2018년 8월
석사학위논문
Study on edge detection with color-to-grayscale algorithms
IJAZ AHMAD

2018년 8월

석사 학위 논문

Study on edge detection with color-to-grayscale algorithms

조선대학교 대학원

컴퓨터공학과

IJAZ AHMAD

Study on edge detection with color-to-grayscale algorithms

컬러-회색 스케일 알고리즘의 에지 검출에 대한 연구

2018년 8월 24일

Graduate School of Chosun University

Department of Computer Engineering

AHMAD IJAZ

Study on edge detection with color-to-grayscale algorithms

지도교수 신 석 주

이 논문을 공학석사학위 신청 논문으로 제출함

2018년 4월

조선대학교 대학원

컴퓨터공학과

AHMAD IJAZ

IJAZ AHMAD 의 석사학위논문을 인준함

위원장 조선대학교 교 수 강 문 수 (인)

위 원 조선대학교 교 수 권 구 락 (인)

위 원 조선대학교 교 수 신 석 주 (인)

2018년 5월

조선대학교 대학원

TABLE OF CONTENTS

TABLE OF CONTENTS.....i
LIST OF FIGURES.....iii
LIST OF TABLES.....iv
ABBREVIATIONS.....vi
ABSTRACT.....vii
요약.....viii
I. INTRODUCTION.....1
1.1. Motivation.....
1.2. Thesis Contribution.....
II. BACKGROUND.....6
2.1. Edge.....
2.1.1. Gray Edge.....
2.1.2. Color Edge.....
2.1.3. Edge Detection.....
2.2. Color-to-Grayscale Conversion.....
2.2.1 Color-to-Grayscale Conversion Algorithms.....
III. EDGE DETECTORS EVALUATION.....14
3.1. Non-Reference Objective Measures.....
3.2. Reference-Based Objective Measures.....
3.2.1 Assessment based-on Confusion Matrix.....
3.2.2 Assessment involving Distance of Misplaced Pixels

IV. EXPERIMENTAL RESULTS AND ANALYSIS.....22
4.1. Discussion.....
V. CONCLUSION.....63
VI. REFERENCES.....64
VII. ACKNOWLEDGMENTS.....69

LIST OF FIGURES

Figure 1. Illustration of TPs, FNs and FPs 18
Figure 2. Performance measure of color-to-gryascale algorithms on edge detectors for Berkely natural images dataset.....30
Figure 3. Grayscale representations of synthetic image.....31
Figure 4. Resultant Sobel edge maps.....32
Figure 5. Resultant Roberts edge maps.....35
Figure 6. Resultant Prewitt edge maps.....38
Figure 7. Resultant LoG edge maps.....41
Figure 8. Resultant Canny edge maps.....44
Figure 9. Performance measure of color-to-grayscale algorithms for synthetic image.....47
Figure 10. Grayscale representations of natural image.....48
Figure 11. Resultant Sobel edge maps.....49
Figure 12. Resultant Roberts edge maps.....52
Figure 13. Resultant Prewitt edge maps.....55
Figure 14. Resultant LoG edge maps.....58
Figure 15. Resultant Canny edge maps.....62

LIST OF TABLES

Table 1. Color-to-grayscale conversion algorithms.....23
Table 2. Parameters used in the edge detection of synthetic images..... 24
Table 3. Parameters used in the edge detection of natural images..... 24
Table 4. Performance measurement metrics (A).....25
Table 5. Performance measurement metrics (B).....25
Table 6. Performance measurement metrics (C).....26
Table 7. Evaluation of Sobel edge maps using Table 5 measures.....33
Table 8. Evaluation of Sobel edge maps using Table 4 measures.....33
Table 9. Evaluation of Sobel edge maps using FoMs given in Table 6.....34
Table 10. Evaluation of Sobel edge maps using Table 6 measures.....34
Table 11. Evaluation of Roberts edge maps using Table 5 measures.....36
Table 12. Evaluation of Roberts edge maps using Table 4 measures.....36
Table 13. Evaluation of Roberts edge maps using FoMs given in Table 6.....37
Table 14. Evaluation of Roberts edge maps using Table 6 measures.....37
Table 15. Evaluation of Prewitt edge maps using Table 5 measures.....39
Table 16. Evaluation of Prewitt edge maps using Table 4 measures.....39
Table 17. Evaluation of Prewitt edge maps using FoMs given in Table 6.....40
Table 18. Evaluation of Prewitt edge maps using Table 6 measures.....40
Table 19. Evaluation of LoG edge maps using Table 5 measures.....42
Table 20. Evaluation of LoG edge maps using Table 4 measures.....42
Table 21. Evaluation of LoG edge maps using FoMs given in Table 6.....43
Table 22. Evaluation of LoG edge maps using Table 6 measures.....43
Table 23. Evaluation of Canny edge maps using Table 5 measures.....45
Table 24. Evaluation of Canny edge maps using Table 4 measures.....45
Table 25. Evaluation of Canny edge maps using FoMs given in Table 6.....46
Table 26. Evaluation of Canny edge maps using Table 6 measures.....46
Table 27. Evaluation of Sobel edge maps using Table 5 measures.....50
Table 28. Evaluation of Sobel edge maps using Table 4 measures.....50
Table 29. Evaluation of Sobel edge maps using FoMs given in Table 6.....51

Table 30. Evaluation of Sobel edge maps using Table 6 measures.....51
Table 31. Evaluation of Roberts edge maps using Table 5 measures.....53
Table 32. Evaluation of Roberts edge maps using Table 4 measures.....53
Table 33. Evaluation of Roberts edge maps using FoMs given in Table 6.....54
Table 34. Evaluation of Roberts edge maps using Table 6 measures.....54
Table 35. Evaluation of Prewitt edge maps using Table 5 measures.....56
Table 36. Evaluation of Prewitt edge maps using Table 4 measures.....56
Table 37. Evaluation of Prewitt edge maps using FoMs given in Table 6.....57
Table 38. Evaluation of Prewitt edge maps using Table 6 measures.....57
Table 39. Evaluation of LoG edge maps using Table 5 measures.....59
Table 40. Evaluation of LoG edge maps using Table 4 measures.....59
Table 41. Evaluation of LoG edge maps using FoMs given in Table 6.....60
Table 42. Evaluation of LoG edge maps using Table 6 measures.....60
Table 43. Evaluation of Canny edge maps using Table 5 measures.....62
Table 44. Evaluation of Canny edge maps using Table 4 measures.....62
Table 45. Evaluation of Canny edge maps using FoMs given in Table 6.....63
Table 46. Evaluation of Canny edge maps using Table 6 measures.....63

ABBREVIATIONS

MSE	Mean Square Error
RMSE	Root Mean Square Error
PSNR	Peak Signal-to-Noise Ratio
MAE	Mean Absolute Error
SSIM	Structure Similarity Index Measure
B_{SNR}	Binary Signal-to-Noise Ratio
FOM	Figure of Merit
SSR	Segmentation Success Ratio
$P_m^{'}$	Complemented Performance Measurement
$\Phi^{'}$	Complemented Φmeasure
$F_\alpha^{'}$	Complemented α measure
Over	Over-detection error
Under	Under-detection error
Loc	Localization error
FOM_{pr}	Pratt's Figure of Merit
FOM_{pi}	Pinho's Figure of Merit
D_p	Panetta's edge map quality measure
H	Hausdorff distance measure
D^k	Distance to Ground Truth
Y	Yasnoff Measure
f_2d_6	Maximum distance
S^k	Symmetric distance
Ψ	Symmetric distance measure
LoG	Laplacian of Gaussian

ABSTRACT

Study on edge detection with color-to-grayscale algorithms

Ijaz Ahmad
Advisor: Prof. Seokjoo Shin , PhD
Department of Computer Engineering,
Graduate School of Chosun University

In image processing, color-to-grayscale conversion techniques are used to simplify the algorithms and to reduce the computational cost for several tasks. Edge detection is one of the widely studied areas in image processing, however, the evaluation of edge detectors with color-to-grayscale conversion algorithms have not been considered. In this study, I have evaluated the effect of various color-to-grayscale conversion algorithms on the performance of edge detection process. Experimental results show that edge detection is not only dependent on the detector used but also on the color-to-grayscale conversion algorithms. The *Lightness* grayscale representation performed better for first order derivative edge detectors. For edge detectors (e.g, Canny or Laplacian of Gaussian edge detection), involving de-noising step, the *Gleam* conversion algorithm performed better. Overall, the gamma corrected grayscale representations results in better edge maps. The results are confirmed with different reference-based objective measures including probabilistic measures and figure of merits for edge detection evaluation.

요약

컬러-회색 스케일 알고리즘의 에지 검출에 대한 연구

Ijaz Ahmad
Advisor: Prof. Seokjoo Shin , PhD
Department of Computer Engineering,
Graduate School of Chosun University

이미지 처리 기술에서 색상 대 그레이 스케일 변환 기술은 알고리즘을 단순화하고 여러 작업의 계산 비용을 줄이기 위해 사용됩니다. 엣지 검출은 이미지 처리 분야에서 널리 연구되고 있는 분야 중 하나이지만, 컬러 대 그레이 스케일 변환 알고리즘을 사용하는 엣지 검출 기술은 고려되지 않았습니다. 본 연구에서는 다양한 색상 대 그레이 스케일 변환 알고리즘이 에지 감지 프로세스의 성능에 미치는 영향을 평가했습니다. 실험 결과는 에지 검출이 사용된 검출기의 방식뿐만 아니라 컬러-그레이 스케일 변환 알고리즘에도 의존함을 보여줍니다. 밝기(Lightness) 그레이 스케일 표현은 1 차 미분 에지 검출기에서 더 잘 수행됩니다. 잡음 제거 단계를 포함하는 에지 검출기 (예를 들어, Canny 또는 Laplacian of Gaussian edge detection)의 경우, Gleam 변환 알고리즘이 더 잘 수행됩니다. 전반적으로 감마 보정 된 그레이 스케일 표현은 더 우수한 엣지 맵을 생성합니다. 또한 성능 평가 결과는 확률론적 측정 및 경계 검출 평가를 위한 figure of merits를 포함하는 다양한 참조 기반 오브젝티브 측정으로 확인되었습니다.

I. Introduction

Partitioning an image into regions representing different objects is the first step in the analysis and understanding of image [3]. It prepares an image as an input to an automatic visual system [1]. It aids image processing tasks by substantially reducing the amount of information to be processed [15]. The reduction is done by extracting information from an input in such a way that the resultant image contains much fewer information than the original image [1][16]. And the extracted information is much more relevant to the other modules of an automatic vision system [1]. The partitioning may be done by grouping pixels that have some common characteristics. For example, pixels having similarity in brightness, or color, may indicate that they are part of the same object or facet of an object [1].

There are two approaches to partition an image into regions: Segmentation and Edge detection [1][3]. In segmentation, those pixels are grouped together which correspond to an object. Further, they are marked to indicate that these pixels belong to the same region. Some criterion that distinguishes pixels from the rest of the image is used, to assign pixels to regions. Two very important principles in segmentation are value similarity and spatial proximity. Two pixels having similar intensity characteristics or if they are close to one another may be assigned to the same region. In edge detection, pixels that lie on a region boundary are found in order to perform image partitioning. These pixels are called edges and can be found by looking at neighboring pixels. Since edge pixels lies on the boundary, gray values may vary in the regions on either

side of the boundary, the difference between neighboring pixels may be measured to find a region boundary. Intensity characteristics is widely used by edge detectors as the basis for edge detection, although derived characteristics, such as texture and motion, may also be used [3].

The history of computational edge detection is extremely rich [10]. A variety of edge detection algorithms have been proposed since the beginning of 60's for both color and grayscale or monochromatic images [4]. However, most of the algorithms refer to grayscale images. One reason is that for many years color digital cameras and internal storage of large arrays associated with multi-spectral data were expensive [1][2]. A lot of image processing techniques were developed around the type of image that was available [1]. Therefore, edge detection in gray-level or monochromatic images received more attention and is well-established than edge detection in color images. Another reason is the conventional edge detection algorithms for grayscale images can be extended for color image edge detection either by applying them to each band separately [1] or by converting the color images to grayscale in advance [9]. However, information loss will occur during the process [9][19]. Finally, the challenge of extracting additional color information without incurring large complexity in the system [7].

The extension of conventional edge detection algorithms for color images; by applying them to each band separately (monochromatic-based techniques), or by treating the channels of color images holistically (vector-valued techniques), leads to high computational complexity or complicates the algorithm,

respectively [6][19]. To minimize the computational cost and simplify the complexity of edge detection algorithms on color images, instead of full color images, grayscale images are used for edge detection [19]. The grayscale representations are obtained by using weighted sum methods. However, it is assumed that color-to-grayscale algorithms are of little importance when using a robust edge detector. Therefore, grayscale conversion methods other than weighted sum methods have not been considered.

1.1. Motivation

The impact of color-to-grayscale conversion algorithms is first studied by Kanan and Cottrell [38]. They have shown that efficiency of image recognition system is also dependent on grayscale representations of input color image. They conducted experiments on four different descriptors across four different datasets. Their results have shown that Gleam is almost always the best performer for object and face recognition, while for texture recognition, Luminance' and Luminance are good choices. Further, they have shown that methods that incorporate gamma correction usually performed better than purely linear methods. In addition, they also proposed to evaluate edge-based and gradient-based descriptors as a future work.

Many conversion methods have been proposed in computer vision. Every method results in different grayscale representations. In addition, conversion methods map a three-dimensional color pixel value to a single value, the information loss is also different [9][19]. Therefore, it is necessary to

evaluate these methods for edge detection.

The grayscale conversion methods that incorporate gamma correction techniques introduces non-linearity to grayscale representation and roughly distributes the tones across the entire range in a uniform way. Since, edge detection is the process of finding sharp change in intensity of image, gamma corrected conversion methods may result in efficient edge detection.

1.2. Thesis Contribution

In color image edge detection, grayscale representation as the first step of the process simplifies the algorithm and reduces computational requirements. I have reviewed grayscale conversion algorithms, paying special attention to color image edge detection techniques that involve color-to-grayscale conversion algorithms, and edge detection evaluators; both subjective and objective edge measures. Moreover, I have reviewed the studies evaluating performance of edge detectors in the presence of noise.

