pixel i, and $n(i)$ is the noise. It is evident that the noise is dependent on signal; which means that, $n(i)$ will be greater when $u(i)$ is bigger. In noise models, the normalized values of $n(i)$ and $n(j)$ at different pixels are assumed to be independent random variables, and one talks about “white noise.”

There are many reasons for an image to get contaminated like transmission channel, environment and some other factors also like noise introduced during compression, acquisition, transferring etc. which leads to image distortion and information loss. Due to the noise present in the image the following operations like image and video processing, image analysis to be performed on the image becomes difficult. This is the major reason that image denoising plays a pivotal role in image analysis system.

The process of removing noise from an image in called Image denoising and is done to restore the true image. But as edge, texture and noise, all of them have high frequency which is why it is difficult to differentiate between them in the denoising process which makes the process even harder and ultimately leads to losing some details. This makes the removal of noise, to obtains good images after noise removal with losing minimum information, an important problem nowadays. [62]

Nowadays digital image processing technology is on rise and its main area of research is denoising. Digital image is very vulnerable to noise during transmission and acquisition which can occur due to channel-bit errors or photo-electric sensors faults. Because of this reason many imaging applications nowadays consists of noisy images. The reason for the corruption of TV images is imperfections in the reception of the image and atmospheric interference. Scanning the damaged surfaces of digital artwork can also introduce noise in them. Noise can also be introduced in the images during image transmission which can

be due to many reasons like switching transients in power lines, ignition system of a car, etc. Daily life applications like computer tomography, satellite television and research areas like astronomy, both are heavily dependent on digital images.

For any signal processing system, filtering is a very important step, which is described as predicting the degraded signal mostly by add-on arbitrary noise. Linear processing technique have been the choice of many filtering algorithms for many years because of its simplicity. But for almost all of the filtering algorithms A gaussian model is taken for and then the parameters are optimized for the system model. Recently nonlinear methods are used more and more because of their ability to minimize Gaussian noise which retains the important signal elements like fine details and edges and also helps to remove deteriorations caused while transmission through non-linear channels or signal formation. Fuzzy-logic techniques are important in non-linear algorithms because of their ability to reason uncertain and vague information. [41]

There are many ways to preserve the original image in image processing; image enhancement is one of them. It can be defined as a method of improving the quality of the image such that it looks inherently better. The resulting image after enhancement is observed to provide more details or having less noise or having a better contrast[63].

There are many ways to perform image enhancement like removing interference and additive noise, increasing the contrast, removing multiplicative interference, also minimizing the blurring. Then the image filtering process determines the best outcome among several filters.

4.2. Description of the problem of Image denoising

Image denoising equation can be represented as[62]:

$$y = x + n \tag{49}$$

y = noise in the observed image; x = clean image, and n is the additive white Gaussian noise (AWGN) having standard deviation σn. Various methods can be used for its estimation like block-based estimation[64], median absolute deviation[65], and principle

component analysis (PCA)- based methods[66]. Natural images' noise is reduced by denoising meanwhile also improving the signal-to-noise ratio (SNR) and reducing the loss of original features. Challenges faced while image denoising are:

- Smooth flat areas,
- Edges should not be blurred,
- Image texture should remain same, and
- New information should not be generated.

Through Eq. (1) it is not possible to recover a clean image x because the model being operated on is a noisy one. This is the reason that the image denoising has been studied extensively for past several years is so that a good estimation image $\hat{x}$ can be obtained.

4.3. Mean Filtering

In linear filters, mean filter is one of the easiest to implement. A local mean operation is performed, and each pixel's value is substituted by the mean of the local neighbourhood pixel values:

$$h[i,j] = \frac{1}{M}\sum_{(k,l)\epsilon N} f[k,l] \qquad (50)$$

M = total pixels in the neighbourhood N[40][39]. For example, for a 3 x 3 neighbourhood about [i, j] gives:

$$h[i,j] = \frac{1}{9}\sum_{k=i}^{i+1}\sum_{l=j-1}^{j+1} f[k,l] \qquad (51)$$

Mean filter is of many types as discussed below:

