COLORISATION OF GRAYSCALE MEDIA

A PROJECT REPORT

Submitted By

GAURAV PANDEY	**312212104031**
HARIHARAN S	**312212104034**
LAKSHMAN RAMESH	**312212104045**

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

SSN COLLEGE OF ENGINEERING

KALAVAKKAM 603110

ANNA UNIVERSITY :: CHENNAI - 600025

April 2016

ANNA UNIVERSITY : CHENNAI 600025

BONAFIDE CERTIFICATE

Certified that this project report titled **“COLORISATION OF GRAYSCALE MEDIA”** is the *bonafide* work of “**GAURAV PANDEY (312212104031)**, **HARIHARAN S (312212104034)**, and **LAKSHMAN RAMESH (312212104045)**” who carried out the project work under my supervision.

Dr. CHITRA BABU
Head of the Department
Professor,
Department of CSE,
SSN College of Engineering,
Kalavakkam - 603 110

Ms. K. MADHESWARI
Supervisor
Assistant Professor,
Department of CSE,
SSN College of Engineering,
Kalavakkam - 603 110

Place:
Date:

Submitted for the examination held on. . . . .

Internal Examiner **External Examiner**

ACKNOWLEDGEMENTS

We thank God, the almighty for giving us strength and knowledge to do this project.

We would like to thank and express a deep sense of gratitude to our guide **Ms. K. MADHESWARI**, Assistant Professor, Department of Computer Science and Engineering, for her valuable advice and suggestions as well as her continued guidance, patience and support that helped us to shape and refine our work.

Our sincere thanks to **Dr. CHITRA BABU**, Professor and Head of the Department of Computer Science and Engineering, for her words of advice and encouragement and we would like to thank our project Coordinator **Dr. S. SHEERAZUDDIN**, Associate Professor, Department of Computer Science and Engineering for his valuable suggestions throughout the first phase of this project.

We express our deep respect to the founder **Dr. SHIV NADAR**, Chairman, SSN Institutions. We also express our appreciation to our **Dr. S. SALIVAHANAN**, Principal, for all the help he has rendered during this course of study.

We would like to extend our sincere thanks to all the teaching and non-teaching staffs of our department who have contributed directly and indirectly during the course of our project work.

Finally, We would like to thank our parents and friends for their patience, cooperation and moral support throughout my life.

Gaurav Pandey **Hariharan S** **Lakshman Ramesh**

ABSTRACT

When an image is converted to grayscale, all the properties of colour are lost in translation. Several algorithms exist to provide the base for the lost information but none of them are perfect as the basis of colorisation is assumption. Our method in this endeavour is to completely automate the process where we require the input of a grayscale image (image to be colorised) and based on whether the choice of colorisation is ***pattern matching*** or ***user provided reference image*** (image based on which it is colorised) we transfer colour from the reference image to the original image after performing several functions including correlation analysis, histogram equalisation, luminance comparison and histogram specification.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

Introduction

1.1 Premise

Since the human visual system is sensitive to colours rather than gray shades, we aim to emphasize the appearance of black and white movies and recolor them to obtain near natural coloured movies that look like their original colours.

Any coloured image, also called RGB image, consists of three channels each of 8 bit/pixel bit depth, while a grayscale image, like a black and white photograph or TV movie, contains only grays represented as only one channel has values from 0 to 255 (1 byte), also called the intensity channel. To give any gray image its colours back, there are extra two channels of 256 x 256 combinations values are needed, what make it impossible to estimate the image original colours.

There are three methods involved when colouring the image, they are manual, semi-automatic and automatic colouring techniques. Manual method is a labour intensive process where the user colours each and every frame but it is supposed to produce the best result. In semi-automatic method we scribble a colour at a particular region and the colour spreads to the entire region of same intensity. In automatic colouring method we transfer the colour from one image to another without any human intervention. Since we have proposed automatic colouring our concern is about colouring images using a similar source image.

There are two methods involved in this process, they are Segmentation and Histogram Transfer. Segmentation is the process of identifying discrete segments in our target image in terms of grayscale intensities and transferring colour from those respective gray intensities from source image to target image. But it is a time consuming process providing a better result than Histogram Transfer.

In our proposed method we have used histogram transfer where we already have the RGB histogram for the source image. Since our target image is grayscale, we transfer the RGB histogram of the source image to the target image. For colouring a movie, the time is a major factor for colouring process. Since one hour movie of frame rate 30 fps has 108000 (60 x 60 x 30) frames, the process of colouring the movie frame by frame will cost so much money and effort and spend so much time.

For instance, the colouring of the classic movie Casablanca in 1988 took more than two months and nearly USD 450,000 to complete; in a process that involved among many other labour intensive steps researching historical wardrobe notes from original movies set to discover the specific colours that the actors and actresses wore in most of the scenes.

1.2 Histogram Equalisation

Histogram equalisation is a method in image processing of contrast adjustment using the image's histogram.

Histogram equalisation seeks to flatten your image histogram. It models the image as a probability density function (or in simpler terms, a histogram where you normalise each entry by the total number of pixels in the image) and tries to ensure that the probability for a pixel to take on a particular intensity is equi-probable (with equal probability).

The premise behind histogram equalisation is for images that have poor contrast. Images that are too dark, or too washed out, or too bright are good candidates to apply histogram equalisation. Plotting the histogram, the spread of the pixels is limited to a very narrow range. By applying histogram equalisation, the histogram will thus flatten and give you a better contrast image. The effect of this with the histogram is that it stretches the dynamic range of your histogram.

The method is useful in images with backgrounds and foregrounds that are both bright or both dark. In particular, the method can lead to better views of bone structure in x-ray images, and to better detail in photographs that are over or under-exposed. A key advantage of the method is that it is a fairly straightforward technique and an invertible operator.

So in theory, if the histogram equalisation function is known, then the original histogram can be recovered. The calculation is not computationally intensive. A disadvantage of the method is that it is indiscriminate. It may increase the contrast of background noise, while decreasing the usable signal.

Histogram equalisation often produces unrealistic effects in photographs; however it is very useful for scientific images like thermal, satellite or x-ray images, often the same class of images to which one would apply false colour.

Histogram equalisation will work the best when applied to images with much higher colour depth than palette size, like continuous data or 16-bit grayscale images.

