Image Decolourisation:

Preserving Perceptual Differences Between Colors

Final Year Project Report

Author: Jiaqi Wang
Supervisor: Prof. Paul Rosin

Abstract

Traditional image decolourisation methods often impair color contrast when converting color images to greyscale images. Gooch et al. proposed Salience-preserving color removal model, which better preserves information such as contrast and brightness of an image during grayscale of color images. However Gooch's model usually performs less well than the traditional greyscale representation when it tries to map a wide range of chromatic gradients to some independently distinguishable levels. To address this shortcoming, an optimization algorithm is proposed that preserves the perceptual differences between colors. The algorithm proposes the concept of "extended neighborhood", adopts the CIE L * a * b * color space and local mapping techniques, and aims to improve detail capture and contrast. Finally, quantitative experimental results verify the feasibility and effectiveness of the proposed algorithm.

Acknowledgments

I would like to extend my deepest gratitude to Professor Paul Rosin for his invaluable guidance and support throughout the course of this research project. His insightful feedback and expert advice have been instrumental in shaping not only this paper but also my broader understanding of image processing techniques. Professor Rosin's commitment to academic rigor and his tireless dedication to his students have inspired me to strive for excellence in my own work. I am truly appreciative of the opportunity to learn from such an esteemed scholar and educator. Thank you, Professor Rosin, for your unwavering encouragement and for helping me realize the potential of this research.

Contents

1. Introduction 5
    1.1 The Problem 6
    1.2 Objectives 7
2. Related Research 8
    2.1 Color Space Based Decolorization Modeling Approach 10
        2.1.1 CIEXYZ Color Space 11
        2.1.2 YCrCb Color Space 12
        2.1.3 CIE Lab Color Space 13
    2.2 Tone Mapping 14
        2.2.1 Global Mapping 14
        2.2.2 Local mapping 19
        2.2.3 Mixing of global and local mappings 21
3. Methodology 22
    3.1 Existing methods 23
        3.1.1 The basic idea of "Color2Gray" 23
        3.1.2 Salience-Persevering Color Removal 24
        3.1.3 Target differences 25
        3.1.4 Optimization 25
    3.2 The proposed algorithm 26
        3.2.1 Introduction to the algorithm 26
        3.2.2 Adoption of local mapping approach 27
        3.2.3 Delta Calculation for Extended Neighbors 28
4. Evaluation and analysis 30
    4.1 Overview 30
    4.2 Subjective quantitative evaluations 31
    4.3 Objective quantitative assessment 32
        4.3.1 Decolorized image metrics: CCPR, CCFPR and E-score 33
        4.3.2 Structural Similarity Index Measure SSIM 35

4.3.3 Evaluation based on CCPR, CCFR, and E-score Metrics 36
4.3.4 Structural Similarity Index Measure (SSIM) Evaluation 40
4.3.5 Comprehensive evaluation of image decolorization 42
5. Conclusion 45
6. Feature work 46
Reference 49

1. Introduction

In the ever-evolving realm of computer science, image processing emerges as a beacon of innovation, shaping the way we visualize, analyse, and interpret the vast array of visual data that constantly surrounds us. Harnessing the power of algorithms and computational techniques, it breathes life into a multitude of applications, ranging from the life-saving realm of medical imaging to the sophisticated intricacies of facial recognition and the vigilant eye of surveillance systems [1, 2]. It involves the processing and analysis of images to extract meaningful information, improve image quality and facilitate interpretation. Image processing techniques have become more sophisticated with advances in technology and computing power, enabling us to extract more value from images.

In some cases, images are converted into greyscale images for ease of processing or compatibility with certain systems. This process is known as "image decolorization", which focuses on mapping colors into different shades of grey based on luminance, thus converting a color image into a single-channel image [3]. It has a wide range of application scenarios in daily life, such as black-and-white printing [4], e-ink technology (Kindle reader), black-and-white scanning and so on. In the field of image processing, grey-scale maps are also often used in non-realistic image rendering and single-channel image processing, such as image recognition and edge detection. The reason for this is that intensity typically captures much of the visually important information presented in the color counterpart [5]. Since decolorization reduces the dimensionality of the input signal, it inevitably leads to information loss. The purpose of decolorization is to maintain as much information as possible about the original color image while providing perceptually satisfactory greyscale outputs [6, 7].

1.1 The Problem

Image decolorization is the process of converting a color image of 3D data into a greyscale image of 1D data. When an image is decolorised, the data undergoes a dimensionality reduction, which inevitably results in the loss of detail and structure in the color image [6, 7]. The widely used color-to-grey scale image transformation method in image processing is to take the Y channel in the chroma i.e. YUV color space or the L channel in the luminance [8, 9], i.e. CIE Lab space as the greyscale image [6]. Neither of them considers the correlation between the pixels, so the contrast information is lost when the image is decolorized, resulting in the loss of details and features in the grey scale map. loss of details and features in the grey scale map. As shown in Figure 1, the blue numbers 2 and 5 cannot be distinguished in the grey scale map in the results of Y and L channels. Therefore, it is very necessary to study the decolorization algorithm.

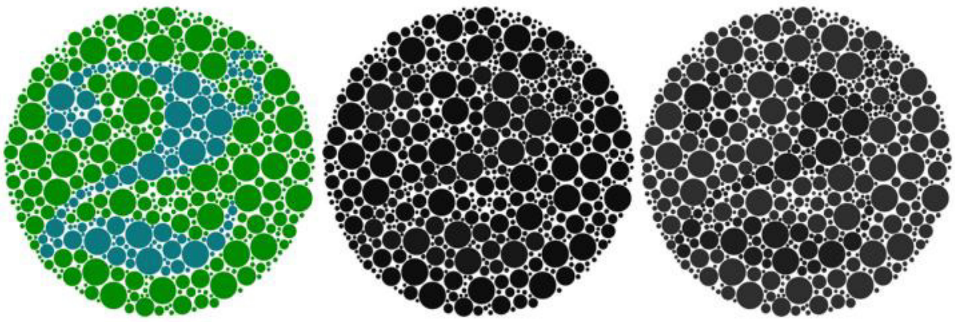


Figure 1: Left: original image; Middle: Y-channel image; Right: L-channel image

In summary, image decolorization of color images inevitably results in greyscale images, and although decolorization works well in many practical situations, intuitive methods such as extracting the luminance channel in the CIE Lab color space [9] do not explicitly capture important appearance features and tend to reduce saliency structures. In this process, how to present the original chroma, luminance and color saturation of color images in greyscale solid images is the focus of decolorization research. Therefore, how to maintain the brightness and contrast information becomes the core of decolorization research, which is what we call contrast maintenance. In order to maintain color contrast, some recent methods for color-to-grey conversion impose constraints on spatial intensity variations and require that the grey scale contrast be similar to that of the color input.

1.2 Objectives

The overall goal of this research is to refer to and improve the Color2Gray [10] party, and to develop an improved image decolourisation method that preserves as much of the perceived difference between colours as possible during the conversion of colour images to greyscale images. This goal stems from the problem observed in the field of image processing, where key details are lost during the conversion process, thus affecting the analysis and interpretation of the resulting image.

The specific objectives of this study are as follows:

To study the mechanisms of information loss: this research will first provide insight into the mechanisms by which important information is lost during decolorization. It will investigate how color channels in an image are converted to greyscale and why certain features become indistinguishable when color information is removed. This objective will provide the basis for identifying key areas where current decolorization techniques are inadequate and where improvements can be made.

Examination of existing techniques: an extensive examination of existing decolorization techniques will be undertaken. The study will assess how these techniques work, what image features they are able to retain, and where they fail. Understanding the current state of current decolorization methods will help to identify opportunities for innovation and improvement.

Improvement of the decolorization method: Based on the investigation of the problem and the study of existing techniques, the research aims to improve a new method of image decolorization. The focus of this method of image decolorization is to effectively reduce the loss of critical features during the conversion process and to preserve image characteristics.

Evaluation of Existing Methods: Upon completion of our method, a comprehensive evaluation will be conducted to assess its performance using quantitative assessments. We will compare our method with existing techniques using a variety of images and scenes. Evaluate the extent to which our method preserves important image features.

2. Related Research

The process of converting color images to greyscale (known as decolorization) is a widely used tool in a variety of image and video processing applications, including digital printing and stylised black and white photography. Despite its importance, conventional decolorization methods usually fail to preserve the original color contrast, resulting in significant loss of information. Some of the research on greyscale algorithms has focused on preserving the chromatic aberration, color contrast, etc. of the image while retaining the brightness information of the image from the perspective of maximising the preservation of the original image information. In recent years, a large number of novel decolorization algorithms have been proposed, such as using the Y channel of the CIE XYZ color space [9], Lu, et al. [11] proposed a contrast preserving decolorization algorithm, Gooch, et al. [10] proposed a "color2gray" color image greyscale algorithm, and Bala and Eschbach [12] proposed a spatial method for converting from color to greyscale, and so on.

Currently, greyscale algorithms can be classified as global mapping methods, local mapping methods and mixed methods. In addition, there are some image decolorization techniques that use the color space for decolorization, but these techniques do not belong to the traditional mapping function paradigm. Table 1 lists some representative classical methods in each category.

Table 1: Some methods of image decolorization

<table>
<tr><th>Section</th><th>Author</th><th>Method</th></tr>
<tr><td rowspan="4">Global mapping method</td><td>Gooch [10]</td><td>Salience-preserving color removal</td></tr>
<tr><td>Rasche [13, 14]</td><td>Detail preserving reproduction of color images for monochromats and dichromats

Re-colouring images for Gamuts of lower dimension</td></tr>
<tr><td>Grundland [15]</td><td>The decolorize algorithm for contrast enhancing, color to grayscale conversion</td></tr>
<tr><td>Kim [16]</td><td>Robust color-to-gray via nonlinear global mapping</td></tr>
<tr><td rowspan="2">Local mapping method</td><td>Bala [12]</td><td>Spatial color-to-grayscale transform preserving chrominance edge information</td></tr>
<tr><td>Smith [17]</td><td>Apparent greyscale: A simple and fast conversion to perceptually accurate images and video</td></tr>
<tr><td rowspan="2">Local mapping and global mapping</td><td>Lu [11]</td><td>Contrast preserving decolorization</td></tr>
<tr><td>Lu [18]</td><td>Real-time contrast preserving decolorization</td></tr>
<tr><td rowspan="2">Color Space</td><td rowspan="2"></td><td>CIEXYZ [9]</td></tr>
<tr><td>YcrCb [19]</td></tr>
</table>

		CIELab [20]

First I will describe three methods of color space based decolorization.

2.1 Color Space Based Decolorization Modeling Approach

Color Space is also known as the color gamut [21]. In color science, people have established a variety of models used to portray the color space, usually one-dimensional, two-dimensional, three-dimensional or even four-dimensional space to specifically portray the color, the color range that can be defined by this structure is the color space. At present, the more commonly used color space includes: CIEXYZ color space [9], YCrCb color space [19], CIELab color space [20]. The color space referenced in the methodology of this thesis, the CIELab color space, is the aforementioned color space that I will introduce below.