Evaluating the performance of an edge detector is a non-trivial task. In this work, I have evaluated edge detectors in terms of detected edge and non-edge elements, comparing the output of an edge detector with that generated by human, which is considered as ground truth. In this way confusion matrices; assessment based on the common edge/non-edge points presence, and assessment involving distance between actual and ideal edge pixels. To measure the performance of an edge detector is an open problem. In the literature many performance measures have been proposed. However, there seems to be no agreement

on the best option. In the thesis, a chapter is dedicated to the edge evaluators, in which I reviewed the proposed evaluation measures and pointed out their features. Moreover, a best edge detector, based on how close its result is to human evaluation, is selected.

In the experiments, I have used 40 natural images from the Berkeley Segmentation Dataset (BSDS) [39]. These images have a resolution of 481x321 pixels. In addition, each of them comes together with 5-10 hand-made segmentations. Since those segmentations are provided in the shape of region boundaries, they can be used as ground truth for the quality assessment of the edge detection results. Out of 5-10 segmentation images I have chosen the one with many segments in-order to test the performance of methods to their best.

II. Background

2.1. Edge

Edge detection is one of the crucial components image recognition systems, object detectors, image segmentation algorithms and image analysis operations. Both in biological vision and in computer vision, edges are often essential clues toward the interpretation and analysis of image information [19].

Usually, an edge is interpreted as a physical one, caused either as a result of difference in the inherent material properties of the objects or by the shapes of physical objects in three dimensions. Physical edges can be divided into two types and can be described in geometric terms as: (1) an abrupt change in local orientation along a set of points of a physical surface and (2) a boundary between two or more materially different regions of a physical surface defined by a set of points [19].

Image edges have a different notion than physical edges, because an image is a projection of an imaging device from a 3D scene to a two dimensional representation, with respect to the viewpoint of the imaging device. The precise definition of an edge is application domain specific. Generally, an edge can be defined as a set of points along a boundary or contour that separates adjacent regions in an image which have relatively distinct attributes according to some feature of interest. The most common feature for edge detection is gray level or luminance. However, sometimes, other features like reflectance, color, or texture are used. In most applications, where luminance is of prime

importance, then locations of abrupt gray level change are the edge pixels [19].

2.1.1. Gray Edge

In grayscale images, pixel locations of abrupt gray-level change are considered as edges. A gradient can describe change in the image function, that points in the direction of the largest growth of the image function. Therefore, one approach to edge detection is to find the gradient vector magnitude at pixel locations. This method performs better (e.g. for a step function), when the gray-level transition is quite abrupt. However, when the transition region gets wider then second-order derivatives like the Laplacian is more advantageous to utilize. In such case, zero-crossings in the results can define the potential edge pixel locations [6].

In gray-level images, typically brightness discontinuities have been used to model edges. Therefore, it can be said that for two regions which are significantly different in brightness values, an edge is the apparent boundary. The significant difference between two regions may depend, for example on local pixel brightness statistics. This variation occurs due to the fact that an edge, between two objects having different intensities, usually represents a physical boundary. The word edge is used to refer to a location on the image where the brightness value appears to change abruptly. These abrupt changes are associated with high values of the first derivative [12].

2.1.2. Color Edge

The core difference between gray-level images and color images is that, in

a gray-level images a scalar value is assigned to a pixel, while a color vector (which consists of three components/channels) is assigned to a pixel in color images. Thus, in color image processing, instead of scalar image functions (like in grayscale image processing), vector-valued image functions are treated [6].

While in gray-level images an edge is indicated as a discontinuity in the gray-level function, the term "color edge" has not been clearly described for color images. Several different color edges definitions have been proposed. A very old definition states that in the color image an edge exists precisely if the intensity image contains an edge. However, possible discontinuities in the hue or saturation values have been overlooked in this definition. For example, in a color image, if two objects of various colors but equally lighted are arranged in juxtaposition, then this technique cannot be used to find the edge that determines the object geometry [6].

Another definition for a color edge states that if at least one of the color components have an edge then in the color image an edge exists. However, merging the edge detection results from the color components may cause problems because of edges localization inaccuracies in the individual color channels. A third monochromatic-based definition for color edges is based on the calculation of the sum of absolute values of the gradients for three color components [6].

The notion of an edge for color images is much more complex than in grayscale images. In color images, intensity, hue and saturation of a color all play a part in determining object boundaries. A physical boundary produces an

edge which needs to be determined using a measure in an appropriate color space that would capture these different color characteristics. The concept of color similarity within the context of a color space becomes important since intensity of pixels alone cannot be used to determine the occurrence of an edge. [12]

2.1.3. Edge Detection

The process of determining which points in the image are the edge pixels is called edge detection. The edge detection process results in a new image called edge map, which has much lesser information as compared to the original image. The edge map describes each original pixel's edge classification and some additional attributes, such as edge magnitude and edge orientation [19].

One of the reasons to show importance of edge detection is that the success of higher level image processing tasks is heavily dependent on good edges. An image contains enormous amount of data, much of which is irrelevant to some applicaitons, e.g., the background of the scene. So at the initial stage, the effort is made to reduce some of the data; the objects are separated from the background and entities such as edges which are physically significant are identified [20].

In the stereo techniques, for instance, images taken from two different locations are utilized to get the depth information. All stereo techniques deal with the "correspondence", where the tokens from the right and left image are matched in order to get the disparity of two images. The edge information has a key role in the tokens selection. In motion, moving objects are detected by

identifying the time varying edges and corners. The approaches to structure from motion, through which the three dimensional structure of objects is computed, also require correspondence of tokens. Object recognition methods based on only the two dimensional shape use the edge information. Most algorithms for recognizing the partially occluded objects assume the presence of good edges [20].

Most edge detection schemes consist of three stages (and in the given order): filtering, differentiation, and detection. In the filtering stage, noise is removed by passing the image through a filter. Noise can be due to the undesirable effects introduced in the sampling, quantization, blurring and defocusing of the camera, and irregularities of the surface structure of the objects. In the differentiation stage, the image locations with significant intensity changes are highlighted. Finally, in the detection stage, those points are localized where the intensity changes are significant [20].

2.2. Color-to-Grayscale Conversion

Grayscale representation of a color image simplifies the image processing algorithms and reduces computational requirements [38]. Color-to-grayscale conversion is one of the image processing applications used in different fields effectively. In publication organizations' printing, a color image is expensive compared to a grayscale image. Thus, color images have to be converted to grayscale image in-order to reduce the printing cost for low priced edition books. The monochrome devices that receive color imagery must perform a

conversion from color to grayscale. Similarly, color deficient viewer requires good quality of grayscale image to perceive the information, as the normal people perceive the color picture. Likewise, various image processing applications e.g. edge detection, require conversion of color image to grayscale image for reducing the computational complexity.

Conversion of a color image to grayscale image requires more knowledge about the color image. In color image, a pixel is combination of three colors Red, Green, and Blue (RGB). The RGB color values are represented in three dimensions, given by the attributes of lightness, chroma, and hue. The quality of a color image depends on the color represented by the number of bits. The basic color image is represented by 8 bits, the high color image is represented by 16 bits, the true color image is represented by 24 bits, and the deep color image is represented by 32 bits. The number of bits decides the maximum number of different colors supported by a digital device. For example, if each Red, Green, and Blue occupies 8 bits then the combination of RGB will represent 16,777,216 different colors. The grayscale image is represented by 8 bits value and ranges from 0 to 255. The conversion of a color image into a grayscale image is mapping the RGB intensities to a single value of grayscale.

2.2.1. Color-to-Grayscale Conversion Algorithms

The simplest color-to-grayscale algorithm is Intensity. It is the mean of the RGB channels.

$$G_{intensity} = (R+G+B)/3$$

Although linear channels are used in Intensity grayscale representation, in practice, when using datasets containing gamma corrected images, gamma correction is correction is incorporated in the method. The method is called Gleam.

$$G_{gleam} = (R' + G' + B')/3$$

Luminance is designed to match human brightness perception using a weighted combination of the RGB channels. Luminance is the standard algorithm for gray-level representation. It is used by image processing software like, GIMP. It is implemented in "rgb2gray" color-to-grayscale conversion function of MATLAB.

$$G_{luminance} = 0.3*R + 0.59*G + 0.11*B$$

Luma is similar weighted sum method to Luminance and include gamma corrected form used in high-definition televisions.

$$G_{luma} = 0.2126*R' + 0.7152*G' + 0.0722*B'$$

Lightness is perceptually uniform gray-level representation used in the CIELAB and CIELUV color spaces. This means an increment in Lightness should more closely correspond to human perception. This is achieved via a nonlinear transformation of the RGB color space.

$$G_{lightness} = 1/100(116f(Y) - 16),$$

$$where\ Y = 0.2126R + 0.7152G + 0.0722B, \text{and}$$

$$f(t) = \begin{cases} t^{1/3} & \text{if } t > (6/29)^3 \\ 1/3(29/6)^2 t + 4/29 & otherwise \end{cases}$$

Value is the decomposition method. It is the achromatic channel in Hue, Saturation, and Value (HSV) color space. Value provides absolute brightness information. It is calculated by finding the maximum of the RGB channels,

$$G_{\max - value} = \max(R, G, B)$$

or by calculating the minimum value of the RGB channels.

$$G_{\min - value} = \min(R, G, B)$$

Luster is the desaturation method. It is the L channel from the HLS (Hue, Lightness, and Saturation) color space. It is the average of the maximum and

minimum RGB values. The sensitive of Luster to changes in brightness is lesser than Value, because Value is maximized with any fully saturated primary color, but to maximize Luster, all three channels must be fully saturated.

$$G_{luster} = (\max(R,G,B) + \min(R,G,B))/2$$

Three additional conversion techniques, i.e. the pure color channels (R,G and B). The idea is to use a grayscale that is an actual output of the camera and not a function of the values given by the camera.

$$G_{red-channel} = ExtractR(R, G, B)$$

$$G_{green-channel} = ExtractG(R, G, B)$$

$$G_{blue-channel} = ExtractB(R, G, B)$$

III. Edge Detectors Evaluation

In computer science, all systems, especially automated information processing structures, must be evaluated before being developed, for several applications [18]. Edge detection is no exception to this rule. One evaluation of edge detection is to analyze the performance of existing edge detectors solely. However, edge detection in color images, the image is processed by a color-to-grayscale conversion algorithm before input to an edge detector. Therefore, it is necessary to study the impact of color-to-grayscale conversion algorithms along with edge detectors on edge detection.

In image processing, the evaluation methods can usually be categorized into subjective and objective evaluation [7]. Subjective evaluation is the quantitative comparisons made by experts in that area using subjective ratings [14]. This approach avoids the issue of pixel-level ground truth. Thus, it does not give any statement about the frequency of false positive and false negative errors at the pixel level. In addition, this method cannot be automated for human vision systems [15]. However, in most image processing applications human evaluation is the final step [7]. Objective evaluation is the qualitative comparisons obtained from probabilistic measures and figure of merits [16]. Objective edge map evaluation measures have many important uses. Obviously, after the development of so many edge detection algorithms over the years, there should be an objective way of determining which edge detection algorithm generally works best or which edge detection algorithm should be used for a certain application [14][15][16]. Secondly, an objective edge map evaluation

measure can select the tuneable parameters that exist in many edge detection algorithms. Assessing the performance of edge detection algorithms is difficult because the performance depends on several factors [14][15][16]. Objective measures are classified as non-reference and reference measures based on the use of ground truth image.

3.1. Non-Reference Objective Measures

A non-reference edge map evaluation measures does not require a reference image as the ground truth [14][15]. It only utilizes the information from the obtained edge map and the original image itself to make the evaluation. However, in their current stage, the use of non-reference edge measures is still restricted because edge quality evaluation is very subjective and difficult to quantify without a ground truth [14]. Example of non-reference-based edge map evaluation measures include probabilistic measures, and edge connectivity and width uniformity measures [15][16]. The obvious advantage of non-reference-based edge map evaluation is that no ground truth is necessary. Therefore, they can be used to evaluate the performance of edge detectors on non-synthetic images [16]. However, in their current stage, such non-reference-based edge map evaluation measures suffer from many biases and are therefore highly unreliable [15][16]. They cannot effectively incorporate the original image data [14].

Probabilistic measures using receiver operating characteristics attempt to estimate an ideal edge map, or ground truth, from multiple edge detector outputs [15][16]. These measures suffers from its bias regarding how the estimated

ground truth image is generated, as the candidate edge maps used directly affect the estimated ground truth which is determined [16]. Therefore, if the quality of majority edge maps being used is not adequate or fail to extract certain edge structures, this will be reflected in the resultant estimated ground truth [16]. Edge connectivity and width uniformity measures assess the quality of edge detector outputs in terms of the formation of proper edge lines [16]. However, it does not take the original data into account [15][16].

3.2. Reference-Based Objective Measures

Reference-based edge map evaluation measures require an ideal edge map of an image known as the ground truth in addition to the resultant edge map to make their assessment [15][16]. The comparison is made numerically using either of two techniques: counting the number of pixels incorrectly detected (assessment based on confusion matrix) and measuring the localization of these errors (assessment involving distance of misplaced pixels) [37]. A reference-based edge map quality measure requires that a displaced edge should be penalized in function not only of FPs and/or FNs but also of the distance from the position where it should be located [18]. The main advantage of these measures is that they can be used to determine edge detector performance within a controlled environment where edge pixel locations are known [15][16]. Their main disadvantage is that since ground truth must be known, there are a little use for evaluating edge detector performance on natural images. This is because for natural images, determining the ground truth is non-trivial as ideal edge locations are unknown. Therefore, reference-based edge map evaluation measures

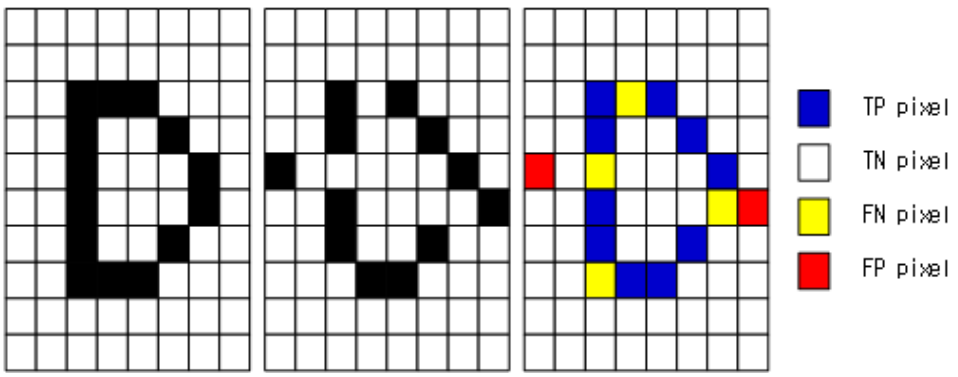
are used predominantly for evaluating edge maps of synthetic images with derived ground truths. Moreover, the performance of edge detectors on synthetic images do not necessarily correlate to their performance on natural images, as edge detection kernels are constructed to detect "real-world" edges taking into consideration noise and deviation from edge models. Reference-based objective edge map evaluation measures include probabilistic measures and figure of merit [16].