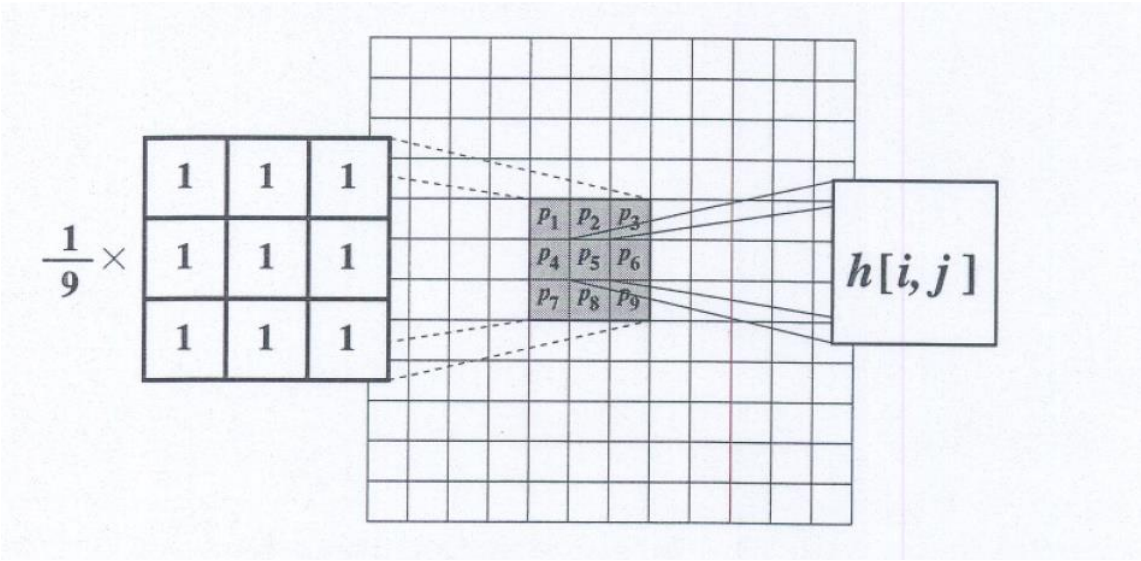


Figure 4.1: Mean filtering for a 3 x 3 neighbourhood[40].

4.3.1. Arithmetic mean filter

It is the easiest to implement mean filter. Assume that S_{xv} being centered at point (x, y) gives the coordinates set in the shape of a rectangle of size mXn which will be a sub-image window. The algorithm calculates the mean value of the deformed image $g(x, y)$ in the area described by S_{xy}. The restored image's value at any point (x, y) is just the arithmetic average calculated using the using the region defined pixels by S. In other words.

$$\hat{f}(x, y) = \frac{1}{mn} \sum_{(s,t)\epsilon S_{xy}} g(s, t) \qquad (52)$$

This process can also be accomplished using a convolution mask where all the coefficients have value $1/mn$. Local variations of the image are smoothed by a mean filter. Blurring causes the noise removal[67].

4.3.2. Geometric mean filter

Its expression is given by:

$$\hat{f}(x, y) = [\prod_{(s,t)\epsilon S_{xv}} g(s, t)]^{\frac{1}{mn}} \qquad (53)$$

The sub-image window's pixels product when raised to the power of $1/mn$ gives the value of restored pixels. Better smoothing can be performed using the geometric mean filter than the arithmetic mean filter but it compromises the image details in the process [67].

4.3.3. Harmonic mean filter

Its expression is given by:

$$\hat{f}(x,y) = \frac{\sum_{(s,t)\epsilon S_{xy}} g(s,t)^{Q+1}}{\sum_{(s,t)\epsilon S_{xy}} g(s,t)^{Q}} \qquad (54)$$

Q = filter's order. The best filter for minimizing or almost eliminating the impulse noise. For positive and negative values of Q it eliminates pepper and salt noise respectively but both operations cannot be performed simultaneously[67].

4.4. Median Filtering

The median filter is a digital filter of a non-linear kind and it operates by replacing the pixels in the image with the median value of the neighbourhood pixels. Sharp signal changes can be preserved by the median filter. The median filter can be used to remove impulse noise (salt and pepper noise) because the gray level of the impulse noise is either higher or lower than the neighbourhood pixels.

$$y[m,n] = median\{x[i,j], (i,j)\epsilon w\} \qquad (55)$$

where "w" is a neighbourhood site on the center around the location [m, n] in the image data[39][40][68].

Median filter is the best order-statistics filter. And it works by replacing the value of a pixel by the median value of the pixels in its neighborhood.

The calculation of the median while considering the original value of the pixel. As Median filters provide excellent noise reduction and less blurring than the linear filters, they are somewhat famous. When both bipolar and unipolar impulse noise are present, Median filters works excellently.