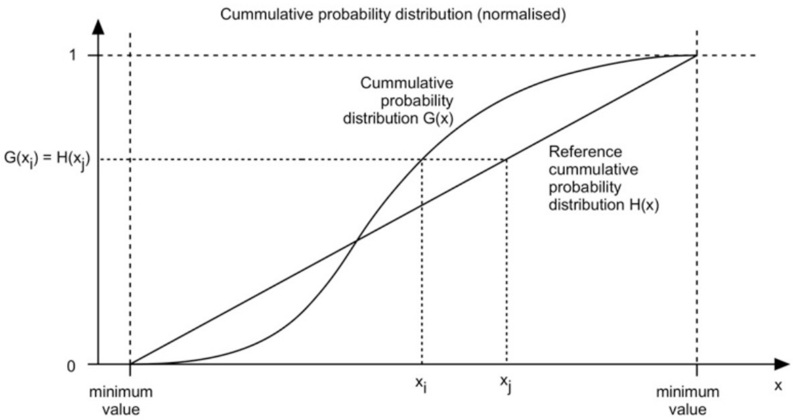

FIGURE 1.1: Histogram Equalization Graph

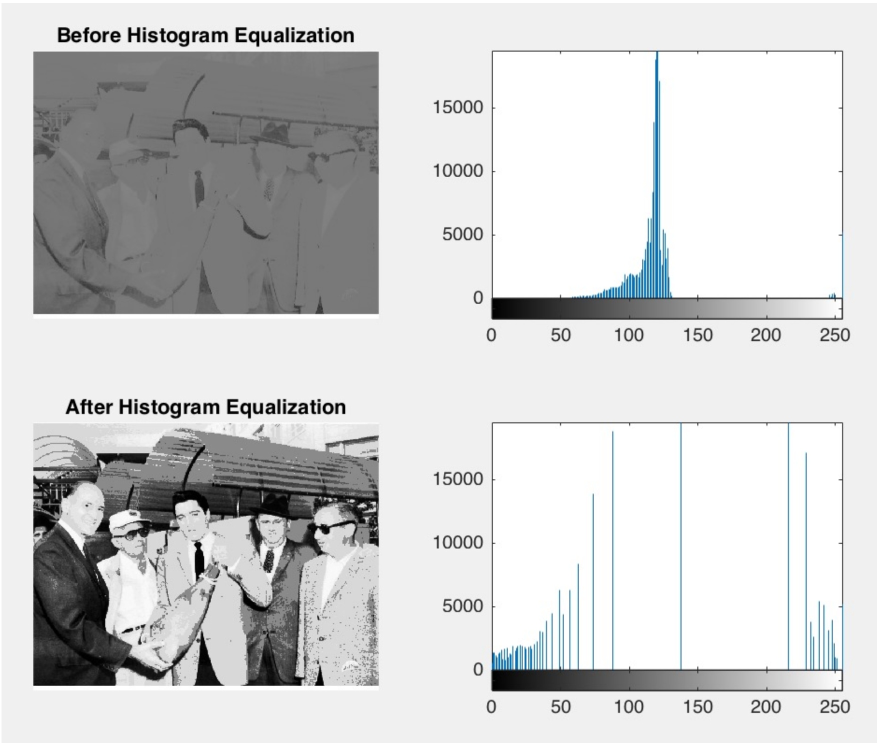

FIGURE 1.2: Histogram Equalization Example - 1

FIGURE 1.3: High Contrast Photograph

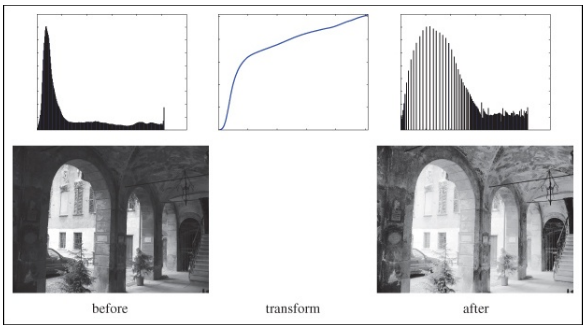

FIGURE 1.4: Histogram Equalization Example - 2

1.3 Histogram Matching (also Histogram Specification)

Histogram matching is a process where a time series, image, or higher dimension scalar data is modified such that its histogram matches that of another (reference) dataset. A common application of this is to match the images from two sensors with slightly different responses, or from a sensor whose response changes over time.

The algorithm is as follows. The cumulative histogram is computed for each dataset, see the diagram below. For any particular value (x_i) in the data to be adjusted has a cumulative histogram value given by $G(x_i)$. This in turn is the cumulative distribution value in the reference dataset, namely $H(x_j)$. The input

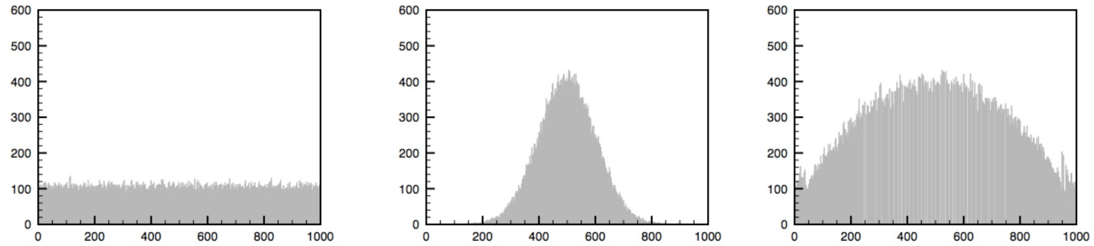

FIGURE 1.5: Histogram Matching Diagram

data value x_i is replaced by x_j. In practice for discrete valued data one does not step through data values but rather creates a mapping to the output state for each possible input state. In the case of an image this would be a mapping for each of the 256 different states.

For RGB images the histogram matching can be applied in one of two ways: it can be applied to each colour channel independently or a single mapping applied to all channels. In the later case this single mapping can be derived from a grayscale version of the image, the intensity, luminance, or other similar single measures. In the case where the matching is applied on a per channel basis, colouration effects can occur particularly if one or more channels has a narrow distribution.

1.4 Cumulative Distribution Function

Cumulative Distribution Function or just distribution function, describes the probability that a real-valued random variable X with a given probability distribution will be found at a value less than or equal to x.

$$T(k) = floor((L-1)\sum_{n=0}^{k} p_n)$$

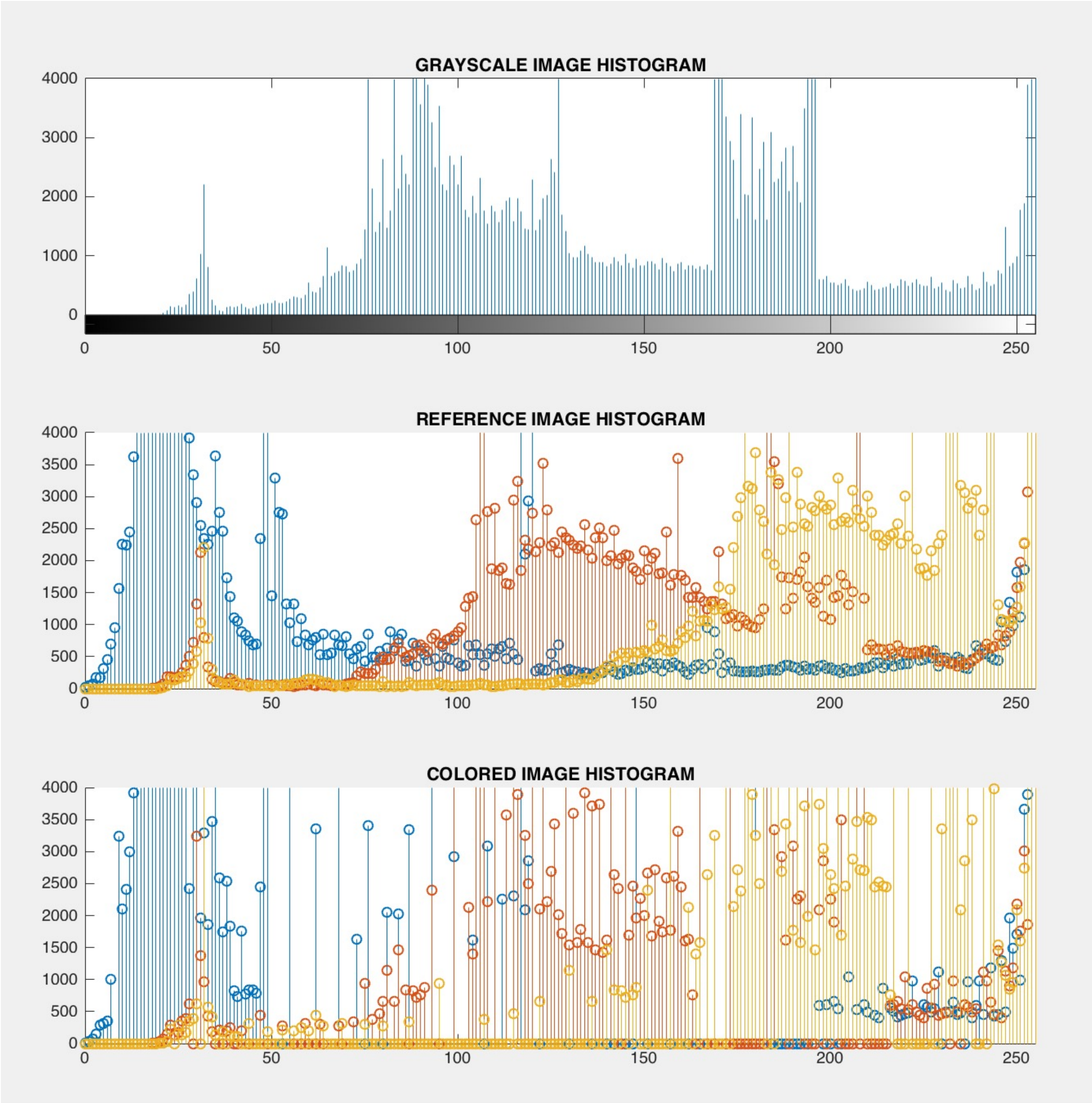

FIGURE 1.6: Histogram Matching Application

- p_n - Probability of encountering a pixel with intensity n in your image.
- L - Number of bins