First of all, a few basic concepts:

- Brightness (lightness or intensity or luminance): is the role of light in the human eye caused by the brightness of the feeling, it is related to the luminous intensity of the object being observed, mainly to show the light of the strong and weak;

- chromaticity (hue): is when the human eye to see one or more wavelengths of light produced by the color feeling, it reflects the type of color, is to determine the basic characteristics of color;

- Color saturation (saturation): refers to the purity of a color, i.e., the degree of adulteration of white light, indicating the degree of color shades.

Traditionally, it is believed that the luminance information mentioned above can be used as the reference result of color image decolorization, however, this operation will have many shortcomings in practical application, in order to make the problem clear, next, we specifically introduce three typical color spaces based on the decolorization method.

2.1.1 CIEXYZ Color Space

The simplest and most widely used method for converting color to grayscale involves utilizing the luminance channel, thereby ignoring the chrominance channels. This technique serves as a grayscale representation of the original color image. One such method employs the Y channel of the CIE XYZ color space [9]. In this method, after converting the color image to the XYZ space, the Y value is taken as the grayscale value. This approach is straightforward and computationally efficient. However, it may not be suitable for all images, especially those with colors of equal luminance. Since it only considers the luminance channel information, it struggles with grayscale conversion of different colors that share the same luminance value. In practice, this can lead to loss of detail or differences between colors in a greyscale image that would be apparent in a color image.

Referring to Figure 2, it can be seen that relative to the original image, the image exported by the method of decolorization through the Y channel of CIE XYZ lacks a lot of image information, for example, the numbers 2 and 5 in the circular image are almost impossible to find in the grey scale image. In another set of comparison images, the color images have obvious color differences, while the CIE_Y images lose the color contrast. Therefore, the method of decolorization using the Y channel of CIE XYZ does not work well.

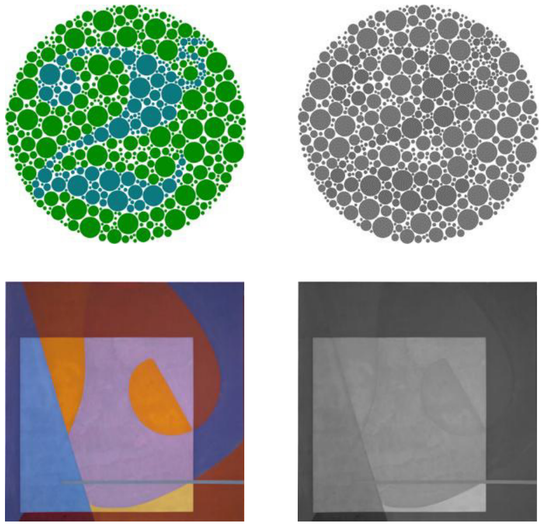


Figure 2: On the left is the original image, on the right is the image after greyscale conversion of the Y-channel of CIE XYZ

2.1.2 YCrCb Color Space

The YCrCb color space, where Y stands for luminance, and "Luminance" is established from the RGB input signal by superimposing specific portions of the RGB signal [19]. The term "chroma" refers to two qualities of color: chroma and color saturation, which are represented by the letters Cr and Cb, respectively. Cr represents the difference between the red component of the RGB input signal and the RGB signal's luminance value, whereas Cb represents the difference between the blue part of the RGB input signal and the RGB signal's luminance value.

The YCrCb color space has a conversion relationship with the RGB color space as follows:

$$\begin{cases} Y = 0.299R + 0.578G + 0.114B \\ Cr = (0.500R - 0.4187G - 0.0813B) + 128 \\ Cb = (-0.1687R - 0.3313G + 0.500B) + 128 \end{cases}$$

Or write it in the form of a matrix:

$$\begin{bmatrix} \mathrm{Y} \\ \mathrm{Cr} \\ \mathrm{Cb} \end{bmatrix} - \begin{bmatrix} 0.299 & 0.578 & 0.114 \\ 0.500 & -0.418 & -0.0813 \\ -0.1687 & -0.3313 & 0.500 \end{bmatrix} \begin{bmatrix} \mathrm{R} \\ \mathrm{G} \\ \mathrm{B} \end{bmatrix} + \begin{bmatrix} 0 \\ 128 \\ 128 \end{bmatrix}$$

The transformation relationship between RGB color space and YCrCb color space can be written in the following form:

$$\begin{bmatrix} \mathrm{R} \\ \mathrm{G} \\ \mathrm{B} \end{bmatrix} = \begin{bmatrix} 1 & 1.4020 & 0 \\ 1 & -0.7141 & -0.3441 \\ 1 & 0 & 1.7720 \end{bmatrix} \begin{bmatrix} 0 \\ \mathrm{Cr} - 128 \\ \mathrm{Cb} - 128 \end{bmatrix}$$

2.1.3 CIE Lab Color Space

The CIE chromaticity model is one of the first models used by the International Commission on Illumination [21]. It is a three-dimensional model, which, x and y two-dimensional definition of color, the third dimension defines the luminance. CIE Lab color space is the CIE in 1976 for non-self-illumination of the color space, called CIE1976 L * a * b *, or CIE Lab. Lab using b *, a * and L * to establish the axes to define the CIE color space. Among them, the L* value represents the luminance, its value from 0 (black) to 100 (white). b* and a* represent the chromaticity coordinates, of which a* represents the red-green axis, b* represents the yellow-blue axis, their values from 0 to 10, a* = b* = 0 means no color, so L* represents the scale factor from black to white [22]. The concept of opposing color coordinates (opponent color coordinate) stems from the idea that a color cannot be both red and green or yellow and blue at the same time, but can be conceived of as a mixture of red and yellow, red and blue, green and yellow, and green and blue [23]. Any bit of color in nature can be expressed in the CIE Lab color space, which is larger than the RGB color space. Furthermore, regardless of the technology, this mode digitally describes the human visual experience. As a result, it compensates for the fact that the RGB color space must be dependent on the color properties of the device. Because the CIE Lab color space is bigger than the RGB color space, any color information that can be expressed by the RGB color system may be shaded in the CIE Lab color space.

2.2 Tone Mapping

Before introducing the methods, we introduce the classification of tone mapping [24] colors with the advantages and disadvantages of the classification methods. Tone mapping methods can be roughly divided into two categories: global methods and local methods. In image processing, the choice between global and local mapping can have a significant impact on the final result. Global algorithms process each pixel uniformly based on its intrinsic characteristics, often yielding results suitable for images with limited depth (e.g., 12-bit), but may flatten images with a wider dynamic range and wash out complex details. In contrast, local mapping takes into account the spatial context of each pixel, ensuring adaptive processing [25]. Although this approach requires higher computational power due to the meticulous processing, it does a good job of preserving and highlighting local details, especially highlights and shadows. While global mapping provides simplicity, the context-aware transformation of local mapping provides a more nuanced image rendering [26].

2.2.1 Global Mapping

Several methods of global mapping are described below:

Researchers such as Gooch, et al. [10] proposed a grayscale method for color images called "Color2Gray". They noticed that when converting a color image to grayscale, because the luminance and color are often equal, it is easy to cause the loss of chromatic aberration information. To address this problem, the algorithm aims to preserve color gradients as much as possible. The method involves comparing the grayscale and color differences between two pixels and trying to maintain the color differences as much as possible in a least squares sense.

The results of Color2Gray provide users with important information that is missing in traditional grayscale image generation methods. Although the Color2Gray method does a better job of maintaining the color contrast of the original image, it may not be able to maintain both luminance and chrominance information when dealing with colorful images. When the algorithm attempts to map a wide range of chromaticity gradients to some independently distinguishable levels, it typically performs less well than a traditional grayscale representation. For example, the first row of Figure 3 shows this limitation. For natural scenes with large variations in brightness, Color2Gray does not provide significant improvement. However, for images with large areas, similar brightness, and only a few different chromaticity values, this method can significantly improve the results. For example, in the second row of Figure 3, Color2Grey succeeded in making the two hills behind the car clearly distinguishable. The Color2Gray optimization process has a computational complexity of $O(N^4)$, which makes outputting the image slow due to the fact that the size of the matrix involved is equal to the square of the number of pixels in the image. The amount of computation can be reduced by sampling a smaller neighborhood of pixels. This report refers to that method, however, this report improves on the method used and the output is better compared to Gooch's method.

Color Source Photoshop Gray Color2Gray

Figure 3: Images with a variety of colors lead to a Color2Gray result, which

does not provide a great improvement over Photoshop greyscale.

The global optimization approach was proposed earlier by Rasche with two related papers [13, 14]. The paper [14] called "Detail preserving reproduction of color images for monochromats and dichromats" defines constraints directly on different color pairs to achieve accelerated linear color mapping. Their strategy aims at maintaining the stability of contrast and luminance at the same time. The error function is set based on matching the gray scale difference with the corresponding color difference. The central goal of the method is to find the optimal color conversion by minimizing this error function. The linear method used in this thesis for monochrome and dichromatic color image reduction has several advantages: the size of the dataset can be reduced quickly and simply using color quantization without manipulating the entire dataset, unlike traditional nonlinear methods. Linear operations are very fast to compute and easy to implement. In contrast, processes like LLE require finding the eigenvectors of very large sparse matrices [27]. It is easy to incorporate other perceptual distance metrics. The linear approach of this thesis does not require tuning, such as the neighborhood size parameter often found in nonlinear methods. See Figure 4 for a demonstration of the results of applying this method to an image in grayscale (for the pigment defect case). Figure 4a shows the original image, Figure 4b shows the original image as seen by a simulated observer with pigmentation defects, and Figure 4c shows the original image after recoloring.

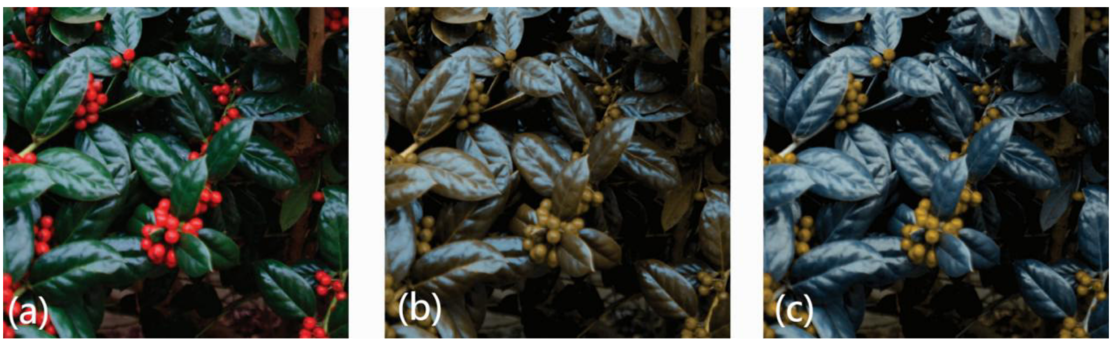


Figure 4: In situ defects: (a) the color image in Figure 1, (b) the image seen by the simulated in situ observer, and (c) the image seen by the simulated in situ observer recolored for the in situ observer.