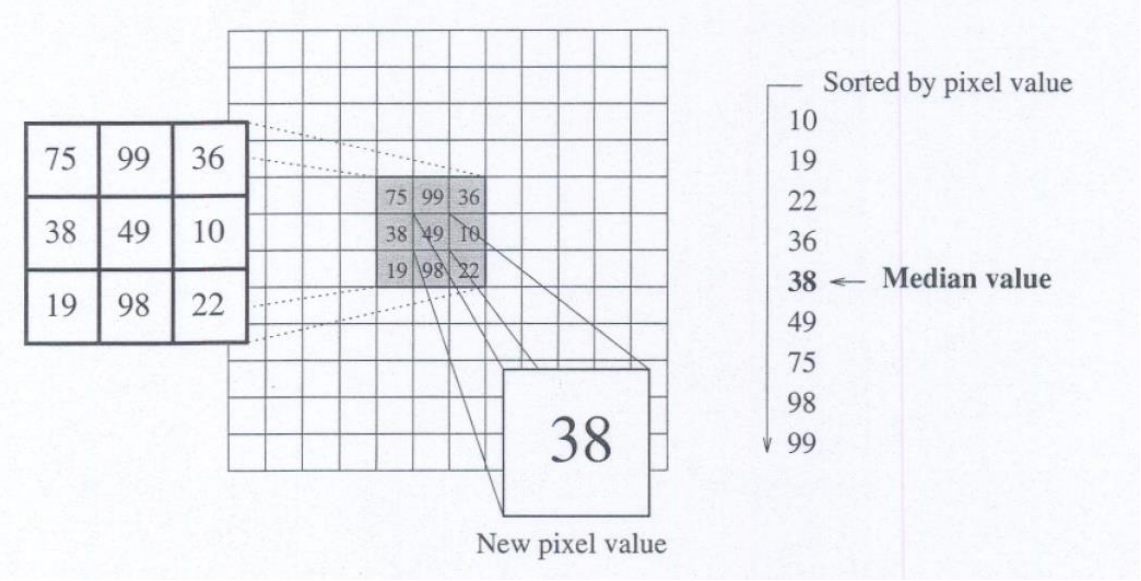


Figure 4.2: Median filtering for a 3 x 3 neighbourhood[40].

4.5. Gaussian Filter

Gaussian filter is a liner filter. Gaussian filter or Gaussian smoothing is a filter that causes the image to blur. A gaussian filter can be used to reduce the image noise and detail simultaneously maintaining the color. The formula is given below.

$$G(x,y) = \frac{1}{2\pi\sigma^2} e - \frac{x^2+y^2}{2\sigma^2} \qquad (56)$$

where, 'x = horizontal axis' origin distance and 'y' vertical axis' origin distance, and 'σ' is the Gaussian distributions' standard deviation[40][63].

In a Gaussian filters the weights are selected based on the shape of the Gaussian function. Normal distribution noise is removed using Gaussian smoothing.

There are five properties of Gaussian functions due to which they are considered very useful in early vision/image processing. These properties indicate that the Gaussian smoothing filters are effective low-pass filters from the perspective of both the spatial and frequency domains, are efficient to implement, and can be used effectively by engineers in practical vision applications.

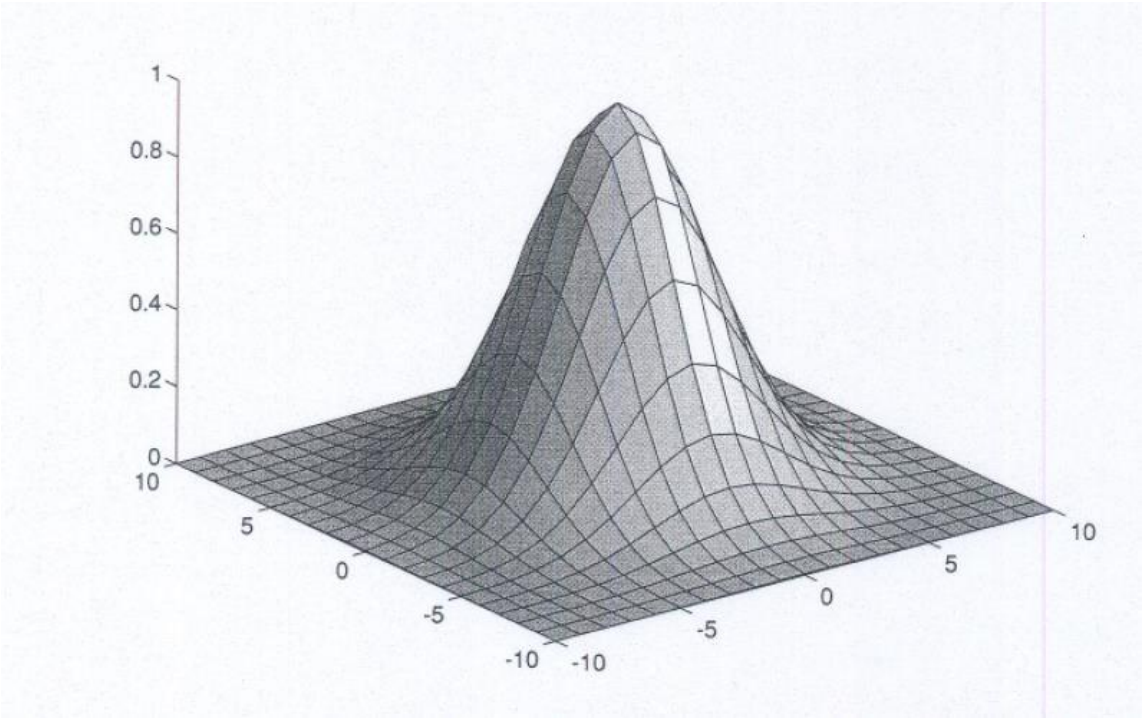


Figure 4.3: The two-dimensional Gaussian function with zero mean[40].

4.6. Conservative Smoothing

Conservative smoothing[41] is a technique based on noise reduction. It is a straightforward fast filtering method that gives up noise suppression power to conserve the high spatial frequency detail (e.g., sharp edges) in an image. It is made in such a way that it removes the noise spikes i.e., the pixels with unusually low or high intensities (salt and pepper noise) and thus is somewhat less efficient for removing additive noise present in an image (Gaussian noise). The fact that noise has a high spatial frequency and can be attenuated with local operation is exploited by conservative smoothing. Thus, each pixel's intensity is approximately consistent with those of its nearest neighbours. Mean filtering accomplishes this by averaging local intensities and median filtering does the same by a non-linear rank selection technique. On the other hand, conservative smoothing makes sure that each pixel's intensity is within the range of intensities defined by its neighbours.