3.2.1. Assessment based-on Confusion Matrix

To assess an edge detection algorithm, the confusion matrix remains a cornerstone. Let G_t and D_c be the reference ground truth and D_c the detected contour map of an image /, respectively. Comparing pixel per pixel G_t and D_c, the first criterion to be assessed is the common presence of edge/non-edge points. A basic evaluation is compounded from statistics resulting from a confusion matrix. To that effect, G_t, and D_c are combined. |.| denoting the cardinality of a set. All points are divided into four sets [13][18]:

- True Positive points (TPs), points decided as 'edges' in both G_t and D_c : TP = $|G_t \cap D_c|$,
- False Positive points (FPs), edge points in D_c, which do not coincide with edge points in G_t: FP = $|D_c \cap \neg G_t|$
- False Negative points (FNs), missing boundary points of D_c i.e. holes in the contour: FN = $|\neg D_c \cap G_t|$
- True Negative points (TNs), common non-edge points of G_t and D_c: TN = $|\neg D_c \cap \neg G_t|$.

False positive points appear in the presence of noise, texture or other contours influencing the filter used by the edge detection operator. And False Negative points represent holes in a contour of D_c (generally, caused by blurred edges in the original image /. For example, an incorrect threshold of the segmentation could generate both FPs and FNs.



(a) G_t (b) D_c (c) $G_t \cup D_c$

Figure 1. Illustration of true positives, false negatives and false positives points. (a) Ground Truth, (b) Detected contour map, D_c has 4 FNs and 2 FPs points, illustrated with color in (c).

In Figure 1. TPs, FNs, and FPs are represented in blue, yellow and red, respectively. Computing only FPs and FNs enables a segmentation assessment to be performed. While, combining at least these two quantities enables a segmented image to be assessed more precisely [18]. A list of these measures is presented in Table 4, Table 5 and Table 6.

The **mean square error [34], root mean square error [35]** and **mean absolute error [36]** decreases with the improved quality of detection and vice versa. The

larger **peak signal-to-noise ratio** and **structure similarity index measure** [33] value the better detection quality is. Also, a good-quality edge detection method should obtain the smallest response for **over-detection error, under-detection error** and **localization error** [31]. The over-detection error increases with increase in number of FPs while under-detection increases with increase in FNs points [55].

The **Binary Signal-to-Noise Ratio** [37] is inspired by Signal-to-Noise Ration. It corresponds to a measure comparing the level of signal in a reference signal to the level of the noise in a desired signal. Thus BSNR computes a global error in function of the FPs and FNs; the fewer the numbers of FPs and FNs, the more the score increases and tends to infinity when $G_t = D_c$. The complemented **Performance Measure** [27] considers directly and simultaneously the three entities TP, FP and FN to assess a binary image. The measure is normalized and decreases with improved quality of detection, with Pm = 0 qualifying perfect segmentation [18]. The **Segmented Success Ratio** [28] is obtained by the combination of the over-segmented success ratio and under-segmented success ratio. This measure is normalized in funciton of $|G_t|.|D_c|$ and decreases with improved quality of detection, with SSR = 0 qualifying the perfect segmentation [18].

The **Φ measure** [29] and **F_α measure** [30] are frequently used in edge detection assessment. The complement of these measures resulting a score to 1 indicating a good segmentation and a score to 0 indicating poor segmentation. The F_α remains the most stable because it does not consider the TNs, which are

dominant in edge maps [18]. These two measures are based on True Positive Rate (TPR) and False Positive Rates (FPR), given by:

$$\begin{cases} TPR = \dfrac{TP}{TP + FN} \\ FPR = \dfrac{FP}{FP + TN} \end{cases}$$

These measures evaluate the comparison of two edge images, pixel per pixel, tending to severely penalize even a slightly misplaced contour. These methods suffer greatly when there is a small variations of the binary edge pixel locations, thus resulting large differences in the final ratios [14][18].

3.2.2 Assessment involving Distance of Misplaced Pixels

First, to achieve a quantitative index of edge detector performance, one of the most popular descriptors is the **Pratt's Figure of Merit** (FOM_{pr})[21]. This measure provides a single quantitative index of edge detector performance by comparing the distance between the actual and ideal edge pixels. Pratt's FOM ranges from 0 to 1, where 1 is a perfect match between the edge map being evaluated and the reference ground truth image. The drawback of Pratt's FOM is that it does not record the distance of FNs [14][18]. **Pinho's FOM** (FOM_{pin})[22] suggested an improvement to Pratt's FOM by considering the closest distance between each pixel of detected edge map with the ground truth image. The second term penalizes false edge pixels. **Panetta's edge map quality measure** (D_p)[14] is the average combination of false positive term and false negative term. The false positive term is similar to Pratt's FOM and is the

distance between the false positive edge pixels of detected edge map and its closest ideal edge pixels in the ground truth edge map. The false negative term measures the false negative edge pixels and their distances from the closest correctly detected edge map. The smaller the Dp, the better is the edge pixel detection/localization performance.

A second measure widely used in matching techniques is represented by the **Hausdorff distance (H)**, which measures the mismatch of two sets of points [24]. The algorithm aims to minimize H. A slight variation in the position of a pixel highly effects the score of H [18]. The propositions of this measure are **Distance to G_t (D^k)**[25] and **Yasnoff measure (Y)**[23] which represent an error distance only in function of d_{Gt}. The Yasnoff measure is similar to D^k except k = 2, using a different coefficient 100/||. Both measures estimate the divergence of FPs; in other words, they correspond to a measure of over-segmentation [18].

Finally, a complete and optimum edge detection evaluation measure should combine assessments of both over- and under-segmentation [18], as **Maximum distance measure ($f_2 d_6$)** [26] and **Symmetric distance measure (S^k)** [25]. The score of maximum distance measure represents the maximum between the over- and under-segmentation. The value of symmetric distance measure corresponds to their means. Symmetric distance measure takes small values in the presence of low level of outliers whereas the score becomes large as the level of mistaken points increases. The **Magnier symmetric distance measure (Ψ)** [18] takes into account the distance of all the misplaced pixels: false positives and false negatives. The score is close to 0 concerning good segmentation.

IV. Experimental Results and Analysis

All the experiments are performed in the following environment:

- Processor: Intel® Core(TM) i5-2500 CPU @ 3.30GHz
- RAM: 4.00 GB
- OS: Windows 7, 64-bit
- MATLAB R2016a

For analysis purpose, I have used eleven color-to-grayscale conversion methods listed in Table 1. However, mainly focusing on gamma corrected methods.

Table 1. Color-to-grayscale conversion algorithms

Name	Equation
G_{red}	$ExtractR(R, G, B)$
G_{green}	$ExtractG(R, G, B)$
G_{blue}	$ExtractB(R, G, B)$
$G_{intensity}$	$(R+G+B)/3$
G_{gleam}	$(R'+G'+B')/3$
$G_{luminance}$	$0.3*R+0.59*G+0.11*B$
$G_{max-value}$	$\max(R,G,B)$
$G_{min-value}$	$\min(R,G,B)$
G_{luster}	$(\max(R,G,B)+\min(R,G,B))/2$
G_{luma}	$0.2126*R'+0.7152*G'+0.0722*B'$
$G_{lightness}$	$1/100(116f(Y)-16),$ $where\ Y=0.2126R+0.7152G+0.0722B,$ and $f(t)=\begin{cases} t^{1/3} & \text{if } t>(6/29)^3 \\ 1/3(29/6)^2t+4/29 & otherwise \end{cases}$

I have used three first order derivative edge detectors; Sobel, Roberts and Prewitt. And two second order derivative edge detectors; Laplacian of Gaussian and Canny edge detector. I have used the parameters for these edge detectors suggested in [14]. The edge detectors and the their parameters are listed in Table 2 and Table 3.

Table 2. Parameters used in the edge detection of synthetic images

Edge Detector	Parameter settings
Sobel	threshold = 0.05
Roberts	threshold = 0.05
Prewitt	threshold = 0.05
LoG	Gaussian standard deviation = 2, threshold = 0.003
Canny	Gaussian standard deviation = 1, threshold = [0.005, 0.01]

Table 3. Parameters used in the edge detection of natural images

Edge Detector	Parameter settings
Sobel	threshold = 0.1246
Roberts	threshold = 0.1246
Prewitt	threshold = 0.1246
LoG	Gaussian standard deviation = 2, threshold = 0.0047
Canny	Gaussian standard deviation = 2, threshold = [0.16, 0.4]

I have used reference-based objective measures for evaluation of edge detectors. Table 4. and Table 5. lists probabilistic performance measurement metrics.

Table 4. Performance measurement metrics (A)

$MSE = \frac{1}{m*n}\sum_{i=1}^{m}\sum_{j=1}^{n}[I^{'}(i,j) - I(i,j)]^2$
$RMSE = \sqrt{MSE}$
$PSNR = 20\log_{10}\left[\frac{255}{RMSE}\right]$
$MAE = \frac{1}{m*n}\sum_{i=1}^{m}\sum_{j=1}^{n}[I^{'}(i,j) - I(i,j)]$
$SSIM(X,Y) = \frac{(2\mu_x\mu_y + C_1)*(2\sigma_{xy} + C_2)}{(\mu_x^2\mu_y^2 + C_1)*(\sigma_x^2 + \sigma_y^2 + C_2)}$

Table 5. Performance measurement metrics (B)

$B_{SNR}(G_t, D_c) = \sqrt{\frac{\lvert D_c\rvert}{FP + FN}}$
$P_m^{'}(G_t, D_c) = 1 - \frac{TP}{TP + FP + FN}$
$SSR(G_t, D_c) = 1 - \frac{TP^2}{\lvert G_t\rvert . \lvert D_c\rvert}$
$\Phi^{'}(G_t, D_c) = 1 - \frac{TPR \,.\, TN}{TN + FP}$
$F_\alpha^{'}(G_t, D_c) = 1 - \frac{P_{REC} \,.\, TPR}{\alpha \,.\, TPR + (1-\alpha) \,.\, P_{REC}}$, $with\, P_{REC} = \frac{TP}{TP+FP}\ \alpha \epsilon 0:1$
$Over(G_t, D_c) = \frac{FP}{\lvert I\rvert - \lvert G_t\rvert}$
$Under(G_t, D_c) = \frac{FN}{\lvert G_t\rvert}$
$Loc(G_t, D_c) = \frac{FP+FN}{\lvert I\rvert}$

Table 6. lists performance measures involving distance of misplaced pixels. The distance $d_{G_t}(p)$ represents the minimal distance between p and

G_t, for a pixel belonging to the desired contour $p \epsilon D_c$. While, the $d_{D_c}(p)$ represents the minimal distance between p and D_c, for a pixel belonging to the ground truth $p \epsilon G_t$. These distances are given by:

$$dG_{t(p)} = \mathrm{Inf}\sqrt{(\mathrm{x_p}-\mathrm{x_t})^2+(\mathrm{y_p}-\mathrm{y_t})^2}, \mathrm{p}\epsilon \mathrm{D_c} \mathrm{and} t \epsilon G_t,$$
$$dD_{c(p)} = \mathrm{Inf}\sqrt{(\mathrm{x_p}-\mathrm{x_t})^2+(\mathrm{y_p}-\mathrm{y_t})^2}, \mathrm{t}\epsilon \mathrm{D_c} \mathrm{and} p \epsilon G_t$$

The distances refer to Euclidean distance and (x_p, y_p) and (x_t, y_t) the pixel coordinates of two points p and t.

Table 6. Performance Measurement Metrics (C)

$FOM_{pr}(G_t, D_c) = \frac{1}{\max(\lvert G_t\rvert, \lvert D_c\rvert)} \sum_{p \epsilon D_c} \frac{1}{1+\alpha d^2_{G_t}(p)}$
$FOM_{\mathrm{pin}}(G_t, D_c) = \left[\frac{1}{N_o}\sum_{i=1}^{N_o}\frac{1}{1+\alpha d_i^2}\right].\left[\frac{1}{1+\beta\frac{N_{FA}}{N_o}}\right]$
$D_p(G_t, D_c) = \frac{1/2}{\lvert I\rvert - \lvert G_t\rvert}.\sum_{p \epsilon D_c}\left(1-\frac{1}{1+\alpha.d^2_{G_t}(p)}\right) + \frac{1/2}{\lvert G_t\rvert}\sum_{p \epsilon G_t}\left(1-\frac{1}{1+\alpha.d^2_{G_t \cap D_c}(p)}\right)$
$H(G_t, D_c) = \max\left(\max_{p \epsilon D_c} d_{G_t}(p), \max_{p \epsilon G_t} d_{D_c}(p)\right)$
$D^k(G_t, D_c) = \frac{1}{\lvert D_c\rvert}.\sqrt[k]{\sum_{p \epsilon D_c} d^k_{G_t}(p)},\ k=1 \text{ for [18]}$
$Y(G_t, D_c) = \frac{100}{\lvert I\rvert}.\sqrt{\sum_{p \epsilon D_c} d^2_{G_t}(p)}$
$f_2 d_6(G_t, D_c) = \max\left(\frac{1}{\lvert D_c\rvert}.\sum_{p \epsilon D_c} d_{G_t}(p), \frac{1}{\lvert G_t\rvert}.\sum_{p \epsilon G_t} d_{D_c}(p)\right)$
$S^k(G_t, D_c) = \sqrt[k]{\frac{\sum_{p \epsilon D_c} d^k_{G_t}(p) + \sum_{p \epsilon G_t} d^k_{D_c}(p)}{\lvert D_c \cup G_t\rvert}},\ k=1 \text{ for [18]}$
$\Psi(G_t, D_c) = \frac{FP+FN}{\lvert G_t\rvert^2}.\sqrt{\sum_{p \epsilon D_c} d^2_{G_t}(p) + \sum_{p \epsilon G_t} d^2_{D_c}(p)}$

To evaluate the effect of color to grayscale conversion algorithms on the performance of edge detection, experiments are conducted on both synthetic and natural images. Some of the experimental results are shown here. First the images are converted to grayscale representations by using the methods listed in Table 1. Figure 3 and Figure 10 shows the grayscale images of synthetic and natural color image, respectively. The edge maps of these images are obtained by using Sobel, Roberts, Prewitt, Laplacian of Gaussian and Canny edge detectors. Figure 4-8 and Figure 11-15 shows the edge maps of the synthetic grayscale image and natural grayscale image, respectively. The edge detectors, parameters are given in Table 2 and Table 3, applied to synthetic and natural images, repectively. This work focuses on the effect of color to grayscale conversion algorithms on edge detectors, therefore only comparisons between color to grayscale conversion are carried out and therefore, the parameters used for edge detection algorithms are fixed. The comparison between color-to-gryascale algorithms is carried out by evaluating the performance of edge detectors when applied to different grayscale representations. I have used multiple evaluators to get the performance score. The methods are described in chapter III and are listed in Table 4,5, and 6.