The paper [13] called "Re-colouring images for gamuts of lower dimension" obtained a similar optimization function between all the color groups, and obtained the sampled colors by color quantization, tried to keep the greyscale differences between the sampled colors as colorful as possible, and obtained the greyscale values of all the colors by the difference, their methodology aims to keep the contrast while maintaining a consistent brightness. The goal is to minimize the error function to find an optimal conversion. But true color images because of the large color sample size linear interpolation sometimes does not give good results. As shown in Figure 5, there is not enough perceptual bandwidth to map each blade of a windmill to a distinct shade of grey in a windmill picture that has too many color variations.

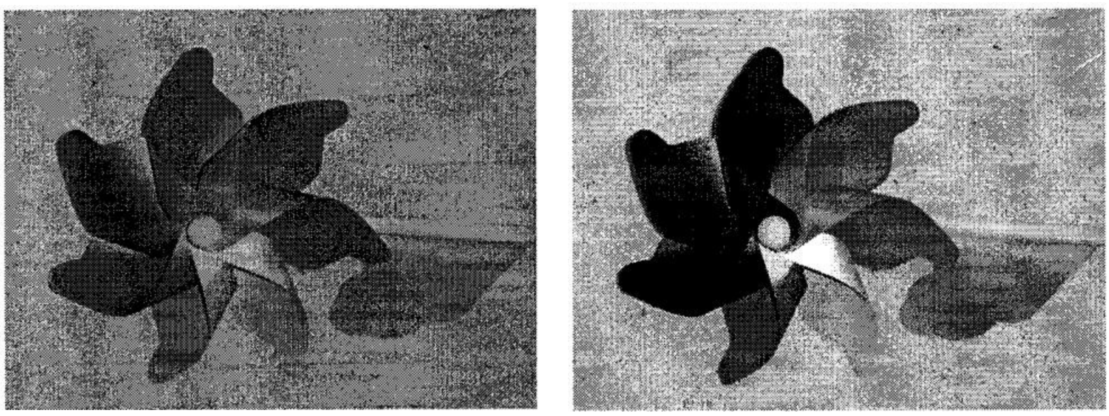


Figure 5: Too many color variations: there isn't enough perceptual bandwidth here to map each windmill blade to a separate shade of grey.

Grundland and Dodgson [15] proposed a decolorize algorithm for contrast enhancement and for converting colors to grayscale. It incorporates innovative techniques for image sampling and dimensionality reduction, color difference sampling using Gaussian pairing, and color difference analysis via principal component analysis. This algorithm not only enhances the contrast of an image, but also performs grayscale conversion, i.e., converting a color image into a grayscale map while adding a certain amount of chromaticity in terms of luminance. Their approach is to achieve global grayscale conversion by representing grayscale as the RGB primary color and its saturation, combined with a segmented linear mapping associated with the image. In addition, They employ three parameters to regulate contrast enhancement, color selection, and noise suppression, and have set image-independent default values for these parameters. The benefits of this technique include continuous mapping, global consistency, and grayscale preservation, in addition to speed and simplicity, as well as predictable brightness, saturation, and hue ordering features. Referring to Figure 6, it can be seen that the example processed image in this decolorization algorithm enhances contrast by increasing luminance to reflect chromatic aberration.

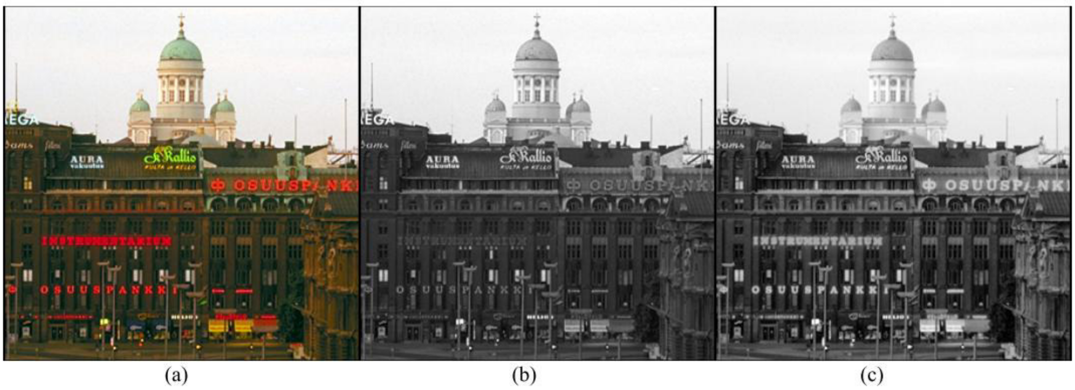
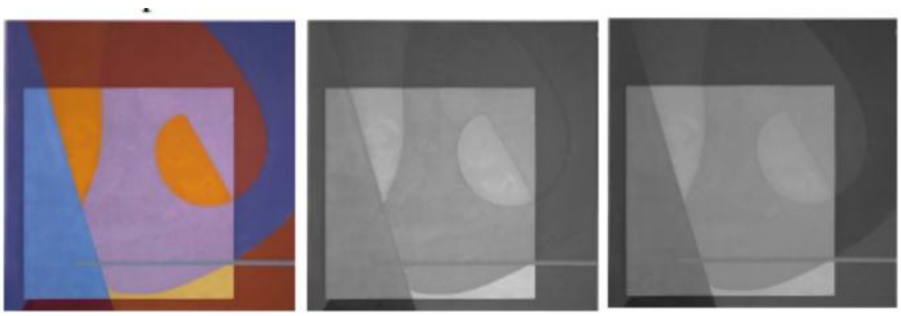


Figure 6: Recovering chromatic contrasts in grayscale:(a) color image, (b) luminance channel, (c) Their enhanced grayscale.

Kim, et al. [16] proposed a more flexible nonlinear mapping in the CIE LCH color space similar to the Fairchild model for measuring the H-K(Helmholtz-Kohlrausch) effect to achieve global color contrast enhancement [16]. The CIE LCH color space is used, which is divided into three channels, L (Luminance) for luminance, C (Chrome) for saturation, and H

(Hue) for chromaticity information. A nonlinear mapping function is established on this color space, in which an important basis is to weight the value of chromaticity change according to the information of saturation, the higher the saturation, the larger the range of chromaticity can be changed, and vice versa, the smaller it is. Although the method can be applied to nonlinear mapping in video transformations, the method causes the image to lose or blur edge features. In the CIE LCH color space, the chroma and luminance information are not independent, which produces a large variation of gray and dark colors, but a small variation of bright colors instead, and it will not give good results.

The color to greyscale conversion in [16] does not explicitly consider features for global contrast, but focuses more on local features. Compared to the method of [10] which may lose smaller feature details, the method of [16] retains more image feature details. See Figure 7 for a graphical representation of the results of this algorithm compared to Gooch's algorithm.



(a)Input (b)Kim (c)Gooch

Figure 7: Comparison of color to grayscale image conversion results. (a) Input image. (b) Kim, et al. [16]. (c) Gooch, et al. [10].

2.2.2 Local mapping

The local mapping-based color removal method focuses on the distribution differences of colors in local spatial locations, and adjusts the pixel intensities at local locations according to the local distribution differences, which can better preserve the local color features of the original color image. For local mapping, there are several methods as follows:

Smith, et al. [17] proposed a decoloring algorithm for converting complex images and videos into perceptually accurate grayscale versions. Their use of a two-step approach starts with assigning grayscale values globally and employing the Helmholtz-Kohlrausch color appearance effect for color ordering, and secondly, extracting edge information using Laplace pyramids for local contrast enhancement. See Figure 8, where Smith, et al. [17] method was applied to two images from Gooch, et al. [10] to demonstrate their ability to distinguish between colors of equal luminance.

Original CIE_Y Smith

Figure 8: Original image on the left, CIE Y-channel grayscale with essentially unsharpened enhancement in the middle, and Smith's method on the right.

And Bala and Eschbach [12] proposed a spatial-based color-gray scale transformation technique. This method preserves color edges locally by adding high-frequency color information to the luminance channel. The final effect is obtained by applying spatial high-pass filtering to the color channel and weighting the output with a luminance-dependent factor, and then combining this output to the luminance channel. As shown in Figure 9, this algorithm helps to recover the chromatic details lost in a standard grey scale (L*) image, but the local algorithm enhances all the color differences in the image, and along with the image detail enhancement, it enhances some of the defects at the same time.

(a) (b)

Figure 9: (a) standard and (b) spatial color to greyscale mapping

The greyscale algorithms proposed by Bala and Eschbach [12] and Smith, et al. [17]. both give a greyscale algorithm that enhances local features by obtaining high-frequency information through high-frequency filtering of the chroma layer saturation layer, and then multiplying it by a luminance-related weight, which is then added to the luminance layer. Although the local mapping method can find out the local color difference of each channel of the color image, so that the decolorized image can better retain the local details of the original color image. However, most of the local mapping algorithms cannot guarantee the overall consistency of the color, i.e., the same color is mapped to the same gray value in the greyscale process of the color image. Moreover, artifacts will appear in the image after decolorization using local mapping algorithms.

2.2.3 Mixing of global and local mappings

In addition to global and local mapping, there are also methods that mix the two: Lu, et al. [28] proposed an optimization method based on maximizing the preservation of original color contrast, i.e., after global mapping using finite multivariate polynomials, the bimodal contrast is preserved by solving a bimodal contrast-preserving objective function based on weakening the constraints of the color order restrictions. This method preserves and enhances the image contrast to some extent, but the time complexity is limited by the values of the weights in the global mapping, so later they proposed a real-time contrast-preserving decolorization method by linearly combining the color channels with three parameters, discretizing the solution space of the parameters, and scaling the high-resolution input image to a fixed size [18]. Although this algorithm has some advantages in contrast preservation and speed, its grayscale results occasionally still have the phenomenon that the color consistency features of the image are corrupted, losing some important contrast information, and they did not consider the problem of grayscale pixel features.

This chapter discusses how greyscale results can be improved by using the theory of color vision from three starting points: global mapping [10, 13-16], local mapping [12, 17] and mixed methods [18, 28], respectively. Although the Gooch, et al. [10] method is computationally expensive for computing the differences between pixel pairs, it may transfer distinct colors of neighbouring pixels to different greyscale values. Therefore, I propose an improved color to greyscale algorithm, based on Gooch, et al. [10].