CHAPTER 5: DISCUSSION AND RESULTS

This section highlights the efficiency of the method proposed.

The tables given below gives the parameters of the original image for comparison

Table 5.1: Comparison between the two methods based on PSNR

Image	Entropy
Original	6.603
After pre-processing	1.096
Proposed method	1.534

Table 5.2: Comparison between the two methods based on Entropy

Image	PSNR
Original Image, Image after pre-processing	28.153
Original Image, Proposed method Image	28.031

5.1. BGR to HSV based text extraction from manuscripts using slidebars

The method is compared with the existing NGCM method in order to depict the similarity between the enhanced image results. The proposed technique is simpler as compared to NGCM with almost the same or even better output in some cases.

NGCM algorithm was mainly developed to extract text from the real time images as in for the video stream or the webcam. Therefore, the same images have been used as a parameter for comparison between BGR to HSV based text extraction and NGCM. It is evident visually that the proposed method produces similar results to NGCM. Additionally, the BGR to HSV based text extraction gives four images stacked with each other. Images are stacked in the fashion of the figure given below.

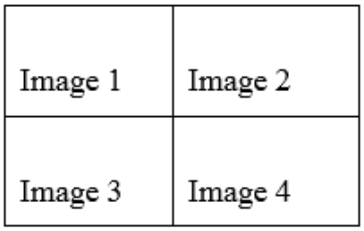


Figure 5.1: Mannerism of stacked imaged in the method

Every stacked image represents a different result described as follows:

- Image 1 displays the original mask image generated using the cv2.inRange() function after being converted to BGR color space.
- Image 2 displays the bitwise AND operation of the image with itself and the original mask generated.
- Image 3 displays the morphological transformed image (Dilated and eroded image)
- Image 4 displays the addition of two images which are the binary thresholded image and the Otsu thresholded image.

The best image among the four can be then selected as a result of text extraction.

Figure 5.2: a) The original Image, b) Image output with NGCM, c) Image output with BGR to HSV text extraction using slidebars

The dataset contains images of the manuscripts written on the palm leaves as showed in figure 6. Since these manuscripts are ancient, most of the leaves have deteriorated due to environmental effects. Contemporarily if researchers try to interpret the text written on these manuscripts, then it would be difficult due to degradation. Therefore, we apply text extraction methods to recover the writings.

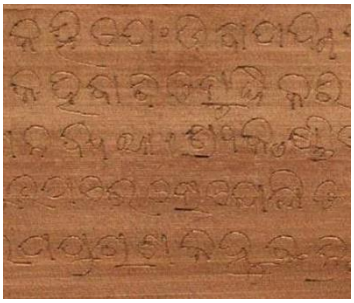


Figure 5.3: A small part of an image from the dataset

The figure 7, 8 and 9 shown below depicts the text extracted from the manuscript image using BGR to HSV based text extraction incorporating slidebars at different thresholds.

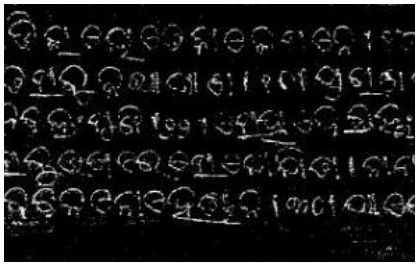


Figure 5.4: A small part of the output image after going through the method proposed (threshold = 244)

Figure 5.5: A small part of the output image after going through the method proposed (threshold = 200)

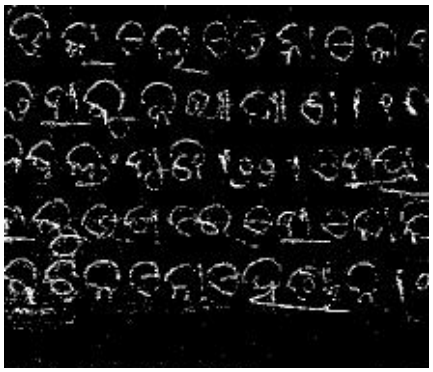


Figure 5.6: A small part of the output image after going through the method proposed (threshold = 150)

Table 5.3: Entropy of the same image corresponding to different thresholds

Hue (0 - 179)		Saturation (0 - 255)		Val (0 - 255)		Threshold	Entropy
Min	**Max**	**Min**	**Max**	**Min**	**Max**		
Thresholded Image						**-**	**7.39(Original Image)**
0	29	0	255	123	154	15	4.76
0	18	115	255	48	145	50	3.83
0	20	109	219	56	149	75	4.22
0	21	86	240	66	149	100	4.21
0	27	100	217	69	153	150	4.64
0	89	0	255	0	147	200	4.09
0	19	0	254	52	152	244	4.57

Table 5.4: PSNR of the same image with different thresholds

img 1	img 2 (threshold)	PSNR(img 1 – img 2)
Original Image	15	27.84
Original Image	50	27.85
Original Image	75	27.85
Original Image	100	27.85
Original Image	150	27.85
Original Image	200	27.84
Original Image	244	27.85