For synthetic image, the edge maps obtained from Sobel, Roberts and Prewitt methods, the Lightness grayscale representation out performs the other grayscale conversion methods. For the Laplacian edge maps, Intensity grayscale image performs better. The Luminance grayscale representation of the synthetic color image performs better for Canny edge detector. Figure 9 is the commulative

ranking of the number of tests a method qualifies for all edge detectors. The Lightness color-to-grayscale conversion method has the highest rank, followed by Intensity and Luminance conversion methods. However, the Lightness method does not qualify any test for the Laplacian and Canny edge maps, the Intensity method only qualify one test for Roberts method and Luminance method qualifies one test for Roberts and Prewitt edge detectors. The commulative ranking shows that the max-value performs most poorly, followed by min-value and pure RGB channel representation.

For natural images, experiments are carried out on 40 images from Berkley dataset [39]. These images are mixture of people's portraits, animals, landscapes and objects. The results of one natural image (The bridge) is shown in Table 7-26. For complete dataset the results are summarized in Figure 2. Figure 2 is the mean accuracy of each method based on the Panetta et al. reference based edge detection evaluator presented in [14]. I chose this measure because the results obtained from this method is similar to human subjective evaluation.

The experimental results in Figure 2. shows the improved performance of an edge detector when a suitable grayscale representation is considered. The first order derivative edge detectors performed better on Lightness equation of CIE LAB. The Sobel and Roberts edge detectors gained 5% efficiency while Prewitt achieved 6% increase in the performance. However, edge detectors (Laplacian of Gaussian (LoG) and Canny) that incorporate de-noising step, result a better edge map on Gleam grayscale representation. The LoG method achieved 2% accuracy while

Canny edge detector achieved 4% efficiency on Gleam representation.

4.1 Discussion

Pure RGB channel grayscale representation methods choose one channel out of three, therefore resulting in loss of edges that occurs because of the intensity difference in the other two channels. The Intensity method is based on averaging, which is simple and fast. However, averaging result is distorted by very small and very large values. Intensity representation smooths out the variance of pixels intensities, thus resulting in edges loss. Luster is a desaturation method which finds the midpoint between the maximum and the minimum of RGB. This method also results in a flatter and softer grayscale image. The Max-Value and Min-Value are decomposition methods, which forces each pixel to highest or lowest of RGB value, respectively. A maximum decomposition losses darker tones while minimum decomposition losses brighter tones thus results in the corresponding edges loss. None of the aforementioned methods includes parameters to adjust shades of gray relative to the way humans perceive luminosity.

Humans do not perceive all colors equally. Alternative method is to weight each color based on how human eye perceives it. Luminance method have three coefficients that represent the intensity perception of humans. However, this method does not adopt the non-linear behavior of human perception. Therefore, this method is less sensitive to darker tones. Gamma corrected methods are proposed to overcome this problem. Gleam, Luma and Lightness are based on gamma

correction. The Lightness method distributes the tones roughly evenly across the entire range. Thus, resulting in good edge map for first order derivative edge detectors. However, the methods, which have de-noising step smooths out this wide range of intensity distribution and results in poor edge map. On the other hand, Gleam performed better for such type of edge detectors because the gamma correction in the equation introduces the non-linearity but preserved the distribution of intensity in a moderate way, thus de-noising could not smoothened them out. Finally, the Luma method includes the gamma correction parameter but still the resultant variance is high enough that de-noising step toned it out.

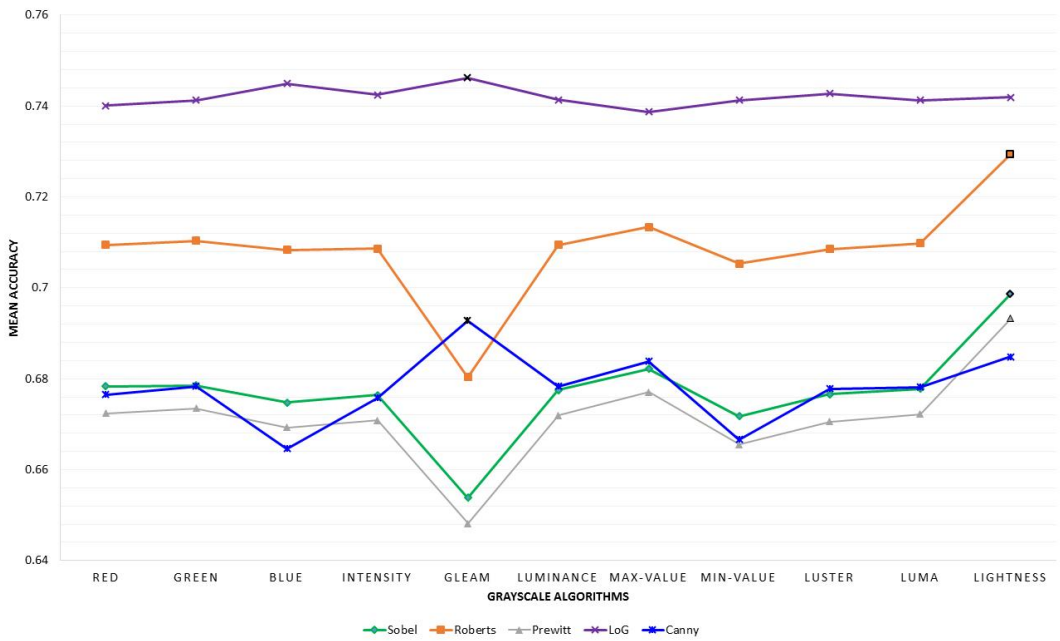
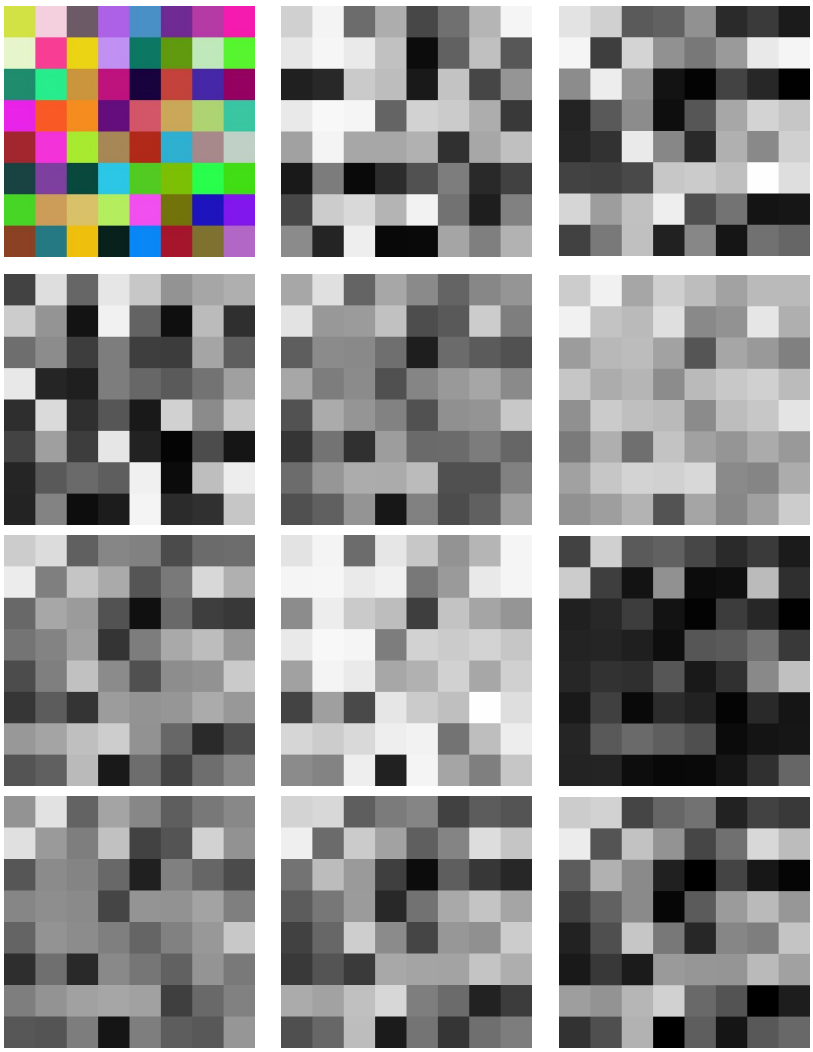


Figure 2. Performance measure of color-to-grayscale algorithms on edge detectors for Berkely natural images dataset. The y-axis is the mean accuracy for each color-to-grayscale algorithm across each edge detector. The conversion methods are given on x-axis.

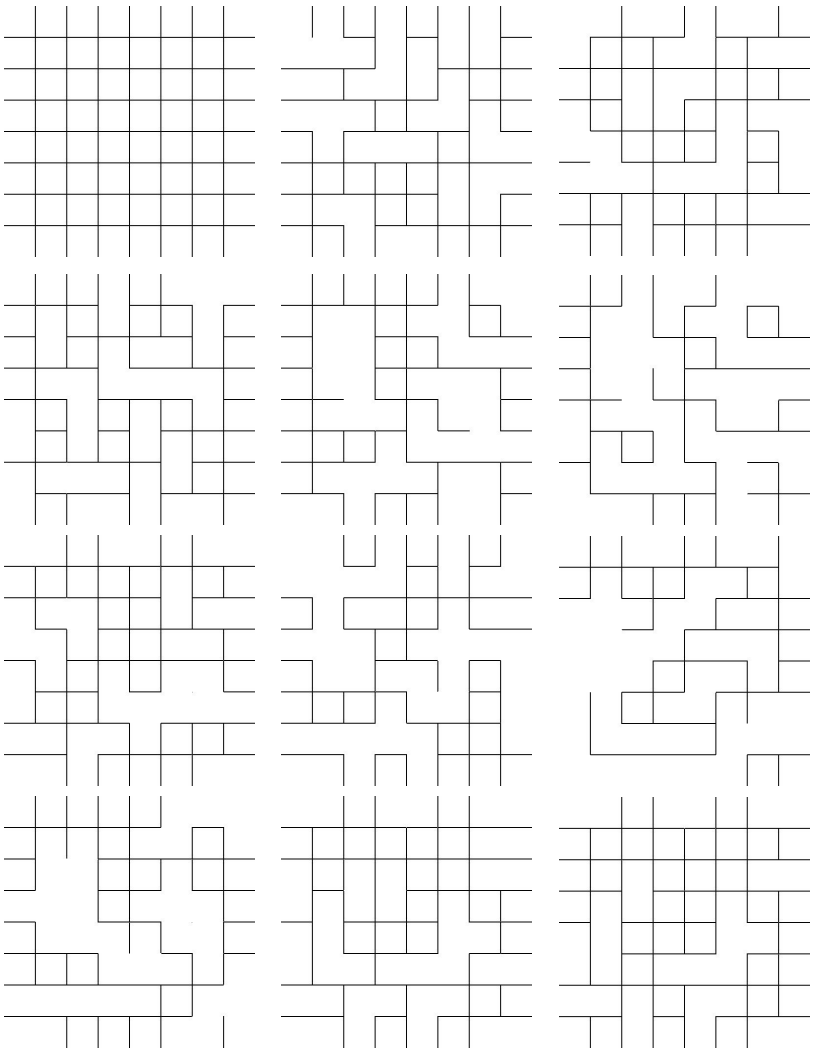
I have used a synthetic color image of check pattern Figure 3. (a). The grayscale representations Figure 3. (b)-(l) of the original image are obtained using the color-to-grayscale algorithms listed in Table 1.



a	b	c
d	e	f
g	h	i
j	k	l

Figure 3. Grayscale representations of synthetic image (a) Original color image. (b)-(l) Grayscale images obtained from methods listed in Table 1.

Figure 4. (a) is the ground truth edge map and (b)-(l) are the corresponding obtained edge maps of Figure 3. (b)-(l), using Sobel edge detection.



a	b	c
d	e	f
g	h	i
j	k	l

Figure 4. Resultant Sobel edge maps (a) Ground Truth. (b)-(l) Edge maps obtained using Sobel edge detection on grayscale images in Figure 3. (b)-(l).

Table 7. Evaluation of Sobel edge maps using Table 5 measures.

Figure 4	BSNR	PM	SSR	Φ	F_α	OVER	UNDER	LOC
b	1.8379	0.2284	0.2284	0.2284	0.2104	0	0.2284	0.0123
c	1.8771	0.2211	0.2211	0.2211	0.2035	0	0.2211	0.0119
d	1.9307	0.2115	0.2115	0.2115	0.1945	0	0.2115	0.0114
e	1.4343	0.3271	0.3271	0.3271	0.3043	0	0.3271	0.0176
f	1.1629	0.4251	0.4251	**0.4251**	**0.3996**	0	0.4251	0.0229
g	1.8294	0.2301	0.2301	0.2301	0.2119	0	0.2301	0.0124
h	1.3782	0.3449	0.3449	0.3449	0.3215	0	0.3449	0.0186
i	1.1835	0.4165	0.4165	0.4165	0.3912	0	0.4165	0.0225
j	1.4318	0.3279	0.3279	0.3279	0.3051	0	0.3279	0.0177
k	1.8778	0.2209	0.2209	0.2209	0.2033	0	0.2209	0.0119
l	**2.2994**	**0.1591**	**0.1591**	0.1591	0.1455	0	**0.1591**	**0.0086**

Table 8. Evaluation of Sobel edge maps using Table 4 measures.