3. Methodology

This chapter focuses on the type of tone mapping to be used in this paper, the proposed new color image decolorization method, which covers specifically the existing methods and theories on which this paper relies and the theoretical model and methodology of this paper. The theoretical model of this paper is carried out on the basis of Gooch's image decolorization method, we change the target difference function in its decolorization model, calculate the difference between each pixel and its neighboring pixels and non-neighboring pixels, and use the nonlinear function crunch to process the chromaticity difference and save all the differences into a sparse matrix, so as to obtain an image decolorization method with better results than the previous methods.

3.1 Existing methods

3.1.1 The basic idea of "Color2Gray"

When a color image is converted to greyscale, key visual features of the image may be diminished. In the paper of Gooch, et al. [10], an algorithm called "Color2Gray" is presented that aims to reduce the loss of these key features in color images. The basic premise of the algorithm is to compute signed chromatic distances in the CIE L * a * b * chromaticity plane. In this way, the chroma and luminance variations of the original image are transposed to a greyscale version, resulting in a target image that captures and preserves the salient features of the original color image.

The algorithm consists of three key steps:

(1) Converting the input RGB image into a perceptually uniform CIE L * a * b * color space. This representation is characterized by the L * a * b * color space, which has three components:

L: represents the luminance of the image. Luminance values are between 0 and 100, where 0 represents black and 100 represents white. larger values of L indicate higher luminance.

a: This component goes from green to red. The value can vary from negative to positive, e.g. -128 to +127 indicates a transition from green to red. b: Indicates the transition from blue to yellow.

b: This represents the spectrum from blue to yellow, and its value can also vary from negative to positive, with -128 indicating blue and +127 indicating yellow.

An important feature of this transition is its perceptual consistency. Essentially, even if the L*a*b* values change by the same amount, the visual perceptual differences remain relatively consistent.

(2) Create target grayscale images: create grayscale target differences using adjacent pixel chromaticity differences and luminance differences.

(3) Optimize the transformation: selectively adjust the luminance difference of the original image to affect the chrominance difference of the original image using the least squares method.

3.1.2 Salience-Persevering Color Removal

Color removal of a color image is the process of converting a color image into a grayscale image, in which we hope that the color contrast of the original color image can be maintained in the new grayscale image by the luminance and chromaticity information as well. According to this purpose, the color difference between pixels in a color image is expressed as a set of signed scalar values, and then a grayscale image is constructed using these values. For each pixel labeled i and its neighboring pixels j, a different scalar value is determined based on their luminance and chromaticity differences δ_{ij}. These scalar values subsequently affect the composition of the grayscale image, denoted by g. The difference in grayscale between a pixel and can be described by $(g_i - g_i)$. The optimal conversion from color to grayscale is to ensure that all these $(g_i - g_i)$ differences are consistent with their respective δ_{ij} values. Therefore, we need to first define what is called the target difference [10].

3.1.3 Target differences

To define the target difference δ_{ij}, we compared the luminance difference ΔL_{ij} and chromaticity difference $\overrightarrow{\Delta C}_{ij}$ in the CIE Lab color system, referred to as (ΔA_{ij}, ΔB_{ij}). Since ΔL_{ij} is a scalar and $\overrightarrow{\Delta C}_{ij}$ is a two-bit vector, we compute its Euclidean norm to map to one dimension and then assign sign to $\overrightarrow{\Delta C}_{ij}$ since it is always positive. The parameter θ controls which color differences are mapped to increases or decreases in the luminance value of the light source, and the chromaticity difference can be calculated by the following equation:

$$\overrightarrow{\Delta C}_{ij} \cdot (\cos\theta, \sin\theta)$$

The target difference is defined as luminance difference or chromaticity difference and δ_{ij} is defined as follows:

$$\delta(\alpha,\theta)_{ij} = \begin{cases} \Delta L_{ij} & |\Delta L_{ij}| > \mathrm{crunch}(||\overrightarrow{\Delta C}_{ij}||) \\ \mathrm{crunch}(||\overrightarrow{\Delta C}_{ij}||) & \overrightarrow{\Delta C}_{ij} \cdot (\cos\theta, \sin\theta) \geq 0 \\ \mathrm{crunch}(-||\overrightarrow{\Delta C}_{ij}||) & \text{otherwise} \end{cases}$$

Crunch is defined as:

$$\mathrm{crunch}(x) = \alpha \cdot \tan(x/\alpha)$$

where α is the amount of chromaticity change that controls the luminance value applied to the light source [10].

3.1.4 Optimization

The algorithm solves an optimization problem that selectively adjusts the grayscale representation according to the chromaticity variation of the source image. This step ensures that the perceived color differences are preserved in the grayscale output. Where g_i and g_j are the grayscale values of the neighboring pixels and GII is the target difference in grayscale between these pixels computed using the aforementioned nonlinear function. The goal of the optimization problem is to minimize the squared difference between the actual grayscale values of adjacent pixels and the target grayscale value [10]. This helps to achieve the desired grayscale representation while preserving the color information. The objective function for minimization is defined as follows:

$$f(g) = \sum_{(i,j)\in K} \left((g_i - g_j) - \delta_{ij}\right)^2$$

Where K is the set of all pixel pairs to be compared.

3.2 The proposed algorithm

3.2.1 Introduction to the algorithm

The main goal of our algorithm is to refer to and improve the Color2Gray[10] method, which converts color images to greyscale while trying to maintain the perceived differences between colors. The algorithm consists of the following three steps:

(1) Color to grayscale conversion using CIE L * a * b * color space.

(2) Calculate the difference between each pixel and its neighboring and non-neighboring pixels under CIE L * a * b * color space, including chromaticity difference and luminance difference. The method of incremental calculation of extended neighbors is proposed.

(3) Process the chromaticity difference to get the greyscale target difference by the nonlinear function crunch(x).

(4) Referring to Color2Gray's method of selectively adjusting the greyscale

representation according to the chromaticity variation of the source image, the optimisation problem is solved to adjust the final greyscale map.

3.2.2 Adoption of local mapping approach

The algorithm proposed by Gooch, et al. [10]. uses the global mapping method, while my algorithm uses local mapping, which has the following advantages over global mapping:

- Adaptive transformation: unlike the global approach which applies a fixed formula to all parts of the image, local mapping adapts to the specific features of each region. This ensures that each part of the image, whether it is shaded, highlighted or has a different color gradient, receives an appropriate grayscale representation.

- preserves perceptual differences: since the transformation is localized, color variations that may be perceptible in one region of the image (but not in others) can be more accurately converted to grayscale. This produces a grayscale reproduction that is more faithful to human perception.

- Enhanced Detail and Contrast: Local mapping amplifies subtle differences between neighboring pixels, ensuring that no intricate detail is lost in the conversion process. This typically results in the generation of grayscale images with better contrast.

Combined with the local mapping approach, my algorithm takes full advantage of local mapping by meticulously considering the luminance information around each pixel, ensuring that the resulting image not only retains its intrinsic contrast, but also vividly demonstrates subtle details that are often overlooked by the global approach.

3.2.3 Delta Calculation for Extended Neighbors

One of the main drawbacks of existing grayscale conversion methods lies in the limited spatial analysis of pixels. Conventional techniques usually evaluate only the nearest neighbors of a pixel to calculate its grayscale value. For example, Gooch's method only considers luminance differences and chrominance differences between neighboring pixels. The resulting grayscale image has low contrast due to the loss of some information in the color image. Our study aims to address this limitation by introducing an "extended neighbor" approach in the incremental computation of grayscale conversion. Instead of considering only neighboring pixels, our method extends the analysis to consider a wider neighborhood around each pixel.

Definitions:

Neighborhood Definition: for each pixel, a "neighborhood" is defined, which includes not only near neighboring pixels, but also pixels further away.

Chromaticity Difference: The color difference between a pixel i and another pixel j within its extended neighborhood, denoted as $\Delta \mathrm{C}_{ij}$.

Methodology:

Our novel approach extends pixel analysis beyond the direct neighborhood by incorporating the concept of extended neighborhood. This extended analysis helps to capture the original color information in more detail, thus enriching the grayscale conversion process.

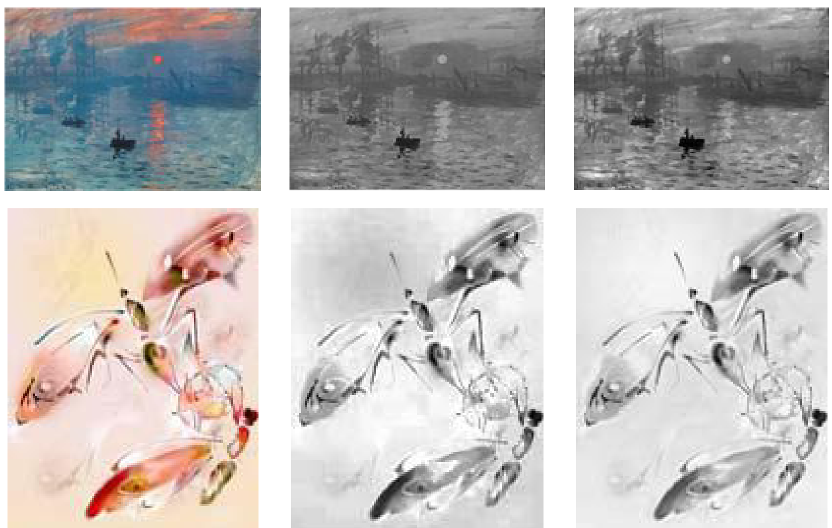
Difference Calculation:

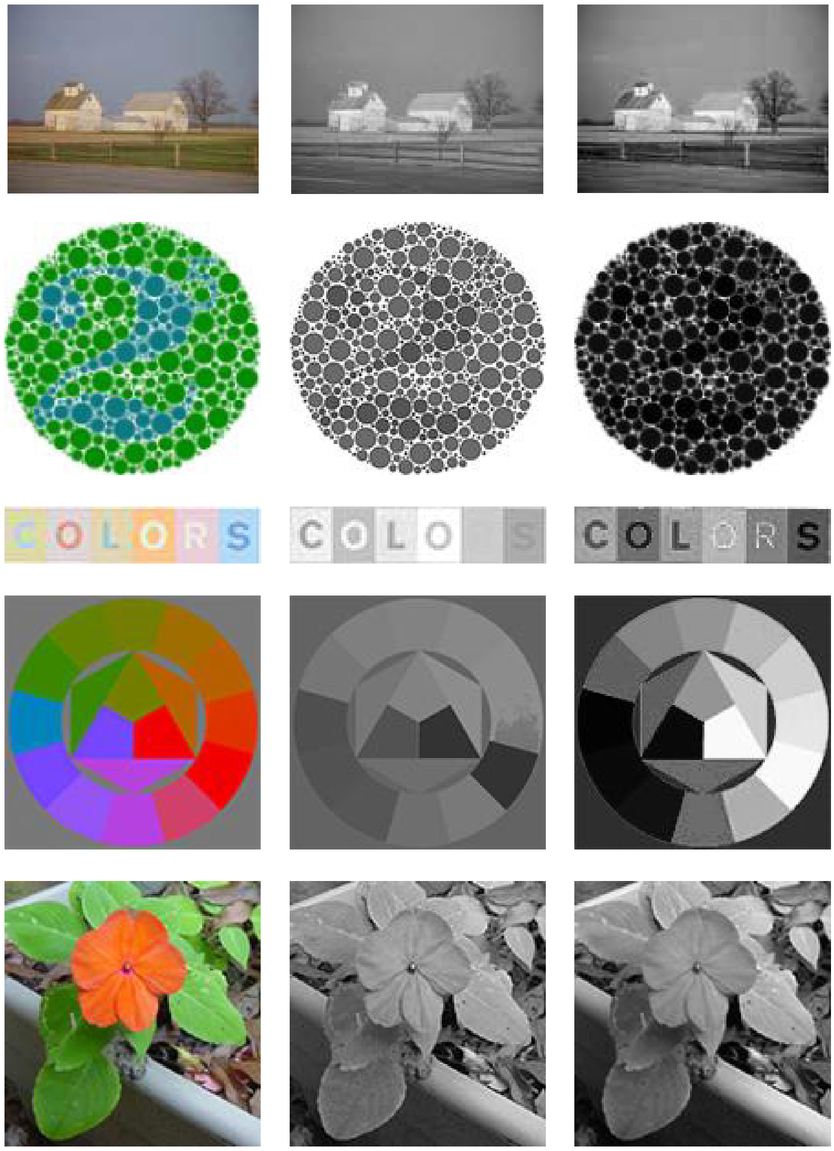
For each pixel i calculate the relationship between the chromaticity difference $\Delta \mathrm{C}_{ij}$ and the luminance difference $\Delta \mathrm{L}_{ij}$ between that pixel i and other pixels j in its extended neighborhood.

3.3 Result

Figure 10 compares the greyscale patterns of the Color2Gray algorithm and our algorithm on various images. In the image in row 5, the low color contrast letter R disappears from the Color2Gray results. The results of our algorithm sometimes retain more detailed features of the image than the Color2Gray algorithm, e.g., the clouds in the sky are noticeably whiter (line 1), the roofs are noticeably darker, and the trees are more clearly distinguished from the houses next to them (line 3).

However, our method is not universally superior; it exhibits limitations in regions with subtle color transitions, such as the number "2"'s in the fourth row of Figure 10 which are not evident in the image. In addition, both algorithms have limited efficacy when dealing with high dynamic range scenes with large fluctuations in brightness, as exemplified in the seventh row. However, our algorithm outperforms Color2Gray in processing images with equal luminance and little change in chroma, and this ability is best demonstrated in row six, where our technique achieves a more accurate representation of chroma so that each color in the chroma table is clearly distinguishable.





original image Color2Gray Ours

Figure 10: Comparison of color source images, Color2Gray results and Our results.

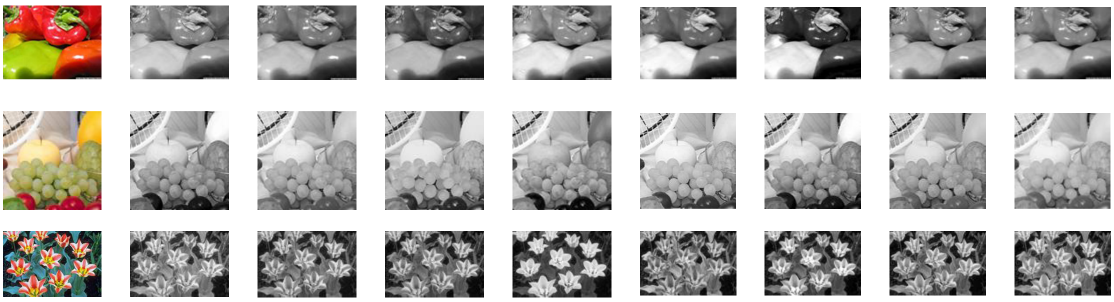
4. Evaluation and analysis

4.1 Overview

In this section, we rigorously assess our proposed decolorization algorithm using both subjective and objective quantitative evaluations. We employ a diverse set of 14 images from the Ĉadík [29] datase, which offer a rich variety of structural textures and color information. Quantitative evaluation of the performance of our algorithms, we compare it against seven widely recognized decolorization methods, namely: Bala [12], CIE Y [9], Color2Gray [10], Grundland [15] , Kim [16], Rasche [13], and Smith [17].

4.2 Subjective quantitative evaluations

To gauge the quality of the grayscale images produced, we first undertake a subjective evaluation. The primary criteria are the maintenance of the original image's overall characteristics and the high contrast between differing colors. We conduct this comparative study across natural landscape images, the results of which are presented in Figure 11.



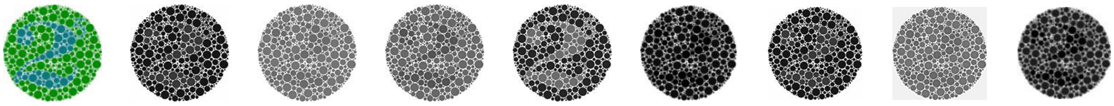
a b c d e f g h i

Figure 11: Natural landscape images processed by our methods and other methods (a) Original color image, (b) Bala [12], (c) CIE Y channel [9], (d) Color2Gray [10], (e) Grundland [15], (f) Kim [16] (g) Rasche [13], (h) Smith [17], (i) Ours.

In Fig. 12, our algorithm demonstrates its strength in generating grayscale images that excel in both luminance and chromaticity differences, resulting in high contrast that closely matches human perception. Although it does not always outperform the other algorithms in all images (especially the image shown in Fig. 13), its performance is commendable.

a b c d e f g h i

Figure 12: Color images with a wide range of colors processed by our algorithm and other methods (a) Original color image, (b) Bala [12], (c) CIE Y channel [9], (d) Color2Gray [10], (e) Grundland [15], (f) Kim [16] (g) Rasche [13], (h) Smith [17], (i) Ours.



a b c d e f g h i

Figure 13: Color image processed by our algorithm and other methods (a) Original color image, (b) Bala [12], (c) CIE Y channel [9], (d) Color2Gray [10], (e) Grundland [15], (f) Kim [16] (g) Rasche [13], (h) Smith [17], (i) Ours.

Moreover, Figures 11-13 reveal an interesting trend. Our algorithm performs modestly on images featuring natural landscapes, which often exhibit significant variations in brightness. Nevertheless, the algorithm excels in images with uniform brightness regions but differing chromaticity.

In summary, while our proposed decolorization algorithm is not universally superior across all types of images, it exhibits marked improvements in grayscale images with balanced brightness regions and chromatic variance, confirming its efficacy under specific conditions.

4.3 Objective quantitative assessment

In this section, I compare the de-coloring method proposed in this paper with the following seven methods by means of two existing types of objective metrics: the : Bala [12], CIE Y [9], Color2Gray [10], Grundland [15] , Kim [16], Rasche [13], and Smith [17].

4.3.1 Decolorized image metrics: CCPR, CCFPR and E-score

CCPR, CCFR and E-score are metrics defined for contrast-preserving color removal algorithms [11].

Color Contrast Preservation Ratio (CCPR) aims to quantitatively evaluate the contrast preserved by the decolorization algorithm: if the color difference δ is smaller than the threshold τ , it is almost indistinguishable in human vision. Therefore, preserving the contrast of color removal means preserving the color change that can be perceived by humans. The specific definition of CCPR is as follows:

$$\mathrm{CCPR} = \frac{\#\left\{(\mathrm{x},\mathrm{y}) \middle| (\mathrm{x},\mathrm{y}) \in \Omega, \left|\mathrm{G_x} - \mathrm{G_y}\right| \geq \tau\right\}}{\parallel \Omega \parallel}$$

where all pixel pairs (x, y) in Ω satisfy the color contrast $\delta_{\mathrm{x,y}} \geq \tau$ and Ω denotes the number of pixel pairs contained in the set Ω. $\#\left\{(\mathrm{x},\mathrm{y}) \middle| (\mathrm{x},\mathrm{y}) \in \Omega, \left|\mathrm{G_x} - \mathrm{G_y}\right| \geq \tau\right\}$ represents the number of pixel pairs in the set Ω for which the grey scale contrast is still not less than the threshold τ after decolorization.

Color Content Fidelity Ratio (CCFR) measures the content similarity and identifies the credibility of a grayscale map in terms of its content structure. CCFR is defined as:

$$\mathrm{CCFR} = 1 - \frac{\#\left\{(\mathrm{x},\mathrm{y}) \middle| (\mathrm{x},\mathrm{y}) \in \Theta, \delta_{\mathrm{x,y}} \leq \tau\right\}}{\parallel \Theta \parallel}$$

where Θ is the set of pixel pairs consisting of pixel pairs whose gray scale contrast $\left|\mathrm{G_x} - \mathrm{G_y}\right|$ is greater than a threshold τ. CCFR can be used to test the percentage of undesired artifacts produced by the decolorization algorithm.

E-score is the reconciled mean of CCPR and CCFR, and its highest E-score value means that the color contrast of the input color image is retained in the output grayscale image without forming any new edges [11]. The specific definitions are as follows:

$$\mathrm{E-score} = \frac{2 \cdot \mathrm{CCPR} \cdot \mathrm{CCFR}}{\mathrm{CCPR} + \mathrm{CCFR}}$$

Figure 14 illustrates the CCPR, CCFR, and E-score results for the second column of images in Figure 12. the E-score is essentially in line with human perception. Ideally, when all color contrast is preserved in the grayscale results without creating new edges, the values of CCPR and CCFR are both close to 1 when the E-score value is the highest.