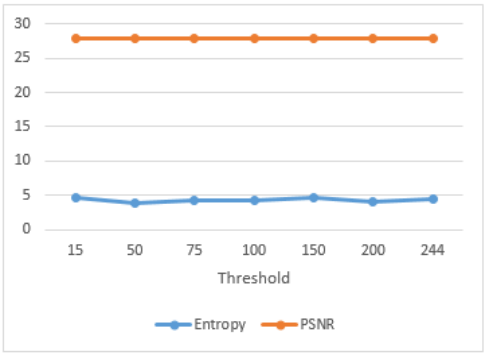


Figure 5.7: Resultant image's Entropy vs PSNR

As it can be seen in Table 5.3, Entropy of the thresholded images is less than the entropy of the original image given in Table 5.1 which indicates that there will be some information loss or it can also be said that all of the characters in the manuscripts will not be recognized and will be left out for which we took our image on different thresholds and then calculated every image's entropy after which the image with the highest entropy is accepted as result even if it's a very less difference in entropy it will e]mean at-least one more recognized character.

5.2. Denoising of Palm leaf manuscripts using Gaussian filter and Conservative Smoothing

The dataset contains images of the manuscripts written on the palm leaves as shown in figure 5.8. Since these manuscripts are ancient, most of the leaves have deteriorated due to environmental effects. Contemporarily if researchers try to interpret the text written on these manuscripts, then it would be difficult due to degradation. Therefore, the filtering techniques are applied to the processed images to interpret them more easily.

Table 5.5: Filtering techniques comparison on image with threshold = 244

Type of filter	PSNR	Entropy
Mean Filter	27.80	6.89
Median Filter	27.86	6.85
Gaussian Filter	27.92	6.51
Conservative Smoothing	28.01	4.57
Gaussian Filter + Conservative Smoothing	27.97	6.61

Table 5.5 shows the result of various filtering algorithms when applied to the image obtained after pre-processing. The results of Gaussian filtering, Conservative smoothing and Gaussian filtering + conservative smoothing is given in Figure 5.10, Figure 5.11, Figure 5.12 respectively. It can be seen that while Mean filtering gives the best performance concerning Entropy[43] and Conservative Smoothing for PSNR[42] but collectively they don't do very good so a new method is tried out which is the combination of Gaussian Filtering and Conservative Smoothing which results in overall good results with a little increase in PSNR and a good improvement in Entropy which will result in a good final image.

The combination of Gaussian filtering and conservative smoothing has the most balanced results when compared to the PSNR and entropy of the original image.

Figure 5.8: An image from the dataset

Figure 5.9 shown below depicts the text extracted from the manuscript image using BGR to HSV based text extraction incorporating slidebars at threshold = 244.

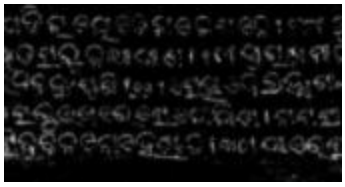


Figure 5.9: A small part of the output image after going through the method proposed (threshold = 244)

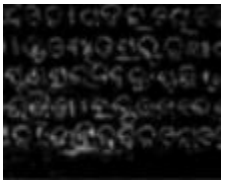


Figure 5.10: Gaussian filtering performed of the selected image

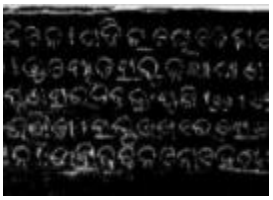


Figure 5.11: Conservative smoothing performed on the selected image

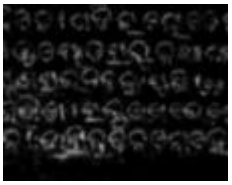


Figure 5.12: Gaussian Filtering + Conservative Smoothing performed on the selected image

CHAPTER 6: CONCLUSION AND FUTURE SCOPE OF WORK

In this work, text extraction from the manuscripts is done using BGR to HSV based text extraction. The manuscript dataset was inherently difficult to decipher. This was due to same color of leaves and the text inscribed and damage over the years. The proposed method makes the text of the resultant image legible. This helps in easy interpretation of the text.

The method proposed here can be used for image noise removal and it also gives out good results when comparing to the algorithms performing individually. The method gives a balanced result focusing on both the measurement parameter, unlike some others which just gives a good result in one of the parameters and poor in another.

Further research can be done to improve the image quality, even more, using various hybrid techniques and thus gather relevant information from ancient scripts. The output image can be further enhanced in quality using various methods. This algorithm can be automated using deep learning or any other relevant method.

The algorithm can be further modified such that variation of hue in the image does not cause any changes to the image because it can be seen in the results that hue does not have a very significant impact on the results.

CHAPTER 7: REFERENCES

[1] R. Mithe, S. Indalkar, and N. Divekar, “Optical Character Recognition,” no. 1, pp. 72–75, 2013.

[2] G. G. Chowdhury, “Natural language Processing,” pp. 51–89.

[3] N. Sharma and Nidhi, “Text Extraction from Images: A Review,” *Lect. Notes Networks Syst.*, vol. 10, pp. 153–165, 2018, doi: 10.1007/978-981-10-3920-1_16.