Figure 4	MSE	RMSE	PSNR	MAE	SSIM
b	0.0123	0.1110	67.2242	0.0123	0.9363
c	0.0119	0.1092	67.3663	0.0119	0.9380
d	0.0114	0.1068	67.5580	0.0114	0.9407
e	0.0176	0.1328	65.6651	0.0176	0.9091
f	0.0229	0.1514	64.5267	0.0229	0.8815
g	0.0124	0.1114	67.1934	0.0124	0.9356
h	0.0186	0.1364	65.4347	0.0186	0.9039
i	0.0225	0.1499	64.6150	0.0225	0.8835
j	0.0177	0.1330	65.6548	0.0177	0.9087
k	0.0119	0.1092	67.3691	0.0119	0.9381
l	**0.0086**	**0.0926**	**68.7963**	**0.0086**	**0.9554**

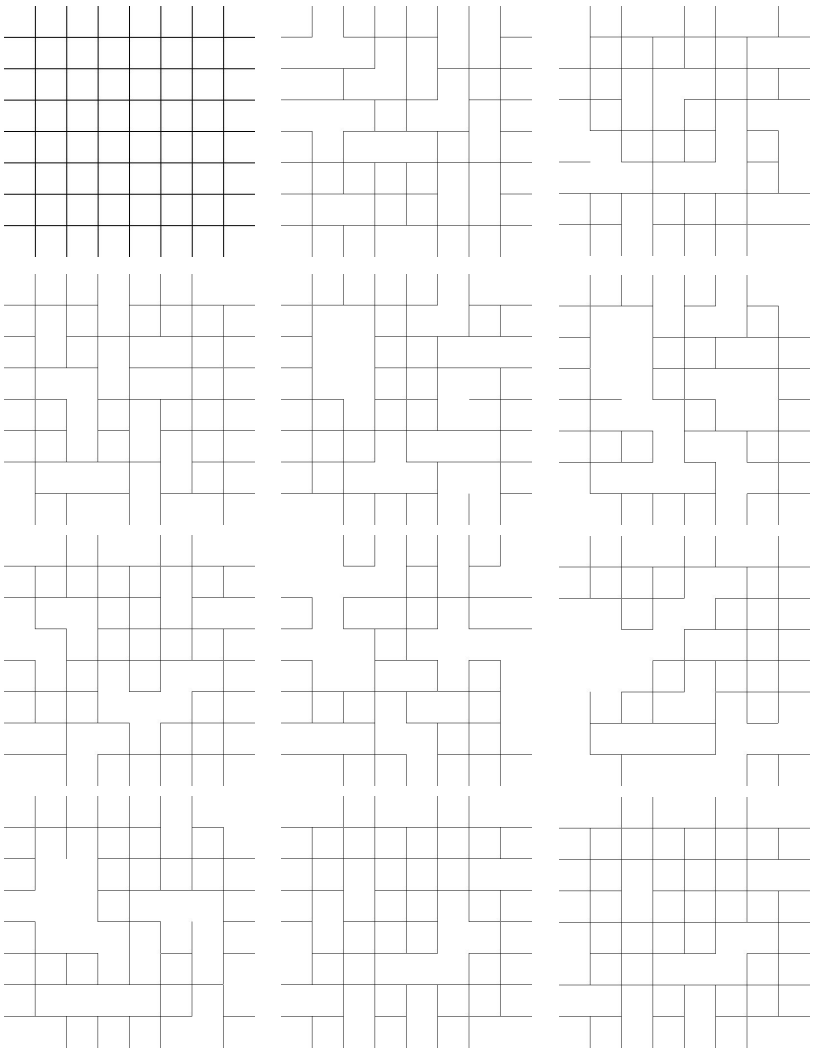
Table 9. Evaluation of Sobel edge maps using Figure of Merits given in Table 6.

Figure 4	FOM_{pr}	FOM_{pin}	D_p
b	0.7716	0.8008	0.9004
c	0.7789	0.8033	0.9017
d	0.7885	0.8133	0.9067
e	0.6729	0.7048	0.8524
f	0.5749	0.6176	0.8088
g	0.7699	0.7962	0.8981
h	0.6551	0.6918	0.8459
i	0.5835	0.6206	0.8103
j	0.6721	0.7059	0.8530
k	0.7791	0.8046	0.9023
l	**0.8409**	**0.8596**	**0.9298**

Table 10. Evaluation of Sobel edge maps using Table 6 measures.

Figure 4	Y	H	D^k	$f_2 d_6$	S^k	Ψ
b	0	**63.0000**	0	3.9384	3.9384	0.0187
c	0	**63.0000**	0	4.6739	4.6739	0.0225
d	0	**63.0000**	0	4.2370	4.2370	0.0199
e	0	63.0079	0	8.3051	8.3051	0.0489
f	0	64.0000	0	**0.0339**	10.0339	0.0673
g	0	63.0079	0	4.6743	4.6743	0.0230
h	0	64.0000	0	7.9091	7.9091	0.0486
i	0	90.5207	0	1.7999	11.7999	0.0792
j	0	64.0000	0	7.5114	7.5114	0.0445
k	0	**63.0000**	0	4.5235	4.5235	0.0218
l	0	**63.0000**	0	3.2506	**3.2506**	**0.0133**

Figure 5. (a) is the ground truth edge map and (b)-(l) are the corresponding obtained edge maps of Figure 3. (b)-(l), using Roberts edge detection.



a	b	c
d	e	f
g	h	i
j	k	l

Figure 5. Resultant Roberts edge maps (a) Ground Truth. (b)-(l) Edge maps obtained using Roberts edge detection on grayscale images in Figure 3. (b)-(l).

Table 11. Evaluation of Roberts edge maps using Table 5 measures.

Figure 5	BSNR	PM	SSR	Φ	F_α	OVER	UNDER	LOC
b	0.8333	0.5902	0.5902	0.5902	0.5645	0	0.5902	0.0318
c	0.8252	0.5949	0.5949	0.5949	0.5693	0	0.5949	0.0321
d	0.8640	0.5762	0.5762	0.5762	0.5466	0	0.5766	0.0309
e	0.8021	0.6085	0.6085	0.6085	0.5831	0	0.6085	0.0328
f	0.7502	0.6399	0.6399	0.6399	0.6153	0	0.6399	0.0345
g	0.8408	0.5859	0.5859	0.5859	0.5601	0	0.5859	0.0316
h	0.7289	0.6530	0.6530	0.6530	0.6288	0	0.6530	0.0352
i	0.7212	0.6579	0.6579	**0.6579**	**0.6338**	0	0.6579	0.0355
j	0.7797	0.6219	0.6219	0.6219	0.5969	0	0.6219	0.0335
k	0.8719	0.5681	0.5681	0.5681	0.5421	0	0.5681	0.0306
l	**0.8880**	**0.5591**	**0.5591**	0.5591	0.5330	0	**0.5591**	**0.0302**

Table 12. Evaluation of Roberts edge maps using Table 4 measures.

Figure 5	MSE	RMSE	PSNR	MAE	SSIM
b	0.0318	0.1784	63.1019	0.0318	0.8347
c	0.0321	0.1791	63.0672	0.0321	0.8332
d	0.0309	0.1757	63.2335	0.0309	0.8394
e	0.0328	0.1812	62.9692	0.0328	0.8298
f	0.0345	0.1858	62.7507	0.0345	0.8211
g	0.0316	0.1778	63.1338	0.0316	0.8357
h	0.0352	0.1877	62.6623	0.0352	0.8173
i	0.0355	0.1884	62.6304	0.0355	0.8158
j	0.0335	0.1832	62.8743	0.0335	0.8260
k	0.0306	0.1751	63.2674	0.0306	0.8406
l	**0.0302**	**0.1737**	**63.3366**	**0.0302**	**0.8430**

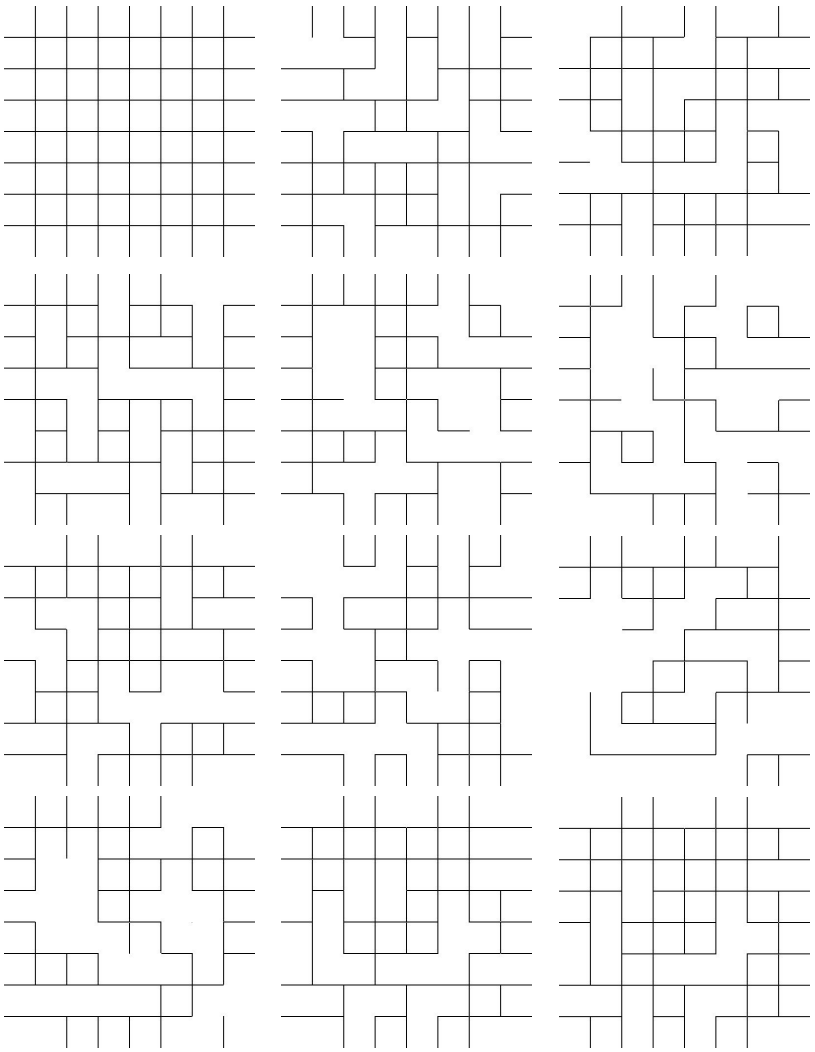
Table 13. Evaluation of Roberts edge maps using FoMs given in Table 6.

Figure 5	FOM_{pr}	FOM_{pin}	D_p
b	0.4098	0.7998	0.8999
c	0.4051	0.7887	0.8944
d	0.4274	0.8276	0.9138
e	0.3915	0.7648	0.8824
f	0.3601	0.7097	0.8549
g	0.4141	0.8048	0.9024
h	0.3470	0.6880	0.8440
i	0.3421	0.6754	0.8377
j	0.3781	0.7407	0.8704
k	0.4319	0.8344	0.9172
l	**0.4409**	**0.8491**	**0.9246**

Table 14. Evaluation of Roberts edge maps using Table 6 measures.

Figure 5	Y	H	D^k	$f_2 d_6$	S^k	Ψ
b	0	**64.0000**	0	3.5948	3.5948	0.0429
c	0	**64.0000**	0	4.5952	4.5952	0.0577
d	0	**64.0000**	0	3.4624	3.4624	0.0457
e	0	**64.0000**	0	5.5694	5.5694	0.0693
f	0	**64.0000**	0	7.3954	7.3954	0.0866
g	0	**64.0000**	0	4.0157	4.0157	0.0511
h	0	**64.0000**	0	7.8428	7.8428	0.0912
i	0	89.8053	0	9.7018	9.7018	0.1125
j	0	**64.0000**	0	6.4299	6.4299	0.0777
k	0	**64.0000**	0	3.4538	3.4538	0.0467
l	0	**64.0000**	0	**3.1728**	**3.1728**	**0.0444**

Figure 6. (a) is the ground truth edge map and (b)-(l) are the corresponding obtained edge maps of Figure 3. (b)-(l), using Prewitt edge detection.



a	b	c
d	e	f
g	h	i
j	k	l

Figure 6. Resultant Prewitt edge maps (a) Ground Truth. (b)-(l) Edge maps obtained using Prewitt edge detection on grayscale images in Figure 3. (b)-(l).

Table 15. Evaluation of Prewitt edge maps using Table 5 measures.

Figure 6	BSNR	PM	SSR	Φ	F_α	OVER	UNDER	LOC
b	1.8364	0.2287	0.2287	0.2287	0.2107	0	0.2287	0.0123
c	1.8751	0.2214	0.2214	0.2214	0.2038	0	0.2214	0.0119
d	1.9307	0.2115	0.2115	0.2115	0.1945	0	0.2115	0.0114
e	1.4339	0.3272	0.3272	0.3272	0.3045	0	0.3272	0.0177
f	1.1619	0.4255	0.4255	**0.4255**	**0.4000**	0	0.4255	0.0230
g	1.8280	0.2303	0.2303	0.2303	0.2122	0	0.2303	0.0124
h	1.3773	0.3452	0.3452	0.3452	0.3218	0	0.3452	0.0186
i	1.1833	0.4166	0.4166	0.4166	0.3913	0	0.4166	0.0225
j	1.4332	0.3274	0.3274	0.3274	0.3047	0	0.3274	0.0177
k	1.8767	0.2211	0.2211	0.2211	0.2035	0	0.2211	0.0119
l	**2.2976**	**0.1593**	**0.1593**	0.1593	0.1457	0	**0.1593**	**0.0086**

Table 16. Evaluation of Prewitt edge maps using Table 4 measures.