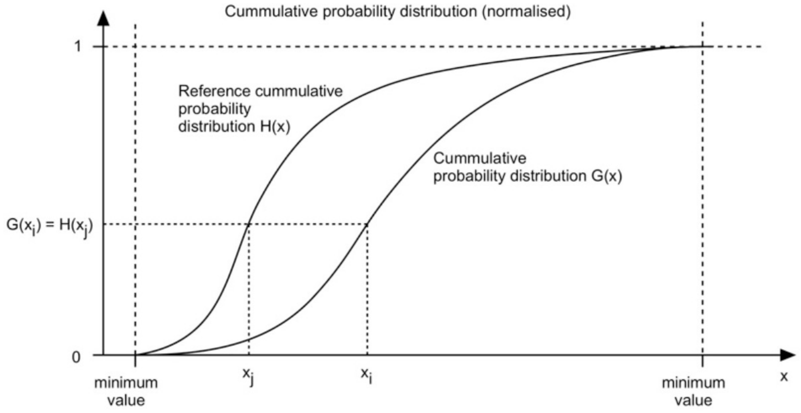

FIGURE 1.7: Histogram Matching Graph

1.5 Digital Image Correlation

Digital image correlation and tracking is an optical method that employs tracking and image registration techniques for accurate 2D and 3D measurements of changes in images. This is often used to measure deformation (engineering), displacement, strain, and optical flow, but it is widely applied in many areas of science and engineering. One very common application is for measuring the motion of an optical mouse.

The two-dimensional discrete cross correlation r_{ij} can be defined several ways, one possibility being:

$$r_{ij} = \frac{\sum_m \sum_n [f(m+i, n+j) - \bar{f}][g(m,n) - \bar{g}]}{\sqrt{\sum_m \sum_n [f(m,n) - \bar{f}]^2 \sum_m \sum_n [g(m,n) - \bar{(g)}]^2}}$$

Here,

- $f(m,n)$ is the pixel intensity or the grayscale value at a point (m,n) in the original image

- $g(m,n)$ is the grayscale value at a point (m,n) in the translated image
- $\bar{f}$ and $\bar{g}$ are mean values of the intensity matrices f and g, respectively.

CHAPTER 2

Literature Survey

Colorisation has been extensively studied in recent years. Various methods have been proposed to solve this challenging task. However, most of them are semiautomatic methods which require some amounts of user interactions.

Rafael Gonzalez and Richard Woods (1987) proposed the method of luminance keying to transfer colour to a grayscale image. This method utilises a user defined look-up table to assign a colour to each grayscale value. When applying different colours at the same intensity level, the user should simultaneously use a few luminance keys for different regions manually, making the process very tedious. [1]

Welsh et al. (2002) introduced a colorisation method by transferring colour from a source colour image to a target grayscale image via matching colour information between the images. It is inspired by the method of colour transfer between images (Reinhard et al., 2001) and the idea of image analogy (Hertzmann et al., 2001). This method needs the user to match the areas using swatches. [8]

Levin et al. (2004) introduced an interactive colorisation method based on the premise that nearby pixels in space that had similar grey levels should also have similar colours. Without considering the boundary and region information, unseemly colour sometimes leaks from one region to others. [2]

Hideki Noda and Michiharu Niimi (2006) introduced a colorisation method in YCbCr colour space which is based on the maximum a posteriori estimation of a colour image given a monochrome image. The proposed colorisation in YCbCr was also applied to JPEG compressed colour images aiming at better recovery of down sampled chrominance planes.[7]

Yatziv and Sapiro (2006) proposed a method for image and video colorisation using chrominance blending. This method is computationally simple and effective. However, to realise desired results, it needs the user to mark some chrominance scribbles. [10]

Nie et al. (2007) improved (Levin et al., 2004) and presented an optimisation based interactive grayscale image colorisation method. It is reported that this method gives the same good quality of colorised images as the method of (Levin et al., 2004) with a fraction of the computational cost.[6] Nie et al. (2005) proposed a grayscale image colorisation method based on a local correlation based optimisation algorithm. [5] However, this method is restricted to some assumptions about the colour correlativity between pixels in different regions.

Luan et al. (2007) presented an interactive system for colorising the natural images. The colorisation procedure is separated into two stages: colour labelling and colour mapping. This method is effective for natural image colorisation. However, to obtain good colorisation results, the user had to draw multiple stokes on similar patterns with different orientation and scales. [4]

Liu et al. (2008) proposed an example based colorisation technique considering illumination differences between grayscale image and colour reference images. This method need to search suitable reference images from the web. To overcome these shortcomings, we proposed a novel automatic grayscale image colorisation method using a reference image provided by the user. [3]

Another topic related to our method is **histogram specification or mapping.**

Xiao and Ma (2009) adopted histogram matching in the lab colour space to preserve the gradients of the images when performing colour transferring. To address the problem of eliminating unwanted colour variation between similar image pairs, Senanayake and Alexander (2007) proposed a colour transfer method by feature based histogram registration. The source and target histograms are aligned based on corresponding features that are persistent peaks through scale space using a polynomial mapping. However, this method will fail when the histograms of the image pairs are widely different. [9]

CHAPTER 3

Problem Definition

When an image is converted to grayscale, all the properties of colour are lost in translation. Several algorithms exist to provide the base for the lost information (colorisation) but most of them arent designed to perfection, as the basis of colorisation is assumption.

Several algorithms exist out of which some are partly automated, some require user input. **Gonzalez and Woods** proposed a luminance keying to transfer colour to a grayscale image with the help of a user define look up table, **Welsh et al** used an automated colorisation algorithm through a reference image provided by the user, **Levin et al** required the input of the user through scribbles, **Leu et al** used a probabilistic approach and boundary detection to colour the images and **Noda and Niimi** introduced a method in YCbCr colour space which was based on the maximum a posteriori estimation given a monochrome image, **Xiao and Ma** adopted histogram matching in the lab colour space to preserve the gradients of the images when performing colour transferring, **Senanayake and Alexander** proposed a colour transfer method by feature based histogram registration.

Our proposed method in contrast is completely automated where we require almost no user intervention except for providing the input source image or source video (based on which the colourisation occurs). After accepting the image we check the pre-built database for a match to the current image using pattern matching and if the specified input is a video, correlation analysis is performed to determine the multiple matches required to colour the image. Prior colourisation, we determine the contrast of the image and perform histogram equalisation if it is of poor contrast to dynamically stretch the image. Following this, we perform histogram specification to transfer the colour from the source image to the target image.

CHAPTER 4

Proposed System

Proposed method in contrast is completely automated where we require almost no user intervention except for providing the input source image or source video. Following this we perform several functions including correlation analysis, histogram equalisation, pattern matching, histogram specification.