[4] F. D. Julca-Aguilar, A. L. L. M. Maia, and N. S. T. Hirata, “Text/Non-Text Classification of Connected Components in Document Images,” *Proc. - 30th Conf. Graph. Patterns Images, SIBGRAPI 2017*, pp. 450–455, 2017, doi: 10.1109/SIBGRAPI.2017.66.

[5] “Image segmentation algorithm by piecewise.pdf.” .

[6] A. Chourasiya and N. Khare, “A Comprehensive Review Of Image Enhancement Techniques,” *International Journal of Innovative Research and Growth*, vol. 8, no. 6. 2019, doi: 10.26671/ijirg.2019.6.8.101.

[7] M. pietikaine. J.Sauvola, “Adaptive document image binarization.pdf.” .

[8] E.-S. Jung, H. Son, K. Oh, Y. Yun, S. Kwon, and M. S. Kim, “DUET: Detection Utilizing Enhancement for Text in Scanned or Captured Documents,” pp. 5466–5473, 2021, doi: 10.1109/icpr48806.2021.9412928.

[9] S. Lawrence, C. L. Giles, and K. Bollacker, “Digital libraries and autonomous citation indexing,” *Computer (Long. Beach. Calif).*, vol. 32, no. 6, pp. 67–71, 1999, doi: 10.1109/2.769447.

[10] H. Han, C. L. Giles, E. Manavoglu, H. Zha, Z. Zhang, and E. A. Fox, “Automatic document metadata extraction using support vector machines,” *Proc. ACM/IEEE Jt. Conf. Digit. Libr.*, vol. 2003-Janua, pp. 37–48, 2003, doi: 10.1109/JCDL.2003.1204842.

[11] V. K. Vijayan, K. R. Bindu, and L. Parameswaran, “A comprehensive study of text classification algorithms,” *2017 Int. Conf. Adv. Comput. Commun. Informatics, ICACCI 2017*, vol. 2017-Janua, pp. 1109–1113, 2017, doi: 10.1109/ICACCI.2017.8125990.

[12] P. C. Sen, M. Hajra, and M. Ghosh, *Supervised Classification Algorithms in Machine Learning: A Survey and Review*, vol. 937. Springer Singapore, 2020.

[13] A. Sharma, G. Hua, Z. Liu, and Z. Zhang, “Meta-tag propagation by co-training an ensemble classifier for improving image search relevance,” *2008 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work. CVPR Work.*, 2008, doi: 10.1109/CVPRW.2008.4562952.

[14] N. Ducheneaut and V. Bellotti, “E-mail as habitat,” *Interactions*, vol. 8, no. 5, pp. 30–38, 2001, doi: 10.1145/382899.383305.

[15] M. Allahyari *et al.*, “A Brief Survey of Text Mining: Classification, Clustering and Extraction Techniques,” 2017, [Online]. Available: http://arxiv.org/abs/1707.02919.

[16] Y. U. Zhong, K. Karu, and A. K. Jain, “LOCATING TEXT IN COMPLEX COLOR IMAGES,” vol. 28, no. 10, 1995.

[17] D. Lopresti, “Locating and Recognizing Text in WWW Images,” vol. 206, pp. 177–206, 2000.

[18] B. Sin, S. Kim, and B. Cho, “Locating characters in scene images using frequency features,” pp. 489–492, 2002.

[19] G. G. Devi, “AUTOMATIC TEXT EXTRACTION FROM COMPLEX COLORED IMAGES USING GAMMA CORRECTION METHOD,” vol. 10, no. 4, pp. 705–715, 2014, doi: 10.3844/jcssp.2014.705.715.

[20] M. Amin, B. Atitallah, and A. Ben Atitallah, “New Gamma Correction Method for real time image text extraction,” pp. 2–7, 2019.

[21] D. Li, Ed., “HSV Color Space,” in *Encyclopedia of Microfluidics and Nanofluidics*, Boston, MA: Springer US, 2008, p. 793.

[22] S. Lu, B. Su, and C. L. Tan, “Document image binarization using background estimation and stroke edges,” *Int. J. Doc. Anal. Recognit.*, vol. 13, no. 4, pp. 303–314, 2010, doi: 10.1007/s10032-010-0130-8.

[23] T. Lelore and F. Bouchara, “Super-resolved binarization of text based on the FAIR algorithm,” *Proc. Int. Conf. Doc. Anal. Recognition, ICDAR*, pp. 839–843, 2011, doi: 10.1109/ICDAR.2011.172.

[24] B. Gatos, I. Pratikakis, and S. J. Perantonis, “Adaptive degraded document image binarization,” *Pattern Recognit.*, vol. 39, no. 3, pp. 317–327, 2006, doi: 10.1016/j.patcog.2005.09.010.

[25] I. Cross, S. Sequence, A. C. Section, and S. Graph, “О Проблеме Борсука Для (0, 1)- И (-1, 0, 1)-Многогранников В Пространствах Малой Размерности,” *Труды Московского Физико-Технического Института*, vol. 4, no. 1–13, pp. 2150–2154, 2012.