Figure 6	MSE	RMSE	PSNR	MAE	SSIM
b	0.0123	0.1111	67.2188	0.0123	0.9363
c	0.0119	0.1093	67.3594	0.0119	0.9379
d	0.0114	0.1068	67.5580	0.0114	0.9407
e	0.0177	0.1329	65.6632	0.0177	0.9091
f	0.0230	0.1515	64.5224	0.0230	0.8814
g	0.0124	0.1115	67.1880	0.0124	0.9356
h	0.0186	0.1365	65.4311	0.0186	0.9038
i	0.0225	0.1499	64.6143	0.0225	0.8835
j	0.0177	0.1329	65.6604	0.0177	0.9087
k	0.0119	0.1092	67.3649	0.0119	0.9381
l	**0.0086**	**0.0927**	**68.7905**	**0.0086**	**0.9554**

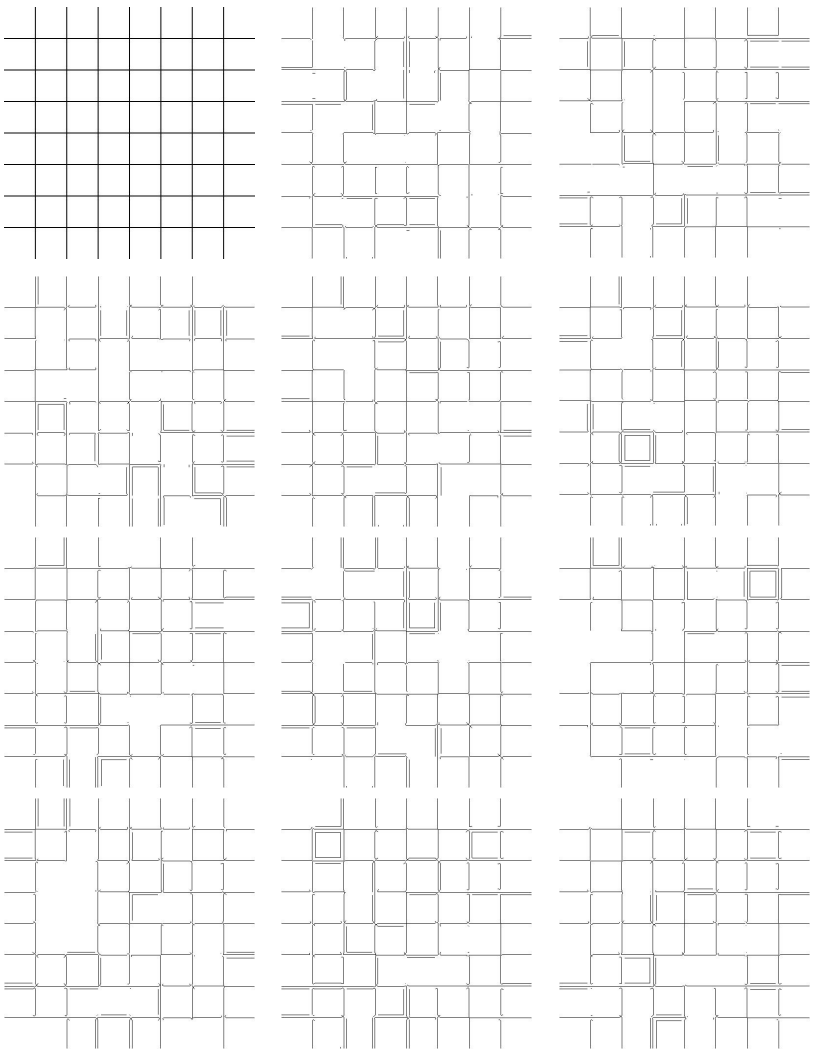
Table 17. Evaluation of Prewitt edge maps using FoMs given in Table 6.

Figure 6	FOM_{pr}	FOM_{pin}	D_p
b	0.7713	0.8008	0.9004
c	0.7786	0.8031	0.9015
d	0.7885	0.8134	0.9067
e	0.6728	0.7050	0.8525
f	0.5745	0.6175	0.8088
g	0.7697	0.7942	0.8971
h	0.6548	0.6916	0.8458
i	0.5834	0.6205	0.8103
j	0.6726	0.7066	0.8533
k	0.7789	0.8047	0.9024
l	**0.8407**	**0.8596**	**0.9298**

Table 18. Evaluation of Prewitt edge maps using Table 6 measures.

Figure 6	Y	H	D^k	$f_2 d_6$	S^k	Ψ
b	0	**63.0000**	0	3.9387	3.9387	0.0187
c	0	**63.0000**	0	4.6795	4.6795	0.0226
d	0	**63.0000**	0	4.2368	4.2368	0.0199
e	0	63.0079	0	8.3047	8.3047	0.0490
f	0	64.0000	0	10.0438	10.0438	0.0674
g	0	**63.0000**	0	5.2267	5.2267	0.0257
h	0	64.0000	0	7.9146	7.9146	0.0486
i	0	90.5207	0	11.8132	11.8132	0.0793
j	0	64.0000	0	7.5055	7.5055	0.0444
k	0	**63.0000**	0	4.5234	4.5234	0.0218
l	0	**63.0000**	0	**3.2508**	**3.2508**	**0.0133**

Figure 7. (a) is the ground truth edge map and (b)-(l) are the corresponding obtained edge maps of Figure 3. (b)-(l), using LoG edge detection.



a	b	c
d	e	f
g	h	i
j	k	l

Figure 7. Resultant LoG edge maps (a) Ground Truth. (b)-(l) Edge maps obtained using Laplacian of Gaussian edge detection on grayscale images in Figure 3. (b)-(l).

Table 19. Evaluation of LoG edge maps using Table 5 measures.

Figure 7	BSNR	PM	SSR	Φ	F_α	OVER	UNDER	LOC
b	0.8487	0.6274	0.6664	**0.5972**	**0.5736**	**0.0049**	0.5953	0.0368
c	0.8649	0.6242	0.6672	0.5888	0.5654	0.0057	0.5865	0.0370
d	0.8860	0.6257	0.6770	0.5792	0.5565	0.0075	0.5762	0.0382
e	0.9071	**0.5919**	**0.6288**	0.5590	0.5350	**0.0049**	0.5569	**0.0347**
f	0.9154	0.5969	0.6402	0.5562	0.5327	0.0061	0.5535	0.0357
g	0.8982	0.6033	0.6447	0.5664	0.5428	0.0056	0.5641	0.0358
h	0.8700	0.6331	0.6836	0.5896	0.5668	0.0072	0.5867	0.0385
i	0.8500	0.6279	0.6676	0.5968	0.5732	0.0051	0.5948	0.0369
j	0.8721	0.6203	0.6634	0.5842	0.5607	0.0058	0.5818	0.0369
k	**0.9306**	0.6074	0.6618	0.5508	0.5284	0.0088	**0.5470**	0.0378
l	0.8994	0.6019	0.6427	0.5654	0.5417	0.0056	0.5631	0.0356

Table 20. Evaluation of LoG edge maps using Table 4 measures.

Figure 7	MSE	RMSE	PSNR	MAE	SSIM
b	0.0368	0.1917	62.4772	0.0368	0.8055
c	0.0370	0.1925	62.4440	0.0370	0.8018
d	0.0382	0.1955	62.3084	0.0382	0.7945
e	**0.0347**	**0.1862**	**62.7320**	**0.0347**	**0.8178**
f	0.0357	0.1888	62.6090	0.0357	0.8113
g	0.0358	0.1891	62.5965	0.0358	0.8112
h	0.0385	0.1961	62.2794	0.0385	0.7956
i	0.0369	0.1920	62.4637	0.0369	0.8052
j	0.0369	0.1920	62.4660	0.0369	0.8048
k	0.0378	0.1944	62.3555	0.0378	0.7953
l	0.0356	0.1887	62.6132	0.0356	0.8119

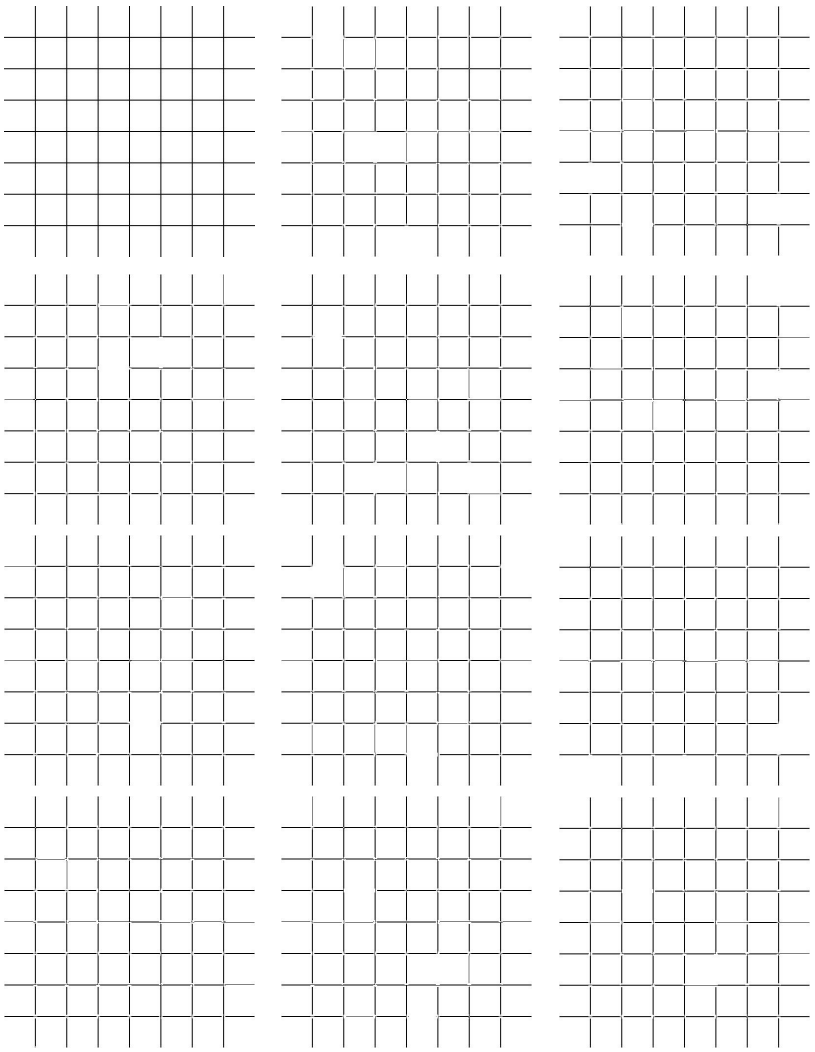
Table 21. Evaluation of LoG edge maps using FoMs given in Table 6.

Figure 7	FOM_{pr}	FOM_{pin}	D_p
b	0.4381	0.8955	0.9088
c	0.4497	0.9185	0.9139
d	0.4680	0.9689	0.9240
e	0.4768	0.9660	0.9427
f	0.4858	**0.9925**	0.9449
g	0.4718	0.9622	0.9347
h	0.4576	0.9484	0.9168
i	0.4372	0.8907	0.9061
j	0.4561	0.9289	0.9184
k	**0.5052**	1.0449	**0.9492**
l	0.4715	0.9614	0.9351

Table 22. Evaluation of LoG edge maps using Table 6 measures.

Figure 7	Y	H	D^k	$f_2 d_6$	S^k	Ψ
b	2.2907	57.0088	0.7588	2.6557	2.7880	0.0396
c	2.4380	64.0000	0.8816	2.9439	3.0874	0.0491
d	2.7377	64.0000	1.1424	2.4175	2.6961	0.0450
e	**2.1477**	**32.0000**	**0.7036**	1.5346	**1.7561**	**0.0258**
f	2.3902	61.1310	0.8860	1.5041	1.8012	0.0293
g	2.4005	64.0000	0.8526	1.9881	2.2241	0.0367
h	2.7177	59.0085	1.0993	2.5555	2.7953	0.0443
i	2.4409	64.0312	0.8247	3.7794	3.8455	0.0613
j	2.4323	64.0000	0.8745	3.7124	3.7831	0.0663
k	2.6441	58.4209	1.1934	**1.3116**	1.7644	0.0287
l	2.4356	59.0085	0.8584	2.1088	2.3396	0.0393

Figure 8. (a) is the ground truth edge map and (b)-(l) are the corresponding obtained edge maps of Figure 3. (b)-(l), using Canny edge detection.



a	b	c
d	e	f
g	h	i
j	k	l

Figure 8. Resultant Canny edge maps (a) Ground Truth. (b)-(l) Edge maps obtained using Canny edge detection on grayscale images in Figure 3. (b)-(l).

Table 23. Evaluation of Canny edge maps using Table 5 measures.

Figure 8	BSNR	PM	SSR	Φ	F_α	OVER*10^{-5}	UNDER	LOC
b	2.7265	0.1186	0.1187	0.1183	0.1077	2.41	0.1182	0.0064
c	2.7896	0.1139	0.1140	0.1135	0.1033	3.22	0.1134	0.0061
d	3.0140	0.0992	0.0992	**0.9900**	0.0900	1.61	0.0989	0.0054
e	2.7400	0.1176	0.1176	0.1173	0.1069	1.61	0.1173	0.0063
f	2.7816	0.1145	0.1146	0.1141	0.1039	2.82	0.1141	0.0062
g	**3.1086**	**0.0938**	**0.0938**	0.0935	0.0850	2.01	**0.0935**	**0.0051**
h	2.5290	0.1353	0.1353	0.1350	0.1232	1.61	0.1350	0.0073
i	2.7318	0.1182	0.1182	0.1179	0.1074	2.01	0.1179	0.0064
j	3.0562	0.0967	0.0967	0.0965	0.0877	**1.20**	0.0965	0.0052
k	2.6923	0.1213	0.1214	0.1209	**0.1102**	2.82	0.1209	0.0065
l	2.9806	0.1012	0.1013	0.1008	0.0917	2.82	0.1008	0.0055

Table 24. Evaluation of Canny edge maps using Table 4 measures.