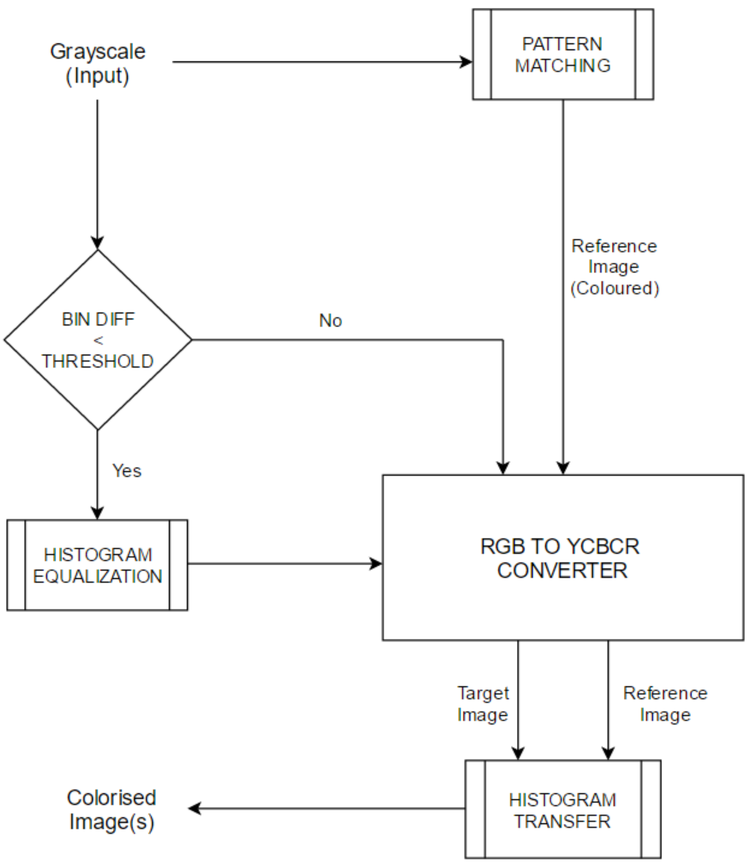

FIGURE 4.1: Proposed System

4.1 Correlation Analysis

The Correlation Analysis Module is required only for videos which consists of multiple sequences. Videos generally occur at a frame rate of at least 20 frames per second and thereby consists of a large number of frames. It would be a tedious process to search for the image which closely matches the frame and transfer colour to each frame in a video. Therefore, we reduce the time complexity by searching for reference images in the database only for the few key frames where the sequence change occurs. Frames which do not deviate much from its luminance intensity and spatial values do not vary much chromatically and so for these set of images we use a single key frame for which the closest match from database is determined. Then we transfer colour from the reference images to the group of images which are found using correlation analysis. We perform this process iteratively to determine the next key frame until all the frames in a video are coloured.

$$corr2(A,B) = \frac{\sum_M \sum_N (A_{MN} - \bar{A})(B_{MN} - \bar{B})}{\sqrt{(\sum_M \sum_N (A_{MN} - \bar{A})^2)(\sum_M \sum_N (B_{MN} - \bar{B})^2)}}$$

Here, A and B are two images for which the correlation value is found.

- A_{MN} and B_{MN} are the intensities of the pixel located in the M^{th} row and the N^{th} column.

- $\bar{A}$ and $\bar{B}$ are the mean intensities of the pixels of the entire respective images.

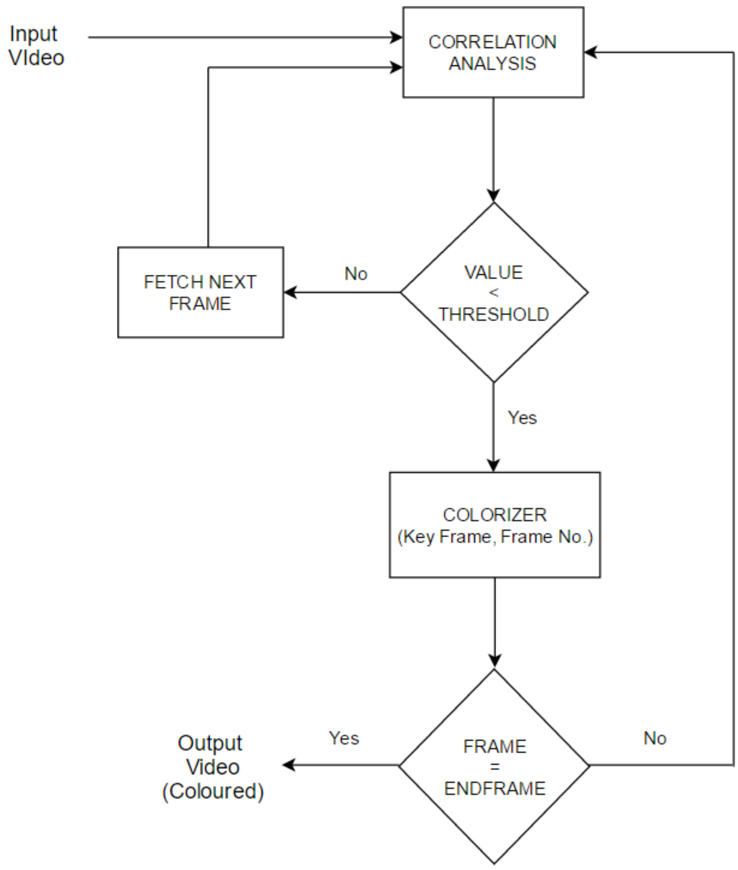

FIGURE 4.2: Correlation Analysis

4.2 Pattern Matching

In the Pattern Matching Module we determine the closest image for our target grayscale image which is to be coloured by going through the set of images in our database. The luminance values of the images generally occur in the range 0 to 255 (0 black , 255 white). Firstly, the count of the pixels with a particular intensity is determine and stored in a histogram vector. This vector is used to find the closest matching image from our database in terms of luminance intensity. Secondly, we convert the colour images in our database to grayscale form and find its histogram vector. Next, we find the absolute difference of the histogram vectors between the target image and fetched images from the database, and finally we find the image with smallest difference giving the closest matching image.

$$ReferenceImage = Min(\sum_{k=0}^{255} abs(I(k) - R_j(k)))$$

Where

- I - Original gray-scale image to be coloured

- R - Images in database

$1 \leq j \leq$ N (Number of images in Database).