[26] N. R. Howe, “Document binarization with automatic parameter tuning,” *Int. J. Doc. Anal. Recognit.*, vol. 16, no. 3, pp. 247–258, 2013, doi: 10.1007/s10032-012-0192-x.

[27] J. Wang, Y. Guo, Y. Ying, Y. Liu, and Q. Peng, “Fast non-local algorithm for image denoising,” *Proc. - Int. Conf. Image Process. ICIP*, no. 0, pp. 1429–1432, 2006, doi: 10.1109/ICIP.2006.312698.

[28] E. Vats, A. Hast, and P. Singh, “Automatic document image binarization using Bayesian optimization,” *ACM Int. Conf. Proceeding Ser.*, pp. 89–94, 2017, doi: 10.1145/3151509.3151520.

[29] N. Mitianoudis and N. Papamarkos, “Document image binarization using local

features and Gaussian mixture modeling," *Image and Vision Computing*, vol. 38. pp. 33–51, 2015, doi: 10.1016/j.imavis.2015.04.003.

[30] Z. Shi, S. Setlur, and V. Govindaraju, "Digital image enhancement using normalization techniques and their application to palmleaf manuscripts," *Cent. Excell. Doc. Anal. Recognit.*, no. February 2016, 2005, [Online]. Available: http://www.cedar.buffalo.edu/~zshi/Papers/kbcs04_261.pdf.

[31] N. S. Kumar, D. S. Kumar, S. Swathikiran, and A. P. James, "Ancient indian document analysis using cognitive memory network," *Proc. 2014 Int. Conf. Adv. Comput. Commun. Informatics, ICACCI 2014*, no. September, pp. 2665–2668, 2014, doi: 10.1109/ICACCI.2014.6968659.

[32] N. Otsu *et al.*, "Otsu_1979_otsu_method," *IEEE Trans. Syst. Man. Cybern.*, vol. C, no. 1, pp. 62–66, 1979.

[33] W. Niblack, "An introduction to digital image processing.," *An Introd. to Digit. image Process.*, 1986, doi: 10.1002/0470035528.ch15.

[34] "PRACHEENA KERALA LIPIKAL - RAVIVARMA L.A (www.malayalambooks.net).pdf." .

[35] G. Saravanan, G. Yamuna, and S. Nandhini, "Real Time Implementation of RGB to HSV / HSI / HSL and Its Reverse Color Space Models," pp. 462–466, 2016.

[36] R. M. Haralick and S. R. Sternberg, "Image Analysis Morphology," no. 4, pp. 532–550, 1987.

[37] R. G. Kuehni, "Color Space and Its Divisions," vol. 26, no. 3, 2001.

[38] V. Gupta, N. Sambyal, A. Sharma, and P. Kumar, "Restoration of artwork using deep neural networks," *Evol. Syst.*, no. 0123456789, 2019, doi: 10.1007/s12530-019-09303-7.

[39] M. Mandar, D. Sontakke, M. Meghana, and S. Kulkarni, "Different Types of Noises in Images and Noise Removing Technique," *Int. J. Adv. Technol. Eng. Sci.*, vol. 03, no. 01, pp. 102–115, 2015.

[40] A. C. Frery, "Image filtering," *Digit. Doc. Anal. Process.*, pp. 55–70, 2013, doi: 10.1201/b10797-8.

[41] M. Poulose, "Literature Survey on Image Deblurring Techniques," *Int. J. Comput. Appl. Technol. Res.*, vol. 2, no. 3, pp. 286–288, 2013, doi: 10.7753/ijcatr0203.1014.

[42] U. Sara, M. Akter, and M. S. Uddin, "Image Quality Assessment through FSIM, SSIM, MSE and PSNR—A Comparative Study," *J. Comput. Commun.*, vol. 07, no. 03, pp. 8–18, 2019, doi: 10.4236/jcc.2019.73002.

[43] C. H. Thum, "Measurement of the entropy of an image with application to image focusing," *Opt. Acta (Lond).*, vol. 31, no. 2, pp. 203–211, 1984, doi: 10.1080/713821475.

[44] Z. Wang and A. C. Bovik, “Error : Love It or Leave It ?,” no. January, pp. 98–117, 2009.

[45] J. L. Mannos and D. J. Sakrison, “The Effects of a Visual Fidelity Criterion on the Encoding of Images,” *IEEE Trans. Inf. Theory*, vol. 20, no. 4, pp. 525–536, 1974, doi: 10.1109/TIT.1974.1055250.

[46] K. Lyngby, “PEAK SIGNAL-TO-NOISE RATIO REVISITED : IS SIMPLE BEAUTIFUL ? Jari Korhonen Technical Univ . of Denmark (DTU) Dept . of Photonics Engineering Norwegian Univ . of Science and Tech . (NTNU) Dept . of Electronics and Telecommunications,” pp. 37–38, 2012.

[47] X. Min, G. Zhai, J. Zhou, M. C. Q. Farias, and A. C. Bovik, “Study of Subjective and Objective Quality Assessment of Audio-Visual Signals,” *IEEE Trans. Image Process.*, vol. 29, no. 6, pp. 6054–6068, 2020, doi: 10.1109/TIP.2020.2988148.