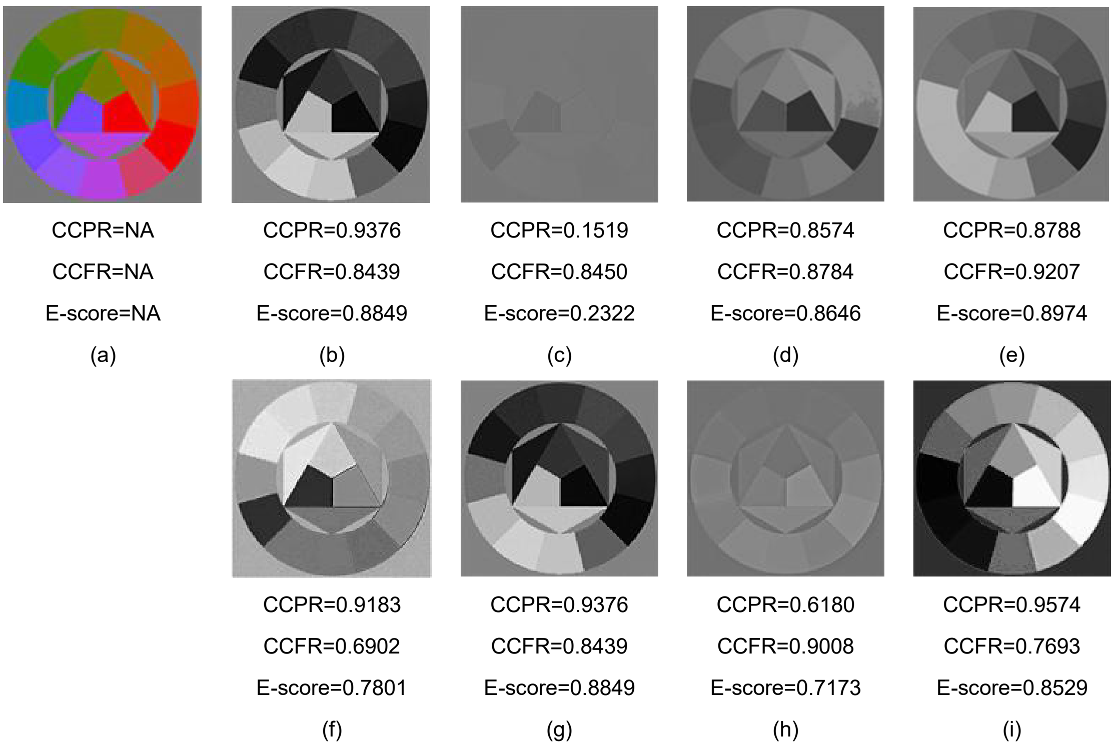


Figure 14: CCPR, CCFR, and E-Score for different grayscale results at τ = 5. a Input images. b-h Results for CIE Y [9], Bala [12], Color2Gray [10], Grundland [15], Kim [16], Rasche [13], and Smith [17]. (i) Our results.

4.3.2 Structural Similarity Index Measure SSIM

The Structural Similarity Index SSIM is used to measure the similarity between two color images and is closely related to the human visual system [30]. The human visual perception system is able to extract structural information from a scene to recognize the information difference between the reference scene and the sample scene, so SSIM evaluates the similarity by three aspects: brightness, contrast and structure.

Let x and y be the image blocks at the same position of the images to be compared, and the luminance comparison term $l(x, y)$, the contrast comparison term $d(x, y)$ and the structure comparison term $s(x, y)$ at x and y are defined as.

$$l(x, y) = \frac{2\mu_x\mu_y + C_1}{\mu_x + \mu_y + C_1}$$

$$d(x, y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x + \sigma_y + C_2}$$

$$s(x, y) = \frac{\sigma_{xy} + C_3}{2\sigma_x\sigma_y + C_{31}}$$

where μ, σ and σ_{xy} represent the mean, standard deviation and covariance, respectively. C_1, C_2 and C_3 are very small normal values, avoiding the case where the denominator is 0. The SSIM index is obtained by multiplying the three components:

$$SSIM(x, y) = l(x, y)^{\alpha} \cdot d(x, y)^{\beta} \cdot s(x, y)^{\gamma}$$

where α, β and γ are positive parameters. The SSIM metrics for the whole image are obtained by traversing the image blocks of the whole image and taking the mean values.

4.3.3 Evaluation based on CCPR, CCFR, and E-score Metrics

To ensure a comprehensive assessment of image decolorization techniques, we employed a curated dataset that combines 14 images from the renowned Ĉadík [29] dataset along with other greyscale images generated using the decolorization method of Kim, et al. [16]. This augmentation was undertaken to bring a broader context to the evaluation. We employed three pivotal metrics—Color Contrast Preservation Ratio (CCPR), Color Constancy Fidelity Ratio (CCFR), and E-score—for the quantitative evaluation. Each of these metrics brings unique perspectives into the decolorization quality and artifact formation.

In our analytical framework, we selected two distinct threshold values, $\tau = 15$ and $\tau = 40$, to scrutinize the impact of different intensity levels on the aforementioned metrics. The outcomes are systematically documented in Tables 2 and 3, and illustrated in Figures 15 and 16.

Table 2: CCPR, CCFR, and E-score at $\tau = 15$ for different grayscale results.

	Bala	CIE Y	Color2Gray	Grundland	Kim	Rasche	Smith	Ours
CCPR	0.7545	0.6733	0.7914	0.8201	0.8235	0.8524	0.6889	0.847
CCFR	0.8414	0.9039	0.8795	0.8807	0.8853	0.8413	0.8729	0.9368
E-score	0.7672	0.7092	0.8237	0.8369	0.834	0.8345	0.7161	0.8857

Table 3: CCPR, CCFR, and E-score at $\tau = 40$ for different grayscale results.

	Bala	CIE Y	Color2Gray	Grundland	Kim	Rasche	Smith	Ours
CCPR	0.5965	0.5184	0.5886	0.6512	0.6784	0.7153	0.5283	0.7161
CCFR	0.8363	0.8802	0.8570	0.8505	0.8754	0.7750	0.8376	0.8723
E-score	0.6244	0.5684	0.6386	0.6879	0.7323	0.7170	0.5651	0.7679

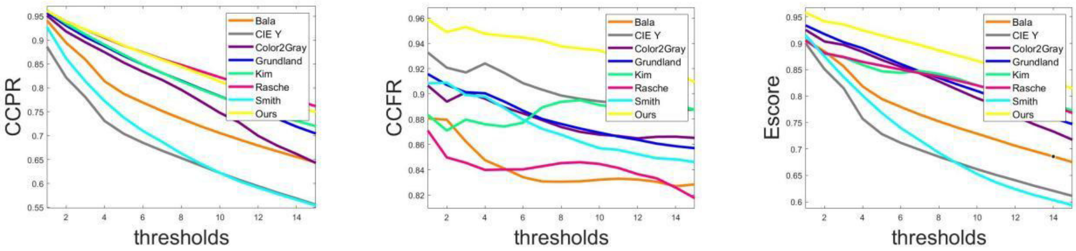


Figure 15: Comparison of Bala [12], CIE Y [9], Color2Gray [10], Grundland [15] , Kim [16], Rasche [13], and Smith [17], and our method in terms of CCPR, CCFR, and E-score on 14 images at τ = 15. The x-axis represents the different thresholds τ, and the y-axis represents the corresponding CCPR, CCFR, and E-score values. score values.

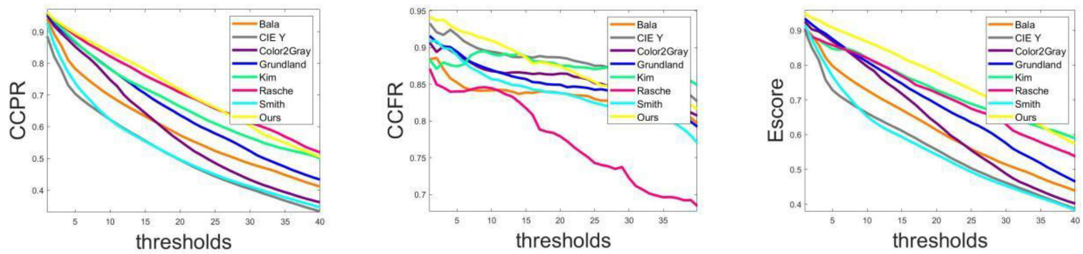


Figure 16: Comparison of Bala [12], CIE Y [9], Color2Gray [10], Grundland [15] , Kim [16], Rasche [13], and Smith [17], and our method in terms of CCPR, CCFR, and E-score on 14 images at τ = 40. The x-axis represents the different thresholds τ, and the y-axis represents the corresponding CCPR, CCFR, and E-score values. score values.

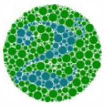
Upon applying our analytical lens to the image decolorization methodologies sourced from the Ĉadík [29] dataset and Kim's [16] approach, several salient trends emerged across the metrics of CCPR, CCFR, and E-score. Techniques devoid of content-specific considerations, typified by the CIE Y-channel [9], routinely produced subpar CCPR outcomes, thus highlighting their inefficacy in preserving color contrast. Contrarily, methods such as Bala manifested lower CCFR scores, signaling the likelihood of artifact introduction. Notably, E-score revealed itself as a nuanced evaluation metric that gains interpretive depth when juxtaposed with CCPR and CCFR metrics. Our approach manifested consistently robust E-score values, thereby corroborating its balanced decolorization efficacy.

Our algorithm demonstrates robust performance, particularly excelling in CCFR with a score of 0.9368 at $\tau = 15$. It also achieves competitive results in CCPR and E-score, suggesting a balanced decolorization performance. While methods such as Rasche [13] exhibit higher CCPR scores in some instances, their performance lacks consistency across all metrics. In terms of E-score, a crucial aggregate metric, our method consistently outperforms competitors at both intensity levels. Specifically, at $\tau = 40$, our algorithm reaches a maximum E-score of 0.7679, illustrating its aptitude for generating high-quality grayscale images even under varying conditions.

For a broader understanding, Table 4 evaluates our algorithm at $\tau = 5$ across six representative images. Here, we observed high E-score values, peaking at 0.9644, affirming our algorithm's efficacy.

While our CCPR metrics were robust, they were occasionally outpaced by other techniques like Rasche [13]. Despite this, our CCFR scores were exceptionally high, cresting at 0.9897 in certain scenarios. Collectively, these findings lend credence to our algorithm's balanced and efficacious performance, especially when evaluated across the multifaceted metrics of CCPR, CCFR, and E-score.

Table 4: Comparison of CCPR, CCFR and E-score of Image1-6 at $\tau = 5$

Test image	Metric	Bala	CIE Y	Color2Gray	Grundland	Kim	Rasche	Smith	Ours
	CCPR	0.7969	0.8020	0.8389	0.8937	0.9132	0.9008	0.8105	0.8692
	CCFR	0.7419	0.8019	0.8294	0.7542	0.8100	0.6817	0.7575	0.9073
	E-score	0.7680	0.8020	0.8341	0.8158	0.8570	0.7702	0.7827	**0.8878**
	CCPR	0.9299	0.9095	0.9217	0.9371	0.9653	0.9299	0.9280	0.9805
	CCFR	0.8147	0.8265	0.8184	0.8121	0.9644	0.8147	0.8046	0.9449
	E-score	0.8682	0.8658	0.8667	0.8696	**0.9648**	0.8682	0.8618	0.9623
	CCPR	0.9327	0.9285	0.9212	0.9404	0.9339	0.9327	0.9275	0.9372
	CCFR	0.8677	0.8621	0.7812	0.8347	0.9598	0.8677	0.8151	0.9897
	E-score	0.8984	0.8925	0.8428	0.8828	0.9462	0.8984	0.8660	**0.9626**
	CCPR	0.9092	0.9122	0.9194	0.9145	0.8630	0.9225	0.9162	0.9225
	CCFR	0.9811	0.9906	0.9774	0.9845	0.9862	0.9844	0.9681	0.9879
	E-score	0.9432	0.9493	0.9472	0.9478	0.9197	0.9522	0.9412	**0.9539**
	CCPR	0.9342	0.9327	0.9326	0.9249	0.9334	0.9436	0.9358	0.9346
	CCFR	0.9664	0.9964	0.9945	0.9937	0.9928	0.9928	0.9898	0.9965
	E-score	0.9500	0.9633	0.9624	0.9578	0.9620	**0.9675**	0.9619	0.9644
	CCPR	0.4441	0.3280	0.9156	0.8825	0.9639	0.9100	0.3789	0.9577
	CCFR	0.5788	0.9910	0.8116	0.8848	0.2031	0.4370	0.9530	0.8516
	E-score	0.4599	0.4233	0.8579	0.8785	0.3295	0.5816	0.4889	**0.9003**

Bold values indicate the highest E-score value for each image.