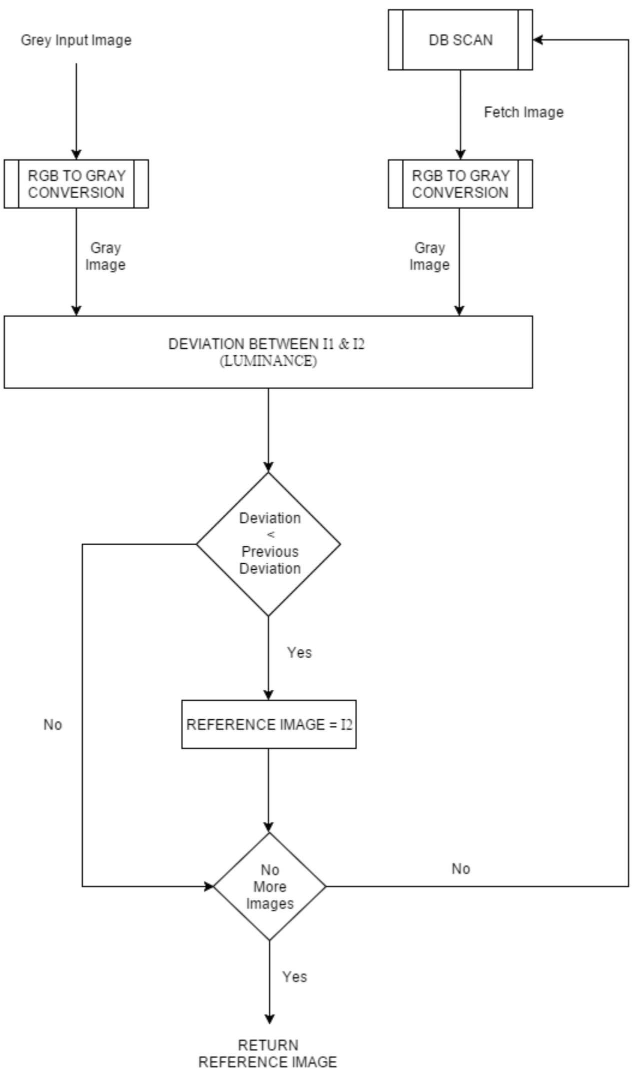

FIGURE 4.3: Pattern Matching

4.3 Histogram Equalisation

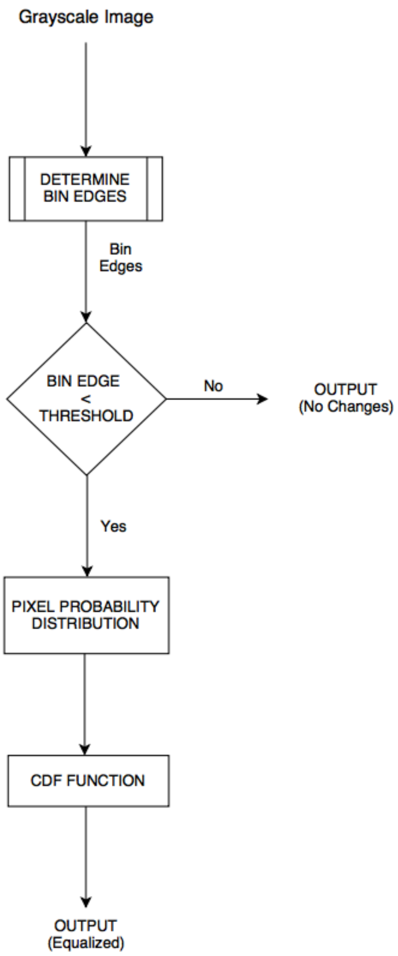

FIGURE 4.4: Histogram Equalisation

Histogram Equalisation Module follows a technique for adjusting the image intensities to enhance contrast. It models the image as a probability density function. The premise behind histogram equalisation are images that have poor contrast. Examples include images that look like they are too dark, or too washed out, or too bright. If you plot the histogram, the spread of the pixels is limited to a very narrow range, histogram equalisation will thus flatten and give you a better contrast image. The effect of this with the histogram is that it stretches the dynamic range of your histogram.

Let f be a given image represented as a m_r by m_c matrix of integer pixel intensities ranging from 0 to $L1$. L is the number of possible intensity values, often 256. Let p denote the normalised histogram of f with a bin for each possible intensity. So

$$p_n = \frac{\text{number of pixels with intensity } n}{\text{total number of pixels}}$$

where n = 0,1,...,L-1.

The histogram equalised image g will be defined by

$$T(k) = floor((L-1)\sum_{n=0}^{k} p_n)$$

- p_n is the probability that you would encounter a pixel with intensity n in your image.
- L - Number of bins

4.4 Histogram Matching

Histogram Matching Module performs histogram specification where a time series, image, or higher dimension scalar data is modified such that its histogram matches that of another (reference) dataset.

The algorithm is as follows. The cumulative histogram is computed for each dataset, see the diagram below. For any particular value (x_i) in the data to be adjusted has a cumulative histogram value given by $G(x_i)$. This in turn is the cumulative distribution value in the reference dataset, namely $H(x_j)$. The input

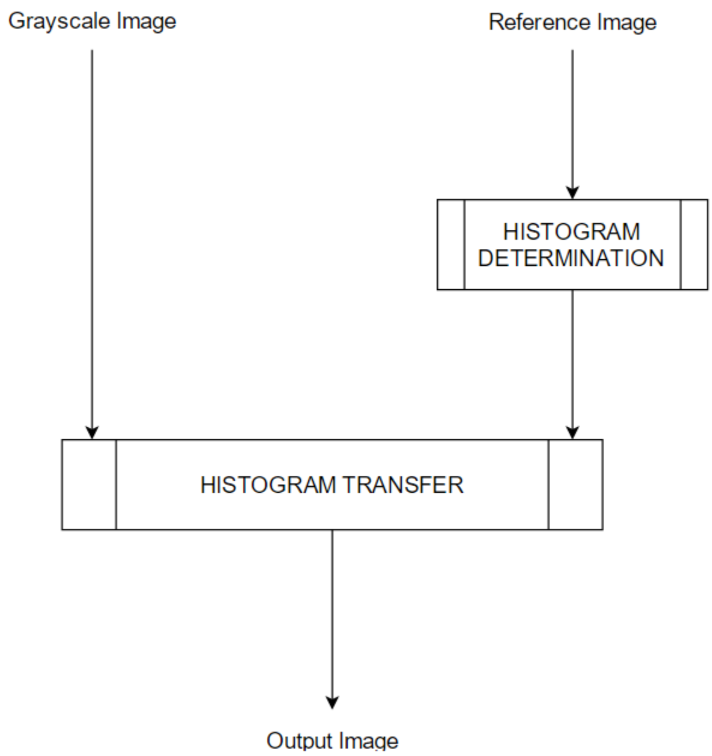

FIGURE 4.5: Histogram Matching

data value x_i is replaced by x_j.

$$|H_x[i] - H_z[j]| = min_k |H_x[i] - H_z[k]|$$

In practice for discrete valued data one does not step through data values but rather creates a mapping to the output state for each possible input state. In the case of an image this would be a mapping for each of the 256 different states.

Find histogram of input image h_x, and its cumulative H_x, the histogram equalisation mapping function. Specify the desired histogram h_z and find its corresponding cumulative mapping function H_z.

CHAPTER 5

Implementation and Results

The input for this setup could be both images and videos. There are no restrictions regarding the dimensions or the limit to the number of frames in a video. All operations are done in MATLAB and a database of images is maintained using MYSQL. Depending on the user provided image the MYSQL Database is searched to determine the exact match and further processing is done in MATLAB.

As per our tests the processing time is 3.66 minutes or 220 seconds for 1000 frames.

5.1 Colorisation Process

5.1.1 Image Colorisation

Our method is completely automated where we require no user intervention except for providing the input source image or source video (based on which the colourisation occurs). After accepting the image we check the pre built database for a match to the current image using pattern matching and if the specified input is a video, correlation analysis is performed to determine the multiple matches required to colour the image. Prior colourisation,we determine the contrast of the image and perform histogram equalisation if it is of poor contrast to dynamically stretch the image. Following this, we perform histogram specification to transfer the colour from the source image to the target image

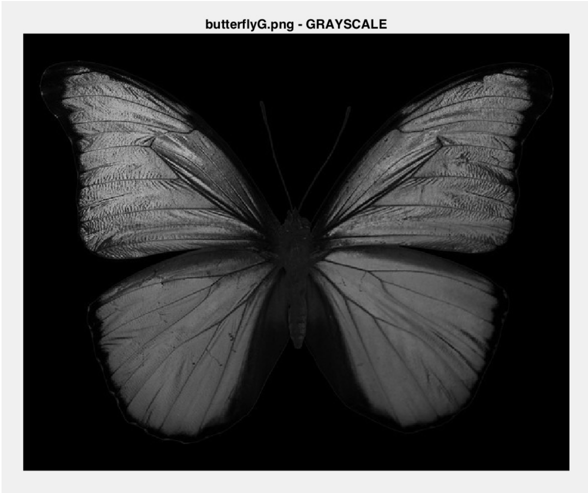

FIGURE 5.1: Input - Grayscale Image

Image colorisation process takes a relatively shorter time. The first step of the process is performing pattern matching for the given image in the database. The closest image is determined and returned. With the returned image histogram matching is performed to colourise the image by matching the histograms.