Figure 8	MSE	RMSE	PSNR	MAE	SSIM
b	0.0064	0.0800	70.0683	0.0064	0.9741
c	0.0061	0.0784	70.2426	0.0061	0.9754
d	0.0054	0.0732	70.8456	0.0054	0.9798
e	0.0063	0.0796	70.1073	0.0063	0.9744
f	0.0062	0.0786	70.2211	0.0062	0.9752
g	**0.0051**	**0.0711**	**71.0875**	**0.0051**	**0.9813**
h	0.0073	0.0854	69.4991	0.0073	0.9693
i	0.0064	0.0799	70.0838	0.0064	0.9741
j	0.0052	0.0722	70.9553	0.0052	0.9802
k	0.0065	0.0809	69.9710	0.0065	0.9731
l	0.0055	0.0739	70.756	0.0055	0.9790

Table 25. Evaluation of Canny edge maps using FoMs given in Table 6.

Figure 8	FOM_{pr}	FOM_{pin}	D_p
b	0.8821	0.9652	0.9824
c	0.8871	0.9667	0.9831
d	0.9013	0.9743	0.9870
e	0.8829	0.9599	0.9798
f	0.8864	0.9738	0.9866
g	**0.9068**	0.9822	0.9909
h	0.8653	0.9562	0.9780
i	0.8824	0.9570	0.9783
j	0.9037	**0.9888**	**0.9943**
k	0.8796	0.9666	0.9831
l	0.8997	0.9747	0.9871

Table 26. Evaluation of Canny edge maps using Table 6 measures.

Figure 8	Y	H	D^k	$f_2 d_6$	S^k	Ψ
b	0.0196	63.0079	0.4810	0.6805	0.6807	0.0043
c	0.0225	31.2570	0.6378	0.5321	0.5323	0.0029
d	0.0157	31.2570	0.3138	0.3829	0.3831	0.0021
e	0.0160	31.2570	0.3204	0.6402	0.6403	0.0033
f	0.0211	63.1269	0.5585	0.5089	0.5092	0.0037
g	0.0174	31.2570	0.3899	0.2387	0.2390	0.0014
h	0.0163	63.1269	0.3269	0.8346	0.8346	0.0053
i	0.0179	63.1269	0.4007	**1.1031**	**1.1031**	0.0063
j	**0.0136**	**2.2361**	**0.2348**	0.1051	0.1053	**0.0003**
k	0.0213	31.2570	0.5628	0.5109	0.5111	0.0030
l	0.0208	31.4006	0.5502	0.3836	0.3839	0.0021

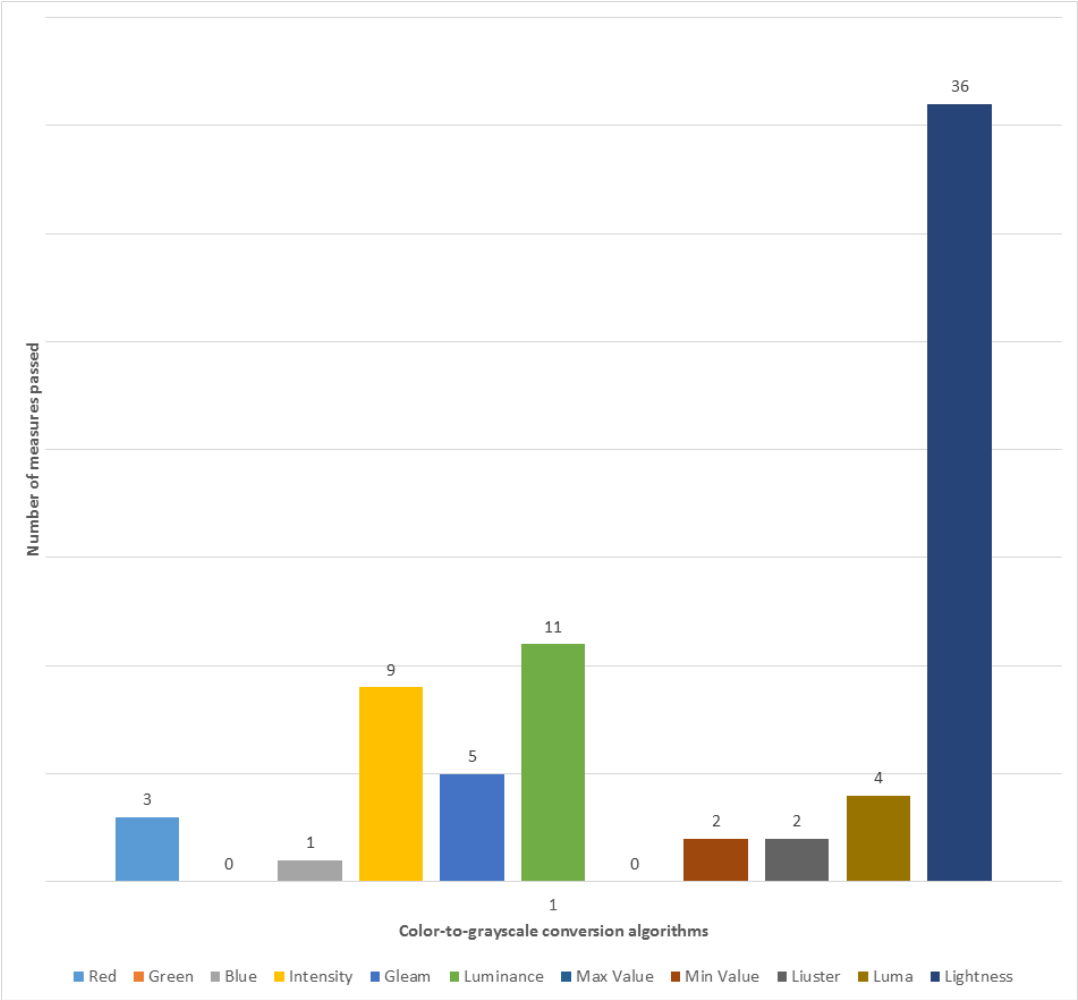
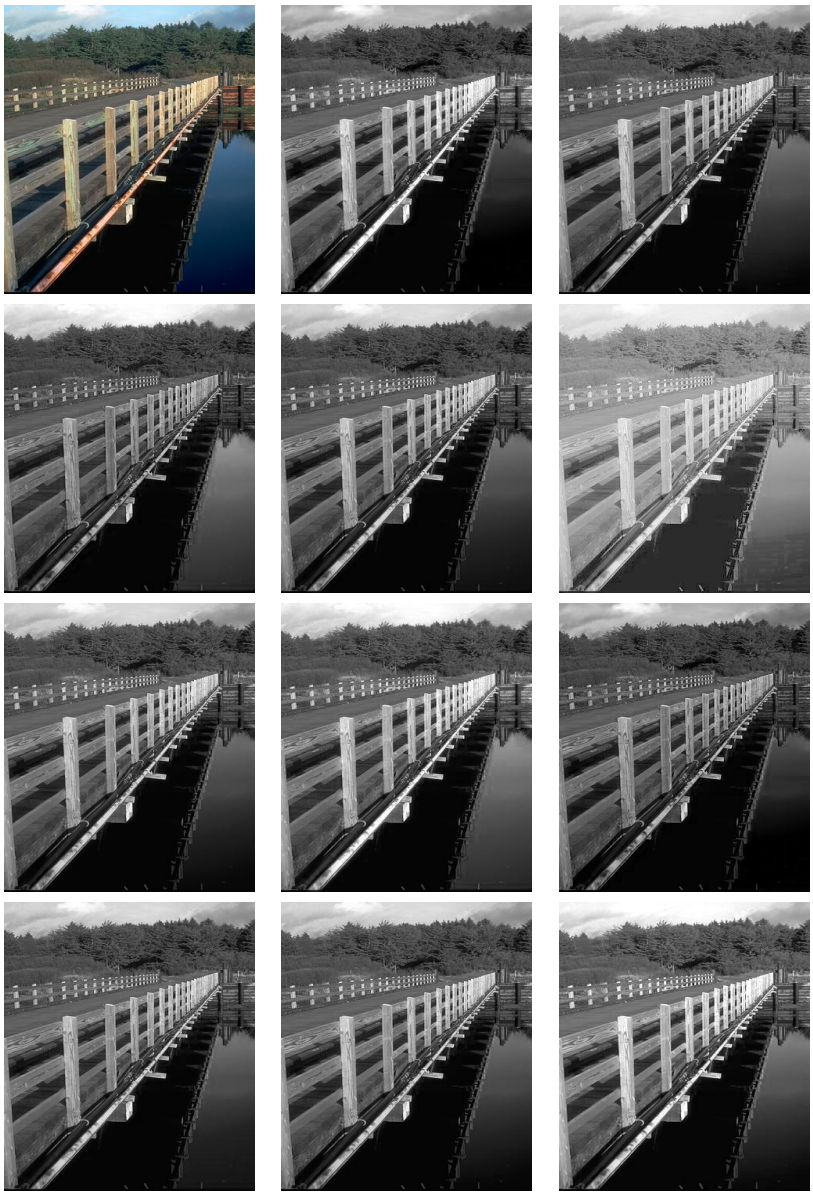


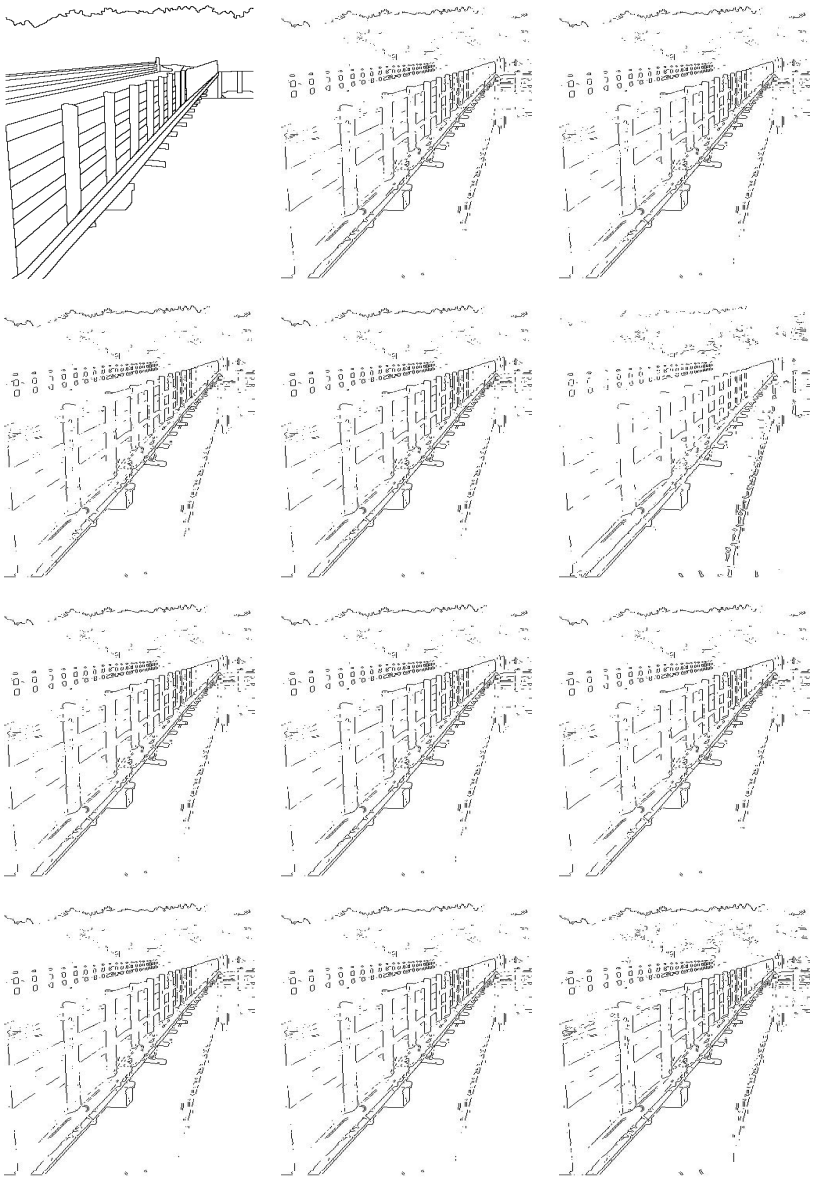
Figure 9. Performance measure of color-to-grayscale algorithms for synthetic image

Each edge map figure is followed by four performance measurement tables. The first two tables are the statistical measures while the other two are the figure of merits based edge measures.



a	b	c
d	e	f
g	h	i
j	k	l

Figure 10. Grayscale representations of natural image (a) Original color image. (b)-(l) Grayscale images obtained from methods listed in Table 1.



a	b	c
d	e	f
g	h	i
j	k	l

Figure 11. Resultant Sobel edge maps (a) Ground Truth. (b)-(l) Edge maps obtained using Sobel edge detection on grayscale images in Figure 10. (b)-(l).

Table 27. Evaluation of Sobel edge maps using Table 5 measures.

Figure 11	BSNR	PM	SSR	Φ	F_α	OVER	UNDER	LOC
b	0.7002	0.9071	0.9701	0.861	0.8521	0.0324	0.8567	0.0788
c	0.6956	0.9068	0.9698	0.8618	0.8528	0.0315	0.8576	0.078
d	0.6882	0.9083	0.9704	0.8652	0.8563	0.0305	0.8613	**0.0773**
e	0.6938	0.9077	0.9702	0.8632	0.8543	0.0314	0.8591	0.0779
f	0.6567	0.9286	0.981	**0.8964**	**0.8891**	**0.0293**	0.8936	0.0779
g	0.696	0.907	0.9699	0.8619	0.8529	0.0316	0.8578	0.0781
h	0.7010	0.9066	**0.9698**	0.8601	0.8511	0.0324	0.8558	0.0787
i	0.6874	0.9092	0.971	0.8666	0.8577	0.0306	0.8627	0.0774
j	0.6939	0.9077	0.9703	0.8632	0.8543	0.0314	0.8591	0.078
k	0.6960	0.9073	0.9701	0.8622	0.8533	0.0317	0.8581	0.0782
l	**0.7419**	**0.9055**	0.9700	0.8474	0.8389	0.0404	**0.8415**	0.0855

Table 28. Evaluation of Sobel edge maps using Table 4 measures.