4.3.4 Structural Similarity Index Measure (SSIM) Evaluation

Ĉadík [29] conducted a comprehensive evaluation of seven major color-to-grayscale algorithms using a dataset of 168 images. Preference scores were aggregated from 20,328 user responses involving 119 participants. The algorithm with the highest preference score per image was considered the best-performing one. Their results show that the algorithms of Smith, et al. [17] and Grundland et al.[15] obtained high scores in the user survey. To extend upon Ĉadík's work, we selected 14 representative images from the original dataset to assess various algorithms, including our own, through Structural Similarity Index Measure (SSIM).

SSIM assesses image similarity in terms of three key components: luminance (brightness), contrast, and structure [30]. Each of these dimensions provides insights into how closely an algorithm approximates the 'best-performing' method as per Ĉadík's preference scores. As shown in table 5.

Observations:

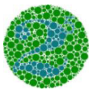
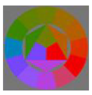
- **High Similarity:** Generally, our algorithm shows high similarity to the 'best-performing' methods across most of the selected images, especially in terms of luminance and structure.

- **Exceptions:** However, the performance dips for specific color images, which invites further investigation into potential causes.

- **Competitive Performance:** As evidenced by Table 5, our method performs competitively, holding middle-ground when ranked based on average SSIM values.

Table 5: The table compares the SSIM values of various algorithms against

each image. A star (★) indicates the most favored algorithm according to Ĉadík's study. Bolded values denote the closest SSIM to the best-performing algorithm, while values in yellow font signify the lowest SSIM for each image.

Id	color image	Bala	CIE Y	Color2Gray	Grundland	Kim	Rasche	Smith	Ours
image1		0.7628	0.8153	**0.8899**	★	0.7917	0.8718	0.8457	0.8296
image2		**0.9503**	0.8360	0.8595	★	0.6881	**0.9503**	0.8469	0.5990
image3		0.8094	0.9800	**0.9442**	0.8948	0.8030	0.8094	★	0.8367
image4		0.9178	0.9555	0.9508	★	0.9062	0.8074	**0.9678**	0.9215
image5		0.9017	**0.9546**	0.9509	0.7993	0.8692	0.9034	★	0.9159
image6		0.8480	**0.9000**	0.7544	★	0.6322	0.5638	0.8859	0.5497
image7		0.8864	0.8647	★	0.8334	0.8864	0.7652	0.8582	**0.9054**
image8		0.7037	0.7690	0.1722	★	0.6776	0.5602	**0.7841**	0.4294
image9		0.9650	**0.9788**	0.9607	0.9398	0.9168	0.9650	★	0.9405
image10		0.6829	**0.8770**	0.7880	★	0.7288	0.6864	0.8388	0.7965
image11		**0.7438**	0.7383	0.6376	★	0.5002	**0.7438**	0.7048	**0.3430**
image12		0.8142	0.8414	0.6281	★	0.8142	**0.8779**	0.8209	0.7791
image13		0.8646	**0.8673**	0.7936	★	0.7738	0.8646	0.8446	0.8025
image14		**0.9621**	0.9475	0.9519	0.9090	0.8355	**0.9621**	★	0.8543

4.3.5 Comprehensive evaluation of image decolorization

A comprehensive evaluation framework is essential for understanding the multifaceted performance of color-to-grayscale conversion algorithms. In this study, we employ four key metrics: Structural Similarity Index Measure (SSIM), Color Contrast Preserving Ratio (CCPR), Color Content Fidelity Ratio (CCFR), and E-score. These metrics capture different dimensions of image quality and conversion fidelity, thereby providing a nuanced and comprehensive assessment. As shown in Table 6.

In a comprehensive evaluation of image processing algorithms utilizing a multifaceted framework, which includes the Structural Similarity Index Measure (SSIM), Color Contrast Preservation Ratio (CCPR), Color Contrast Fidelity Ratio (CCFR), and E-score metrics, two algorithms—Grundland [15] and our proposed method—emerged as leading performers. Grundland [15] demonstrated superior performance in SSIM, alongside commendable results in CCPR, CCFR, and E-score. In contrast, our algorithm excelled in CCPR, CCFR, and E-score while delivering moderate performance in SSIM. Each algorithm manifested distinct strengths: Grundland [15] was notably effective in preserving structural information at the expense of some color contrast and fidelity, whereas our method offered a balanced performance across metrics, even achieving above-average results in SSIM.

Further underscoring the robustness of our proposed algorithm, it consistently yielded competitive performance across the board. Particularly, it outperformed most existing algorithms in CCFR and E-score, signifying a high degree of fidelity and overall image quality. However, our method did exhibit areas for improvement, specifically in the SSIM and CCPR metrics, for particular types of color images. Thus, while our algorithm demonstrates versatility and robustness, these results also highlight avenues for future optimization.

Table 6 : SSIM, CCPR, CCPF, and E-score were scored at τ = 15 on 14 representative images from the original Ĉadík [29] dataset.

Id	Metric	Bala	CIE Y	Color2Gray	Grundland	Kim	Rasche	Smith	Ours
Image1	SSIM	0.7628	0.8153	0.8899	1	0.7917	0.8718	0.8457	0.8296
	CCPR	0.7969	0.8020	0.8389	0.8937	0.9132	0.9008	0.8105	0.8692
	CCFR	0.7419	0.8019	0.8294	0.7542	0.8100	0.6817	0.7575	0.9073
	E-score	0.7680	0.8020	0.8341	0.8158	0.8570	0.7702	0.7827	0.8878
Image2	SSIM	0.9503	0.8360	0.8595	1	0.6881	0.9503	0.8469	0.5990
	CCPR	0.9299	0.9095	0.9217	0.9371	0.9653	0.9299	0.9280	0.9805
	CCFR	0.8147	0.8265	0.8184	0.8121	0.9644	0.8147	0.8046	0.9449
	E-score	0.8682	0.8658	0.8667	0.8696	0.9648	0.8682	0.8618	0.9623
Image3	SSIM	0.8094	0.9800	0.9442	0.8948	0.8030	0.8094	1	0.8367
	CCPR	0.9327	0.9285	0.9212	0.9404	0.9339	0.9327	0.9275	0.9372
	CCFR	0.8677	0.8621	0.7812	0.8347	0.9598	0.8677	0.8151	0.9897
	E-score	0.8682	0.8658	0.8667	0.8696	0.9648	0.8682	0.8618	0.9623
Image4	SSIM	0.9178	0.9555	0.9508	1	0.9062	0.8074	0.9678	0.9215
	CCPR	0.9092	0.9122	0.9194	0.9145	0.8630	0.9225	0.9162	0.9225
	CCFR	0.9811	0.9906	0.9774	0.9845	0.9862	0.9844	0.9681	0.9879
	E-score	0.9432	0.9493	0.9472	0.9478	0.9197	0.9522	0.9412	0.9539
Image5	SSIM	0.9017	0.9546	0.9509	0.7993	0.8692	0.9034	1	0.9159
	CCPR	0.9342	0.9327	0.9326	0.9249	0.9334	0.9436	0.9358	0.9346
	CCFR	0.9664	0.9964	0.9945	0.9937	0.9928	0.9928	0.9898	0.9965
	E-score	0.9500	0.9633	0.9624	0.9578	0.9620	0.9675	0.9619	0.9644
Image6	SSIM	0.8480	0.9000	0.7544	1	0.6322	0.5638	0.8859	0.5497
	CCPR	0.4441	0.3280	0.9156	0.8825	0.9639	0.9100	0.3789	0.9577
	CCFR	0.5788	0.9910	0.8116	0.8848	0.2031	0.4370	0.9530	0.8516
	E-score	0.4599	0.4233	0.8579	0.8785	0.3295	0.5816	0.4889	0.9003

Image7	SSIM	0.8864	0.8647	1	0.8334	0.8864	0.7652	0.8582	0.9054
	CCPR	0.9181	0.9209	0.9229	0.9325	0.9334	0.9340	0.9178	0.9269
	CCFR	0.8755	0.9381	0.9011	0.9066	0.9626	0.8859	0.9168	0.9461
	E-score	0.8961	0.9291	0.9118	0.9192	0.9474	0.9092	0.9169	0.9360
Image8	SSIM	0.7037	0.7690	0.1722	1	0.6776	0.5602	0.7841	0.4294
	CCPR	0.5831	0.5141	0.8594	0.8351	0.7828	0.8660	0.5480	0.9132
	CCFR	0.9129	0.9966	0.9924	0.9940	0.9184	0.8870	0.9916	0.9210
	E-score	0.6987	0.6655	0.9201	0.9059	0.8421	0.8753	0.6915	0.9171
Image9	SSIM	0.9650	0.9788	0.9607	0.9398	0.9168	0.9650	1	0.9405
	CCPR	0.9142	0.8952	0.8597	0.8721	0.8946	0.9142	0.8591	0.9064
	CCFR	0.8547	0.9055	0.8847	0.8731	0.9306	0.8547	0.8645	0.9221
	E-score	0.8833	0.9003	0.8720	0.8726	0.9121	0.8833	0.8617	0.9141
Image10	SSIM	0.6829	0.8770	0.7880	1	0.7288	0.6864	0.8388	0.7965
	CCPR	0.9194	0.9387	0.9017	0.9325	0.9399	0.9192	0.9234	0.9191
	CCFR	0.9275	0.9661	0.9637	0.9655	0.9740	0.9337	0.9312	0.9914
	E-score	0.9232	0.9522	0.9316	0.9487	0.9564	0.9262	0.9271	0.9538
Image11	SSIM	0.7438	0.7383	0.6376	1	0.5002	0.7438	0.7048	0.3430
	CCPR	0.9376	0.1519	0.8574	0.8788	0.9183	0.9376	0.6180	0.9574
	CCFR	0.8439	0.8450	0.8784	0.9207	0.6902	0.8439	0.9008	0.7693
	E-score	0.8849	0.2322	0.8646	0.8974	0.7801	0.8849	0.7173	0.8529
Image12	SSIM	0.8142	0.8414	0.6281	1	0.8142	0.8779	0.8209	0.7791
	CCPR	0.8900	0.9017	0.8857	0.9080	0.8900	0.9275	0.8977	0.9150
	CCFR	0.9937	0.9118	0.8661	0.8992	0.9937	0.8827	0.8863	0.9574
	E-score	0.9383	0.9066	0.8758	0.9036	0.9383	0.9045	0.8920	0.9356
Image13	SSIM	0.8646	0.8673	0.7936	1	0.7738	0.8646	0.8446	0.8025
	CCPR	0.9419	0.9205	0.9136	0.9300	0.9258	0.9419	0.9192	0.9323
	CCFR	0.9394	0.9667	0.9519	0.9432	0.9762	0.9394	0.9495	0.9741
	E-score	0.9407	0.9429	0.9323	0.9366	0.9502	0.9407	0.9340	0.9527

Image14	SSIM	0.9621	0.9475	0.9519	0.9090	0.8355	0.9621	1	0.8543
	CCPR	0.9414	0.9296	0.9275	0.9526	0.9302	0.9414	0.9373	0.9159
	CCFR	0.8843	0.9150	0.9154	0.8874	0.9150	0.8843	0.8808	0.9876
	E-score	0.9119	0.9222	0.9214	0.9188	0.9222	0.9119	0.9082	0.9501

5. Conclusion

The process of image decolorization is a process of data dimensionality reduction, which can easily lead to the loss of structure and details of the original image. This project provides an optimization algorithm for converting color images into greyscale images. Our approach utilizes the CIE L * a * b * color space and employs a local mapping technique, as opposed to the global mapping approach commonly found in the literature, such as that employed by Gooch et al. This efficacy is further enhanced by the inclusion of the "extended neighborhood" concept, which improves detail capture and contrast improvement.