FIGURE 5.2: Input - Reference Image

As shown the grayscale image butterflyG.png is provided as the input. The database is searched returning the reference image butterfly2.png with which the input image is coloured resulting in coloured.png. The image butterfly1.png is the

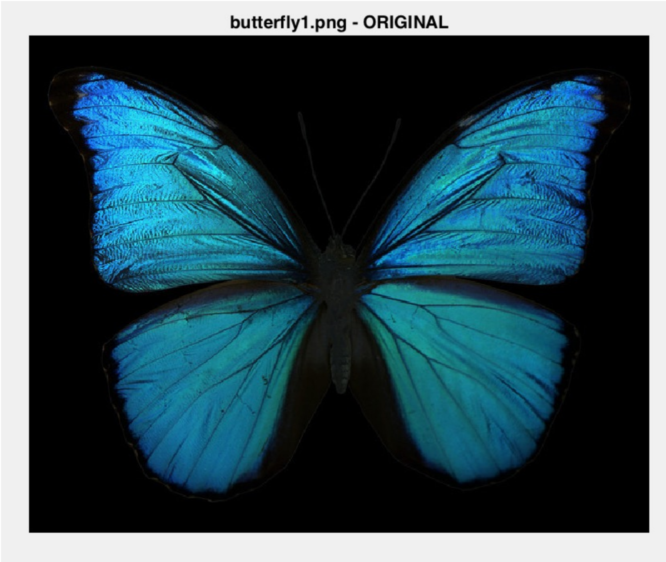

FIGURE 5.3: Reference Image - Original

input image with its original colour. The difference in colour is very perceptible with the saddle and the womans attire comparatively more visible.

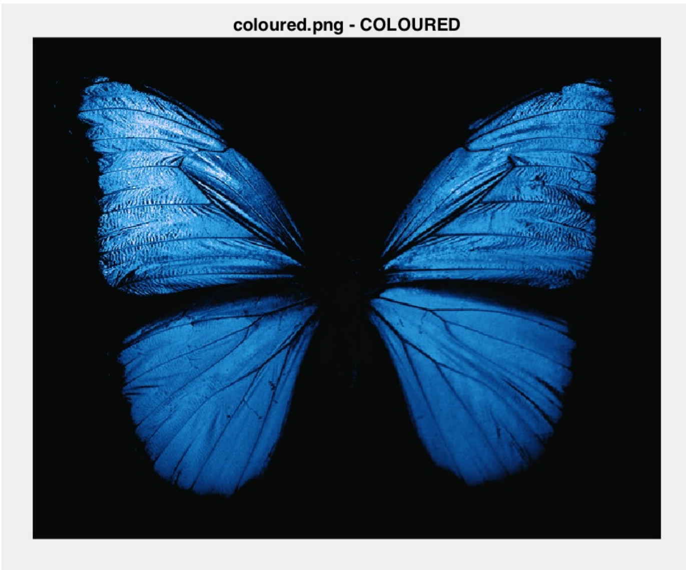

FIGURE 5.4: Output - Colored

5.1.2 Video Colorisation

Frame by frame processing of a video helps determine the key frames required to colour the frames in between them.

The process is as follows, the first frame (considered the first key frame) from the video is pattern matched to an image in a database.

Next, each frame following this is coloured using the first key frame after performing correlation analysis. Correlation analysis is the process where the similarity between two frames is measured, if the value occurs within a certain threshold then they are similar and colorisation occurs by histogram specification.

Following this, when the analysis results in a value lower than the threshold then a new keyframe is determined.

The initial process if followed again till the discovery of a new key frame and is repeated till the last frame in the video.

The first grid displays the frames in a sample video. The key frames are listed as it occurs. For the first key frame pattern matching outputs the first image sunset.png, the second key frame matches horse.png and the third matches silhouette.png.

The second displays the pattern matched images from the database based on the provided input.

The third grid displays the colorised frames following the colorisation process with respect to the pattern matched images.

FIGURE 5.5: Frames - Input Video

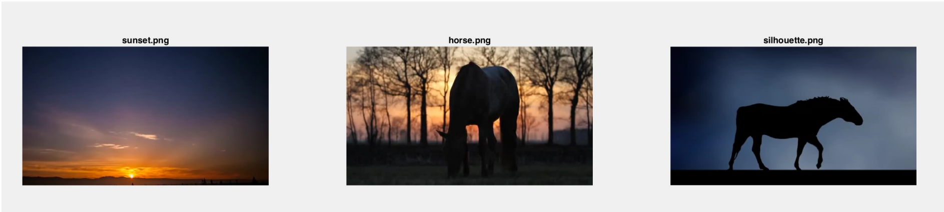

FIGURE 5.6: Pattern Matched Images

FIGURE 5.7: Frames - Colorised Video

5.2 Measures

5.2.1 Entropy

Entropy is a concept which originally arose from the study of the physics of heat engines. It can be described as a measure of the amount of disorder in a system. An organised structure, such as a crystal or a living organism, is very highly ordered and consequently has low entropy. When the crystal is heated sufficiently, it melts

TABLE 5.1: Entropy Sample

Images	R Channel	G Channel	B Channel
sunset.png - Coloured	7.430	6.913	6.592
sunset.png - Original	7.532	6.916	6.768
horse.png - Coloured	6.089	6.082	6.065
horse.png - Original	5.982	6.283	6.192
silhouette.png - Coloured	6.964	6.981	6.675
silhouette.png - Original	7.228	7.099	6.884

and becomes liquid, a much less ordered state. When the organism dies, it decays and becomes completely disrupted. In either system, its entropy increases.

Another way of expressing entropy is to consider the spread of states which a system can adopt. A low entropy system occupies a small number of such states, while a high entropy system occupies a large number of states.

In the case of an image, these states correspond to the grey levels which the individual pixels can adopt. For example, in an 8-bit pixel there are 256 such states. If all such states are equally occupied, as they are in the case of an image which has been perfectly histogram equalised, the spread of states is a maximum, as is the entropy of the image. On the other hand, if the image has been thresholded, so that only two states are occupied, the entropy is low. If all of the pixels have the same value, the entropy of the image is zero.

Note that, in this progression, as the entropy of the image is decreased, so is its information content. We moved from a full grey scale image, with high entropy, to a thresholded binary image, with low entropy, to a single-valued image, with zero entropy.

Information, in this context, refers to the announcement of the unexpected. If the pixels of an image were inspected, and found to be the same. This information could have been communicated in a very short message. The information content is said to be low simply because it can be communicated in a short message. If the pixels are changing in unexpected ways, however, longer messages are required to communicate this fact and the information is said to increase. This assumes, of course, that all changes in the image are meaningful. Changes due to noise are still considered to be information in that they describe the image as it actually is, rather than as it should be.

$$H = -\sum_{k=0}^{M-1} p_k \log_2(p_k)$$

- M is the number of grey levels
- pk is the probability associated with grey level k

Minimum entropy is achieved when the image itself is constant, that is, all of the pixels have the same grey level k.

5.2.2 Mean Intensity

TABLE 5.2: Mean Intensity Sample

Images	Mean I	R Channel	G Channel	B Channel
sunset.png - Coloured	78.959	94.933	69.675	72.271
sunset.png - Original	67.345	77.945	61.552	62.539
horse.png - Coloured	79.648	64.683	76.625	97.636
horse.png - Original	57.307	43.738	55.027	73.157
silhouette.png - Coloured	66.845	78.977	66.720	54.838
silhouette.png - Original	68.500	84.825	67.332	53.344

Intensity images measure the amount of light impinging on a photosensitive device. The input to the photosensitive device, typically a camera, is the incoming light, which enters the camera's lens and hits the image plane. In a digital camera, the physical image plane is an array which contains a rectangular grid of photosensors, each sensitive to light intensity. The output of the array is a continuous electric signal, the video signal. The video signal is sent to an electronic device called frame grabber, where it is digitised into a 2D rectangular array of integer values and stored in a memory buffer.

The interpretation of an intensity image depends strongly on the characteristics of the camera called the camera parameters. The parameters can be separated into extrinsic and intrinsic parameters. The extrinsic parameters transform the camera reference frame to the world reference frame. The intrinsic parameters describe the optical, geometric and digital characteristics of the camera. One parameter, for example, can describe the geometric distortion introduced by the optics.

In an image, intensity is a data matrix, I, whose values represent intensities within some range. MATLAB stores an intensity image as a single matrix, with each element of the matrix corresponding to one image pixel. The matrix can be of class double, uint8, or uint16.While intensity images are rarely saved with a colormap, MATLAB uses a colormap to display them.

The elements in the intensity matrix represent various intensities, or gray levels, where the intensity 0 usually represents black and the intensity 1, 255, or 65535 usually represents full intensity, or white.