[48] E. G. Turitsyna and S. Webb, “Simple design of FBG-based VSB filters for ultra-dense WDM transmission ELECTRONICS LETTERS 20th January 2005,” *Electron. Lett.*, vol. 41, no. 2, pp. 40–41, 2005, doi: 10.1049/el.

[49] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: From error visibility to structural similarity,” *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, 2004, doi: 10.1109/TIP.2003.819861.

[50] M. P. Eckert and A. P. Bradley, “Perceptual quality metrics applied to still image compression,” *Signal Processing*, vol. 70, no. 3, pp. 177–200, 1998, doi: 10.1016/S0165-1684(98)00124-8.

[51] E. A. Silva, K. Panetta, and S. S. Agaian, “Quantifying image similarity using measure of enhancement by entropy,” *Mob. Multimedia/Image Process. Mil. Secur. Appl. 2007*, vol. 6579, p. 65790U, 2007, doi: 10.1117/12.720087.

[52] A. CHAPANIS, “Color Names for Color Space.,” *Am. Sci.*, vol. 53, no. 3, pp. 327–346, 1965.

[53] N. a Ibraheem, M. M. Hasan, R. Z. Khan, and P. K. Mishra, “Understanding Color Models : A Review,” *ARPN J. Sci. Technol.*, vol. 2, no. 3, pp. 265–275, 2012.

[54] A. N. Venetsanopoulos and K. N. Plataniotis, *Color Image Processing and Applications*. 2013.

[55] R. Rasras, E. El, and D. Skopin, “Developing a new color model for image analysis and processing,” *Comput. Sci. Inf. Syst.*, vol. 4, no. 1, pp. 43–55, 2007, doi: 10.2298/csis0701043r.

[56] I. Conference and D. Hutchison, *Image and*. 2014.

[57] C.-Y. Wen and C.-M. Chou, “Color Image Models and its Applications to Document Examination,” *Forensic Sci. Forensic Sci. Forensic Sci. Forensic Sci. Forensic Sci. J. J. J. J. J. Since*, vol. 3, pp. 23–32, 2002.

[58] “Color Space.” http://www.colorbasics.com/ColorSpace/ (accessed Jun. 30,

2021).

[59] “CSS Color Module Level 3.” https://www.w3.org/TR/css-color-3/#hsl-color (accessed Jun. 30, 2021).

[60] O. Marques, “Morphological Image Processing,” *Pract. Image Video Process. Using MATLAB®*, vol. 8491, pp. 299–334, 2011, doi: 10.1002/9781118093467.ch13.

[61] K. A. M. Said and A. B. Jambek, “A study on image processing using mathematical morphological,” *2016 3rd Int. Conf. Electron. Des. ICED 2016*, pp. 507–512, 2017, doi: 10.1109/ICED.2016.7804697.

[62] L. Fan, F. Zhang, H. Fan, and C. Zhang, “S42492-019-0016-7.Pdf,” vol. 7, 2019.

[63] S. Z. Syed Zaini *et al.*, “Image Quality Assessment for Image Segmentation Algorithms: Qualitative and Quantitative Analyses,” *Proc. - 9th IEEE Int. Conf. Control Syst. Comput. Eng. ICCSCE 2019*, pp. 66–71, 2019, doi: 10.1109/ICCSCE47578.2019.9068561.

[64] D. H. Shin, R. H. Park, S. Yang, and J. H. Jung, “Block-based noise estimation using adaptive Gaussian filtering,” *Dig. Tech. Pap. - IEEE Int. Conf. Consum. Electron.*, vol. 51, no. 1, pp. 263–264, 2005, doi: 10.1109/icce.2005.1429818.

[65] D. L. Donoho and J. M. Johnstone, “Ideal spatial adaptation by wavelet shrinkage,” *Biometrika*, vol. 81, no. 3, pp. 425–455, 1994, doi: 10.1093/biomet/81.3.425.

[66] W. Liu and W. Lin, “Additive white gaussian noise level estimation in SVD domain for images,” *IEEE Trans. Image Process.*, vol. 22, no. 3, pp. 872–883, 2013, doi: 10.1109/TIP.2012.2219544.

[67] “Mean Filters.” http://masters.donntu.org/2007/kita/gett/library/eng.htm (accessed Jul. 01, 2021).

[68] M. R. Rakesh, B. Ajeya, and A. R. Mohan, “Hybrid Median Filter for Impulse Noise Removal of an Image in Image Restoration,” *Int. J. Adv. Res. Electr. Electron. Instrum. Eng.*, vol. 2, no. 10, pp. 5117–5124, 2013.

CHAPTER 8: LIST OF PUBLICATIONS

Paper	Title. Author list. Conference/Journal	Status
Paper 1	**BGR to HSV based text extraction from manuscripts using slidebars** Mayank Singh, Prof. S. Indu ASIANCON 2021	Accepted
Paper 2	**Denoising of Palm leaf manuscripts using Gaussian filter and Conservative Smoothing** Mayank Singh, Prof. S. Indu International Conference on Advances in Manufacturing Technologies and Application of Artificial Intelligence(ICAMTAAI) 2021	Accepted