A comprehensive evaluation was carried out using a range of established metrics including Structural Similarity Index (SSIM), Colour Contrast Retention (CCPR), Colour Contrast Fidelity (CCFR) and E-score. Comparative analyses with existing methods show a substantial improvement in the performance of the algorithm, especially in terms of CCFR and E-score. This demonstrates the capability of our algorithm in preserving color features that are critical for human perception and various professional applications. Nonetheless, the algorithm shows moderate degradation in SSIM and CCPR metrics, which points to potential directions for future research.

Despite its strengths, our algorithm is not without limitations. It performs poorly in regions with subtle color transitions. In addition, the algorithm shows limited results in high dynamic range scenes with large brightness fluctuations. Due to the extended neighborhood analysis, our method also requires a higher computational load, which can be a barrier for real-time applications.

6. Feature work

Adaptive Neighbourhood Resizing

One of the most pressing issues identified in our current algorithm is its inability to perform optimally in high dynamic range environments. To address this, one promising avenue for future work is the incorporation of adaptive strategies for dynamically resizing the "extended neighborhood." In such adaptive mechanisms, the neighborhood size could automatically adjust according to the local image statistics, such as luminance or texture complexity. This would enable the algorithm to more effectively handle a wide range of scenarios, from low-contrast scenes to high dynamic range environments.

Computational Efficiency

The computational overhead of our current approach is another crucial concern. Although our algorithm has demonstrated strong efficacy in preserving chromatic aberration and other perceptual features, the computational cost, particularly associated with the "extended neighborhood" analysis, needs to be optimized for real-time applications. Future work could explore the use of more efficient data structures, parallel processing, or even hardware acceleration to mitigate this issue.

Machine Learning and Deep Learning Approaches

Another intriguing direction for future research is the use of machine learning algorithms to predict optimal parameters for our grayscale conversion process. Given a large enough dataset of color and corresponding high-quality grayscale images, a machine learning model could potentially learn to predict the ideal settings for converting new, unseen images. This could include not just basic parameters like neighborhood size but also more complex features like weighting functions for different color channels. Another potential approach is to use a deep learning algorithm, such as a convolutional neural network (CNN) [31], to learn the best grey scale conversion method for a given image or set of images.

Human Perception Models

Our work has primarily focused on quantitative metrics like CCFR and E-score to evaluate image quality. However, human perception of image quality can be influenced by a multitude of factors that may not be entirely captured by these metrics. Therefore, future research could focus on integrating models of human visual perception into the algorithm. Such models could be based on psychophysical experiments that quantify how humans perceive color and grayscale images under various conditions.

Extension to Other Media Types

The current work focuses exclusively on the conversion of static color images to grayscale. An intriguing expansion of this work would be to apply the algorithm to different types of media, such as video or 3D models [32]. The unique challenges presented by these media types, such as temporal color changes in video or the additional spatial dimension in 3D models, would require algorithmic adjustments and could provide interesting avenues for future work.

Large-Scale Validation and Robustness

Our preliminary validation has been constrained by the scope of the current study. Future research should consider large-scale validation efforts that incorporate a diverse array of image types, subject matters, and conditions. This would not only validate the general applicability of the algorithm but also contribute to its robustness by revealing any edge cases or anomalies that have not been previously considered.

In conclusion, although our algorithm has made some progress in the field of greyscale image conversion, there is still much work to be done to further improve its capabilities. These future research directions are crucial for the algorithms to adapt to different application scenarios.

Reference

[1] R. E Woods and R. C Gonzalez, "Digital image processing," ed: Pearson Education Ltd., 2008.

[2] T. Lillesand, R. W. Kiefer, and J. Chipman, *Remote sensing and image interpretation*. John Wiley & Sons, 2015.

[3] C. Liu, J. Yuen, and A. Torralba, "Sift flow: Dense correspondence across scenes and its applications," *IEEE transactions on pattern analysis and machine intelligence,* vol. 33, no. 5, pp. 978-994, 2010.

[4] C. Kanan and G. W. Cottrell, "Color-to-grayscale: does the method matter in image recognition?," *PloS one,* vol. 7, no. 1, p. e29740, 2012.

[5] F. Tong, "Foundations of vision," *Stevens' handbook of experimental psychology and cognitive neuroscience,* vol. 2, pp. 1-61, 2018.

[6] W. Wang, Z. Li, and S. Wu, "Color contrast-preserving decolorization," *IEEE Transactions on Image Processing,* vol. 27, no. 11, pp. 5464-5474, 2018.

[7] S. Liu and X. Zhang, "Image decolorization combining local features and exposure features," *IEEE Transactions on Multimedia,* vol. 21, no. 10, pp. 2461-2472, 2019.

[8] G. Sharma and R. Bala, *Digital color imaging handbook*. CRC press, 2017.

[9] M. D. Fairchild, *Color appearance models*. John Wiley & Sons, 2013.

[10] A. A. Gooch, S. C. Olsen, J. Tumblin, and B. Gooch, "Color2gray: salience-preserving color removal," *ACM Transactions on Graphics (TOG),* vol. 24, no. 3, pp. 634-639, 2005.

[11] C. Lu, L. Xu, and J. Jia, "Contrast preserving decolorization with perception-based quality metrics," *International journal of computer vision,* vol. 110, pp. 222-239, 2014.

[12] R. Bala and R. Eschbach, "Spatial color-to-grayscale transform preserving chrominance edge information," in *Color and imaging conference*, 2004, vol. 12: Society of Imaging Science and Technology, pp. 82-86.

[13] K. Rasche, *Re-coloring images for gamuts of lower dimension*. Clemson University, 2005.

[14] K. Rasche, R. Geist, and J. Westall, "Detail preserving reproduction of color images for monochromats and dichromats," *IEEE Computer Graphics and Applications,* vol.

25, no. 3, pp. 22-30, 2005.

[15] M. Grundland and N. A. Dodgson, "The decolorize algorithm for contrast enhancing, color to grayscale conversion," University of Cambridge, Computer Laboratory, 2005.

[16] Y. Kim, C. Jang, J. Demouth, and S. Lee, "Robust color-to-gray via nonlinear global mapping," in *ACM SIGGRAPH Asia 2009 papers*, 2009, pp. 1-4.

[17] K. Smith, P. E. Landes, J. Thollot, and K. Myszkowski, "Apparent greyscale: A simple and fast conversion to perceptually accurate images and video," in *Computer graphics forum*, 2008, vol. 27, no. 2: Wiley Online Library, pp. 193-200.

[18] C. Lu, L. Xu, and J. Jia, "Real-time contrast preserving decolorization," in *SIGGRAPH Asia 2012 Technical Briefs*, 2012, pp. 1-4.

[19] S. Lee, Y. Kwak, Y. J. Kim, S. Park, and J. Kim, "Contrast-preserved chroma enhancement technique using YCbCr color space," *IEEE Transactions on Consumer Electronics,* vol. 58, 2012.

[20] R. S. Hunter, "Photoelectric Color Difference Meter*," *J. Opt. Soc. Am.,* vol. 48, no. 12, pp. 985-995, 1958/12/01 1958, doi: 10.1364/JOSA.48.000985.

[21] N. A. Ibraheem, M. M. Hasan, R. Z. Khan, and P. K. Mishra, "Understanding color models: a review," *ARPN Journal of science and technology,* vol. 2, no. 3, pp. 265-275, 2012.

[22] K. McLaren, "XIII—The development of the CIE 1976 (L* a* b*) uniform colour space and colour‐difference formula," *Journal of the Society of Dyers and Colourists,* vol. 92, no. 9, pp. 338-341, 1976.

[23] R. M. Evans, *The perception of color*. Wiley-Interscience, 1974.

[24] Y. Ou, P. Ambalathankandy, M. Ikebe, S. Takamaeda, M. Motomura, and T. Asai, "Real-time tone mapping: A state of the art report," *arXiv preprint arXiv:2003.03074,* 2020.

[25] J. Yang, A. Hore, and O. Yadid‐Pecht, "Local tone mapping algorithm and hardware implementation," *Electronics Letters,* vol. 54, no. 9, pp. 560-562, 2018.

[26] C. A. Parraga and X. Otazu, "Which tone-mapping operator is the best? A comparative study of perceptual quality," *JOSA A,* vol. 35, no. 4, pp. 626-638, 2018.

[27] J. Wang, R. K. Wong, and T. C. Lee, "Locally linear embedding with additive noise," *Pattern recognition letters,* vol. 123, pp. 47-52, 2019.

[28] C. Lu, L. Xu, and J. Jia, "Contrast preserving decolorization," presented at the 2012 IEEE International Conference on Computational Photography (ICCP), 2012. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/ICCPhot.2012.6215215.

[29] M. Ĉadík, "Perceptual evaluation of color‐to‐grayscale image conversions," in *Computer graphics forum*, 2008, vol. 27, no. 7: Wiley Online Library, pp. 1745-1754.

[30] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing,* vol. 13, no. 4, pp. 600-612, 2004.

[31] R. Chauhan, K. K. Ghanshala, and R. Joshi, "Convolutional neural network (CNN) for image detection and recognition," in *2018 first international conference on secure cyber computing and communication (ICSCCC)*, 2018: IEEE, pp. 278-282.

[32] H. Gong, G. D. Finlayson, R. B. Fisher, and F. Fang, "3D color homography model for photo-realistic color transfer re-coding," *The Visual Computer,* vol. 35, pp. 323-333, 2019.