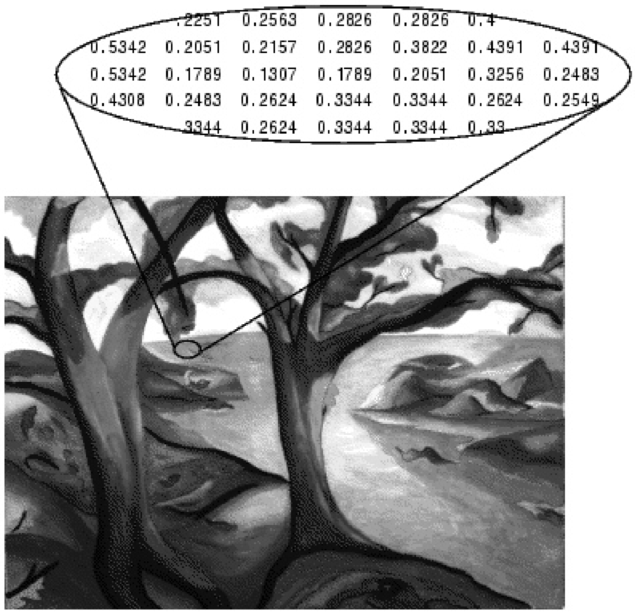

FIGURE 5.8: Intensity Values in an Image

5.2.3 Peak Signal to Noise Ratio

PSNR is most commonly used to measure the quality of reconstruction of lossy compression codecs (e.g., for image compression). The signal in this case is the original data, and the noise is the error introduced by compression. When comparing compression codecs, PSNR is an approximation to human perception of reconstruction quality. Although a higher PSNR generally indicates that the reconstruction is of higher quality, in some cases it may not. One has to be extremely careful with the range of validity of this metric; it is only conclusively valid when it is used to compare results from the same codec (or codec type) and same content.

Image enhancement or improving the visual quality of a digital image can be subjective. Saying that one method provides a better quality image could vary from person to person. For this reason, it is necessary to establish quantitative/empirical measures to compare the effects of image enhancement

TABLE 5.3: PSNR Sample

Images	Peak SNR	SNR
sunset.png Coloured & sunset.png Original	20.372	10.252
horse.png Coloured & horse.png Original	19.731	8.919
silhouette.png Coloured & silhouette.png Original	22.399	13.603

algorithms on image quality. Using the same set of tests images, different image enhancement algorithms can be compared systematically to identify whether a particular algorithm produces better results. The metric under investigation is the peak-signal-to-noise ratio. If we can show that an algorithm or set of algorithms can enhance a degraded known image to more closely resemble the original, then we can more accurately conclude that it is a better algorithm.

Typical values for the PSNR in lossy image and video compression are between 30 and 50dB, provided the bit depth is 8bits, where higher is better. For 16-bit data typical values for the PSNR are between 60 and 80dB. Acceptable values for wireless transmission quality loss are considered to be about 20dB to 25dB.

FIGURE 5.9: PSNR Values at Various Qualities

In the absence of noise, the two images I and K are identical, and thus the MSE is zero. In this case the PSNR is infinite.

5.2.4 Hue, Saturation and Brightness

TABLE 5.4: HSB Sample

Images	H Channel	S Channel	B Channel
sunset.png - Coloured	0.603	0.301	0.380
sunset.png - Original	0.456	0.473	0.362
horse.png - Coloured	0.612	0.471	0.383
horse.png - Original	0.487	0.413	0.287
silhouette.png - Coloured	0.094	0.228	0.310
silhouette.png - Original	0.219	0.287	0.337

Hue, saturation, and brightness are aspects of colour in the red, green, and blue (RGB) scheme. These terms are most often used in reference to the colour of each pixel in a cathode ray tube (CRT) display. All possible colours can be specified according to hue, saturation, and brightness (also called brilliance), just as colours can be represented in terms of the R, G, and B components.

Most sources of visible light contain energy over a band of wavelengths. Hue is the wavelength within the visible-light spectrum at which the energy output from a source is greatest. This is shown as the peak of the curves in the accompanying graph of intensity versus wavelength. In this example, all three colours have the same hue, with a wavelength slightly longer than 500 nanometers, in the yellow-green portion of the spectrum.

Saturation is an expression for the relative bandwidth of the visible output from a light source. In the diagram, the saturation is represented by the steepness of the slopes of the curves. Here, the red curve represents a colour having low saturation, the green curve represents a colour having greater saturation, and the blue curve represents a colour with fairly high saturation. As saturation increases, colours appear more ”pure.” As saturation decreases, colours appear more washed-out.”

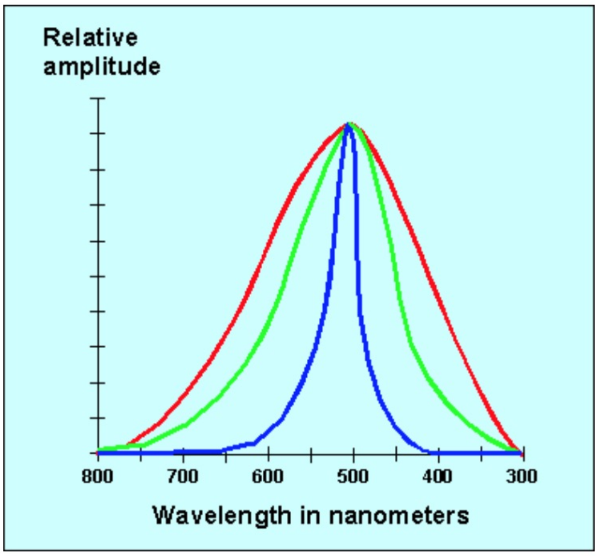

FIGURE 5.10: Amplitude - HSB

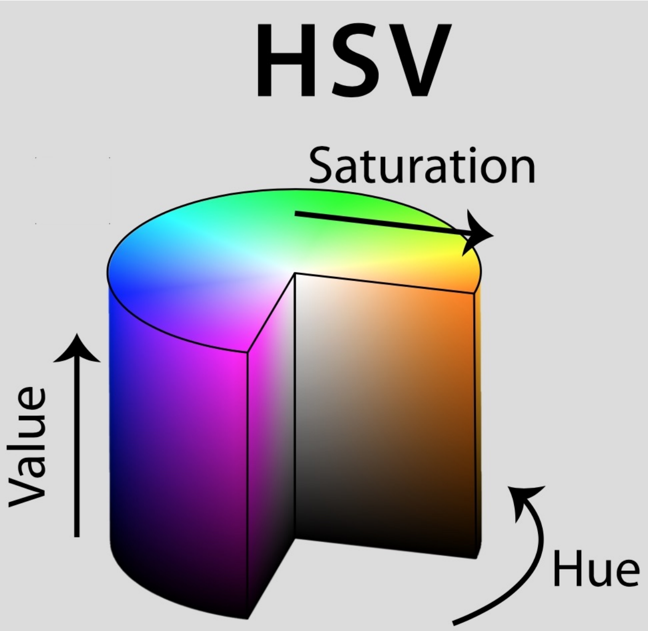

FIGURE 5.11: HSB Cross-Section

Brightness is a relative expression of the intensity of the energy output of a visible light source. It can be expressed as a total energy value (different for each of the curves in the diagram), or as the amplitude at the wavelength where the intensity is greatest (identical for all three curves).

In the RGB colour model, the amplitudes of red, green, and blue for a particular colour can each range from 0 to 100 percent of full brilliance. These levels are represented by the range of decimal numbers from 0 to 255, or hexadecimal numbers from 00 to FF.

5.3 Complexity

5.3.1 Pattern Matching

In our proposed method we have done a sequential search through our database to identify which image represents closest to the target images, thus making the algorithm directly proportional to the total number of images in our database. Complexity for the colour transfer is

$$O(N)$$

- N is the number of images to be processed.

5.3.2 Histogram Transfer

In the Histogram Transfer process we go through each and every pixel of source image. There are 0 to 255 values for R, G and B channel. Since our grayscale image has only Luminance component we retain the Luminance values and tranfer the chrominance histogram of source image to the target image. Thus the complexity for the colour transfer is

$$O(mn)$$

- m is the number of row pixels
- n is number of column pixels.

5.4 Sample Output

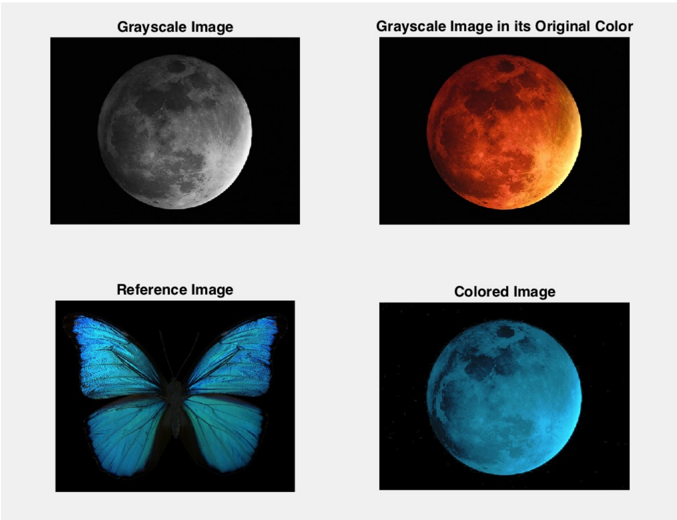

FIGURE 5.12: Colorisation - Example 1

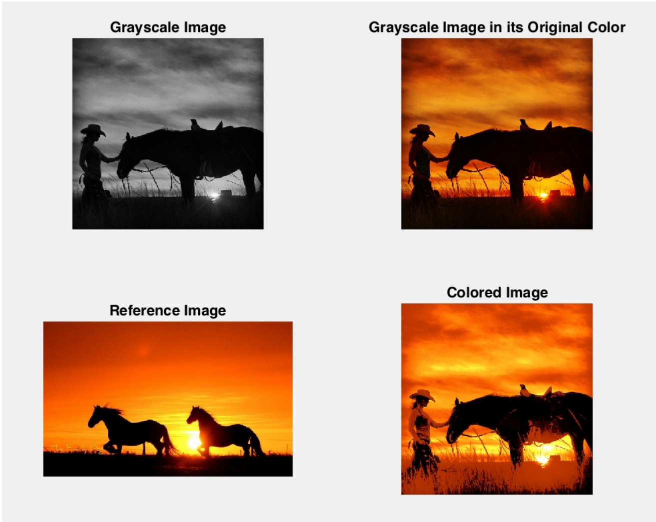

FIGURE 5.13: Colorisation - Example 2

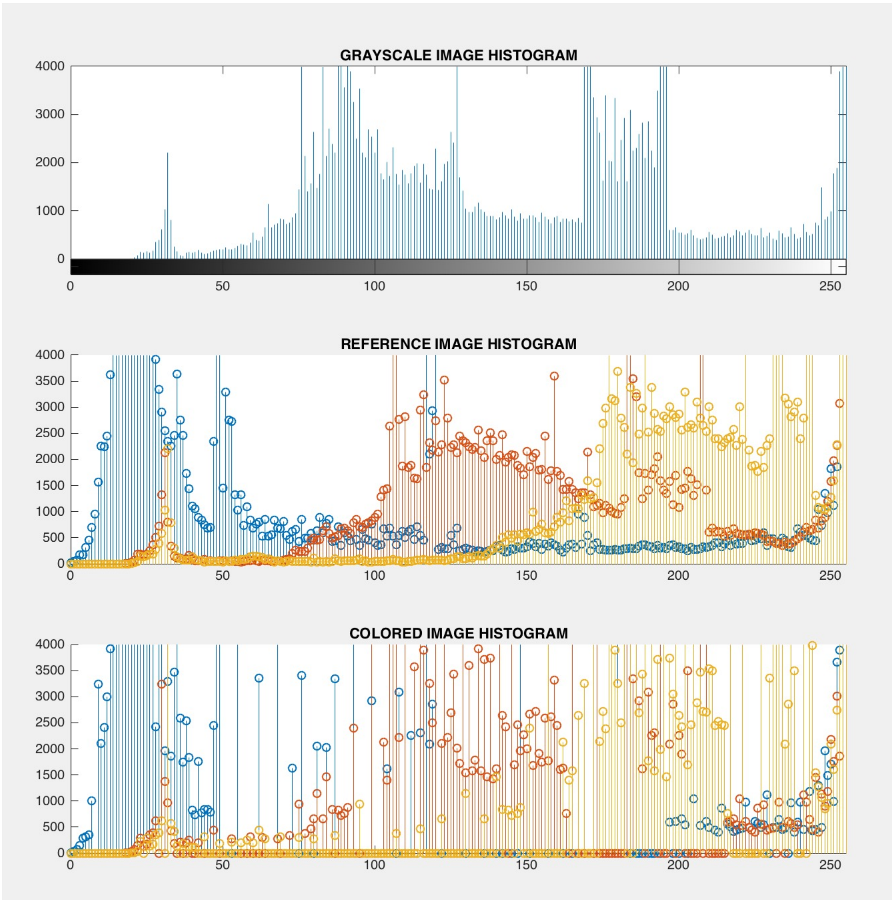

FIGURE 5.14: Colorisation - Example Histogram

CHAPTER 6

Conclusion and Future Work

In conclusion, this algorithm deals with colourisation of grayscale media using reference images. The automated nature of this algorithm is attributed to the fact that it uses a database of reference images, which upon being scanned, returns the closest match to the target image. This is achieved using Pattern Matching and Histogram Equalization. The efficiency is analysed by comparing with other existing techniques and shows promising results.

We can improve this approach by using a method which involves classification of the reference images into different tables of similar images, which are segregated based on their grayscale intensities. Once the set of reference images are provided at the offset, the algorithm automatically clusters them into sets based on the above condition. This is done using Probabilistic Neural Network branch of Machine Learning. This has great scope in Medical studies where computer assistance is demanded due to the fact that it could improve the results of humans in such a domain where the false negative cases must be at a very low rate.

REFERENCES

1. Gonzalez, R.C., Woods, R.E. (1987) 'Digital Image Processing Addison-Wesley Publishing', Vol.3, pp. 230.

2. Levin, P., Leschinski, D., Weis, Y. (2004) 'Colorization using optimization', In Proc. SIGGRAPH'04, pp. 689-694.

3. Liu, X., Wan, L., Qu, Y., Wong, T., Lin, L.S., Heng, P.A. (2008) 'Intrinsic colorization', ACM Trans. Graphic, Vol.27, No.5, pp. 1-9.

4. Luan, Q., Wen, F., Cohen-or, D., Liang, L. et al. (2007) 'Natural image colorization', In Proc. Rendering Techniques'07, pp. 309-320.

5. Nie, D., Ma, L., Xiao, S., Xiao, X. (2005) 'Grey-scale image colorization by local correlation based optimization algorithm', In Proc. VISUAL'05, pp. 13-23.

6. Nie, D., Ma, Q., Ma, L., Xiao, S. (2007) 'Optimization based grayscale image colorization', Pattern Recognition Letters, Vol.28, No.12, pp. 1445-1451.

7. Noda, Hideki, Michiharu Niimi, and Jin Korekuni (2006) 'Simple and efficient colorization in YCbCr color space', 18th International Conference on Pattern Recognition, Vol. 3, pp. 685-688.

8. Welsh, T., Ashikhmin, M., Mueller, K. (2002) 'Transferring color to grayscale images', In Proc. SIGGRAPH'02, pp. 277-280.

9. Xiao, X., Ma, L. (2009) 'Gradient-preserving color transfer', Comput. Graph. Forum, Vol.28, No. 7, pp. 1879-1886.

10. Yatziv, L., Sapiro, G. (2006) 'Fast Image and video colorization using chrominance blending' IEEE Transactions on Image Processing, No. 5, pp. 1120-1129.