Figure 11	MSE	RMSE	PSNR	MAE	SSIM
b	0.0788	0.2806	59.1705	0.0788	0.5723
c	0.078	0.2792	59.2129	0.078	0.572
d	**0.0773**	**0.2779**	**59.2539**	**0.0773**	**0.5762**
e	0.0779	0.2791	59.2165	0.0779	0.5739
f	0.0779	0.279	59.219	0.0779	0.5466
g	0.0781	0.2794	59.2068	0.0781	0.5731
h	0.0787	0.2806	59.1716	0.0787	0.5716
i	0.0774	0.2782	59.2462	0.0774	0.5757
j	0.078	0.2792	59.2151	0.078	0.5746
k	0.0782	0.2796	59.2032	0.0782	0.5725
l	0.0855	0.2924	58.8142	0.0855	0.5346

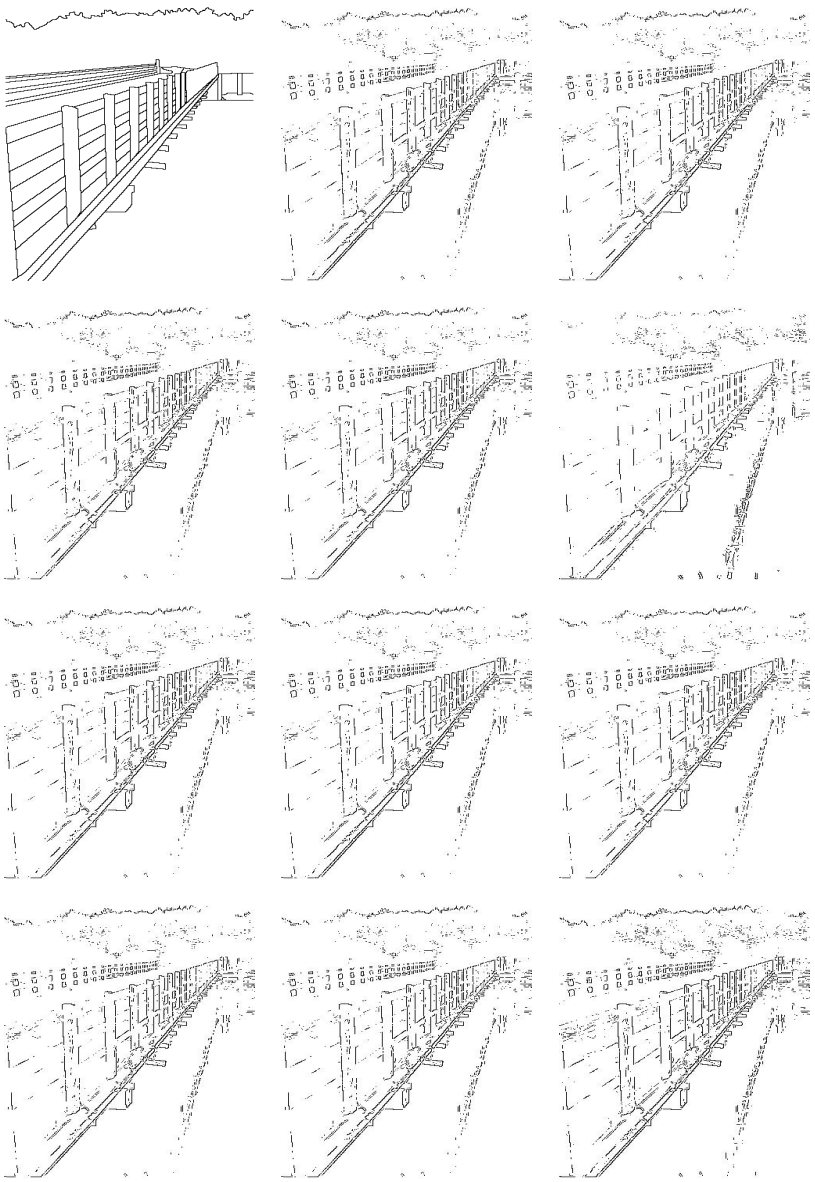
Table 29. Evaluation of Sobel edge maps using FoMs given

in Table 6.

Figure 11	FOM_{pr}	FOM_{pin}	D_p
b	0.4689	0.9897	0.7139
c	0.4596	0.9688	0.7118
d	0.4469	0.9409	0.7099
e	0.4571	0.9643	0.7109
f	0.3423	0.8184	0.6783
g	0.4609	0.9713	0.7119
h	0.47	0.996	0.7161
i	0.4453	0.9366	0.7081
j	0.4573	0.9652	0.7111
k	0.4606	0.9708	0.712
l	**0.528**	**1.136**	**0.726**

Table 30. Evaluation of Sobel edge maps using Table 6 measures.

Figure 11	Y	H	D^k	$f_2 d_6$	S^k	Ψ
b	**19.3099**	106.527	**5.6828**	**5.6828**	**4.3935**	0.1975
c	20.3719	112.0045	5.8706	5.8706	4.4858	0.201
d	20.4565	**103.0777**	5.7839	5.7839	4.5352	0.196
e	19.8389	106.527	5.7038	5.7038	4.4237	0.1955
f	42.8325	161.1987	14.4956	14.4956	8.5482	0.3664
g	19.8704	111.2116	5.7458	5.7458	4.4195	0.1971
h	20.1126	112.0045	5.8611	5.8611	4.4414	0.2046
i	20.0797	**103.0777**	5.7385	5.7385	4.5302	**0.1933**
j	19.9076	107.2987	5.7182	5.7182	4.4411	0.1965
k	20.1275	112.0045	5.8138	5.8138	4.4595	0.1998
l	20.9746	113.6002	7.4039	7.4039	5.1565	0.2738



a	b	c
d	e	f
g	h	i
j	k	l

Figure 12. Resultant Roberts edge maps
(a) Ground Truth. (b)-(l) Edge maps obtained using Roberts edge detection on grayscale images in Figure 10. (b)-(l).

Table 31. Evaluation of Roberts edge maps using Table 5 measures.

Figure 12	BSNR	PM	SSR	Φ	F_α	OVER	UNDER	LOC
b	0.7603	0.86	0.9387	0.7889	0.7772	0.0331	0.7822	0.0753
c	0.7516	0.8636	0.9412	0.796	0.7844	0.0323	0.7897	0.0749
d	0.7444	0.867	0.9436	0.8022	0.7906	0.0318	0.7962	**0.0748**
e	0.7514	0.863	0.9407	0.7954	0.7837	0.0322	0.7891	**0.0748**
f	0.7049	0.8902	0.9592	**0.8401**	**0.8298**	**0.0297**	0.8355	0.0751
g	0.7536	0.862	0.94	0.7935	0.7818	0.0323	0.7872	**0.0748**
h	0.761	**0.8594**	**0.9382**	0.788	0.7762	0.0331	0.7813	0.0752
i	0.7438	0.8675	0.944	0.8029	0.7914	0.0318	0.797	0.0748
j	0.7509	0.864	0.9415	0.7967	0.785	0.0323	0.7904	0.0749
k	0.7524	0.8626	0.9404	0.7946	0.7829	0.0323	0.7883	**0.0748**
l	**0.8049**	0.8602	0.9398	0.7694	0.759	0.0427	**0.7600**	0.0831

Table 32. Evaluation of Roberts edge maps using Table 4 measures.

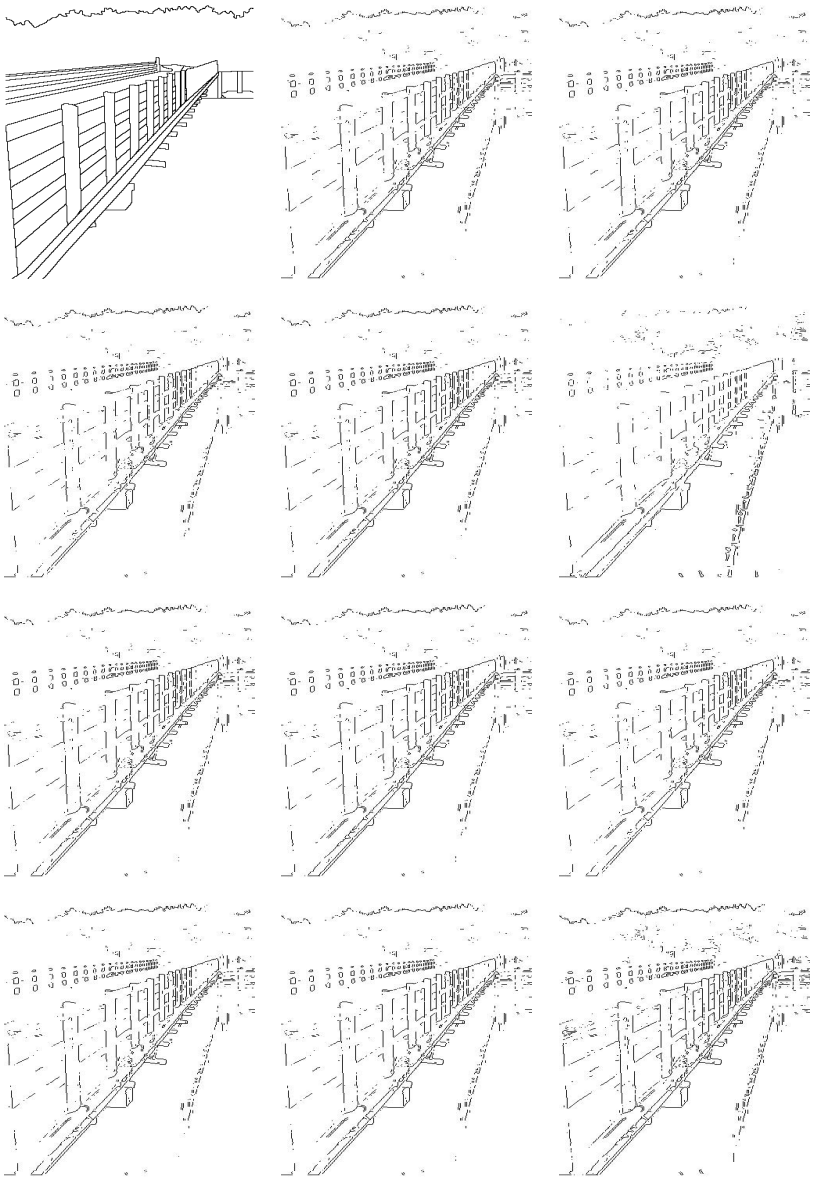
Figure 12	MSE	RMSE	PSNR	MAE	SSIM
b	0.0753	0.2743	59.3664	0.0753	0.5697
c	0.0749	0.2737	59.3866	0.0749	0.5702
d	**0.0748**	**0.2734**	59.3964	**0.0748**	0.5697
e	**0.0748**	**0.2734**	**59.3976**	**0.0748**	0.5707
f	0.0751	0.2739	59.3791	0.0751	0.5495
g	**0.0748**	**0.2734**	59.3949	**0.0748**	0.5707
h	0.0752	0.2742	59.3702	0.0752	**0.5718**
i	**0.0748**	0.2735	59.3934	0.0748	0.5678
j	0.0749	0.2737	59.3863	0.0749	0.5697
k	**0.0748**	**0.2734**	59.3953	**0.0748**	0.5711
l	0.0831	0.2882	58.9378	0.0831	0.5316

Table 33. Evaluation of Roberts edge maps using FoMs given in Table 6.

Figure 12	FOM_{pr}	FOM_{pin}	D_p
b	0.5207	1.0471	0.752
c	0.5059	1.0231	0.7475
d	0.4955	1.0116	0.746
e	0.505	1.0234	0.748
f	0.3751	0.8738	0.7136
g	0.508	1.0254	0.7487
h	0.5216	1.0489	0.7522
i	0.4943	1.0093	0.747
j	0.5044	1.0255	0.7476
k	0.5066	1.0233	0.7484
l	**0.5957**	**1.2408**	**0.7725**

Table 34. Evaluation of Roberts edge maps using Table 6 measures.

Figure 12	Y	H	D^k	$f_2 d_6$	S^k	Ψ
b	19.1243	**122.2989**	6.2508	6.2508	4.8691	0.2087
c	19.6985	125.2997	6.3794	6.3794	4.9655	0.2088
d	19.7581	**122.2989**	6.3623	6.3623	4.9201	**0.2048**
e	19.6716	**122.2989**	6.3764	6.3764	4.9323	0.207
f	41.613	160.4307	15.2336	15.2336	9.3719	0.3794
g	19.5064	**122.2989**	6.3336	6.3336	4.9368	0.207
h	**18.9650**	**122.2989**	**6.2205**	**6.2205**	**4.8416**	0.2069
i	19.9674	**122.2989**	6.4118	6.4118	4.9627	0.207
j	19.5222	**122.2989**	6.3541	6.3541	4.9077	0.2062
k	19.5857	**122.2989**	6.3452	6.3452	4.9385	0.2071
l	20.3054	146	7.9977	7.9977	5.7066	0.2925



a	b	c
d	e	f
g	h	i
j	k	l

Figure 13. Resultant Prewitt edge maps (a) Ground Truth. (b)-(l) Edge maps obtained using Prewitt edge detection on grayscale images in Figure 10. (b)-(l).

Table 35. Evaluation of Prewitt edge maps using Table 5 measures.

Figure 13	BSNR	PM	SSR	Φ	F_α	OVER	UNDER	LOC
b	0.6892	0.9076	0.9701	0.8642	0.8552	0.0306	0.8603	0.0772
c	0.6822	0.9082	0.9702	0.8666	0.8576	0.0295	0.8628	0.0764
d	0.6736	0.9105	0.9713	0.8711	0.8623	0.0286	0.8677	**0.0758**
e	0.6826	0.9073	**0.9697**	0.8654	0.8564	0.0294	0.8617	0.0762
f	0.6407	0.9315	0.9821	**0.9026**	**0.8955**	**0.0273**	0.9001	0.0764
g	0.6836	0.908	0.9702	0.866	0.857	0.0297	0.8622	0.0765
h	0.6899	0.9071	0.9698	0.8634	0.8544	0.0306	0.8595	0.0772
i	0.6729	0.911	0.9716	0.8719	0.8631	0.0285	0.8685	**0.0758**
j	0.6819	0.908	0.9701	0.8664	0.8574	0.0294	0.8627	0.0763
k	0.6830	0.9081	0.9702	0.8662	0.8572	0.0296	0.8625	0.0765
l	**0.7320**	**0.9066**	0.9704	0.8517	0.8431	0.0385	**0.8462**	0.084

Table 36. Evaluation of Prewitt edge maps using Table 4 measures.

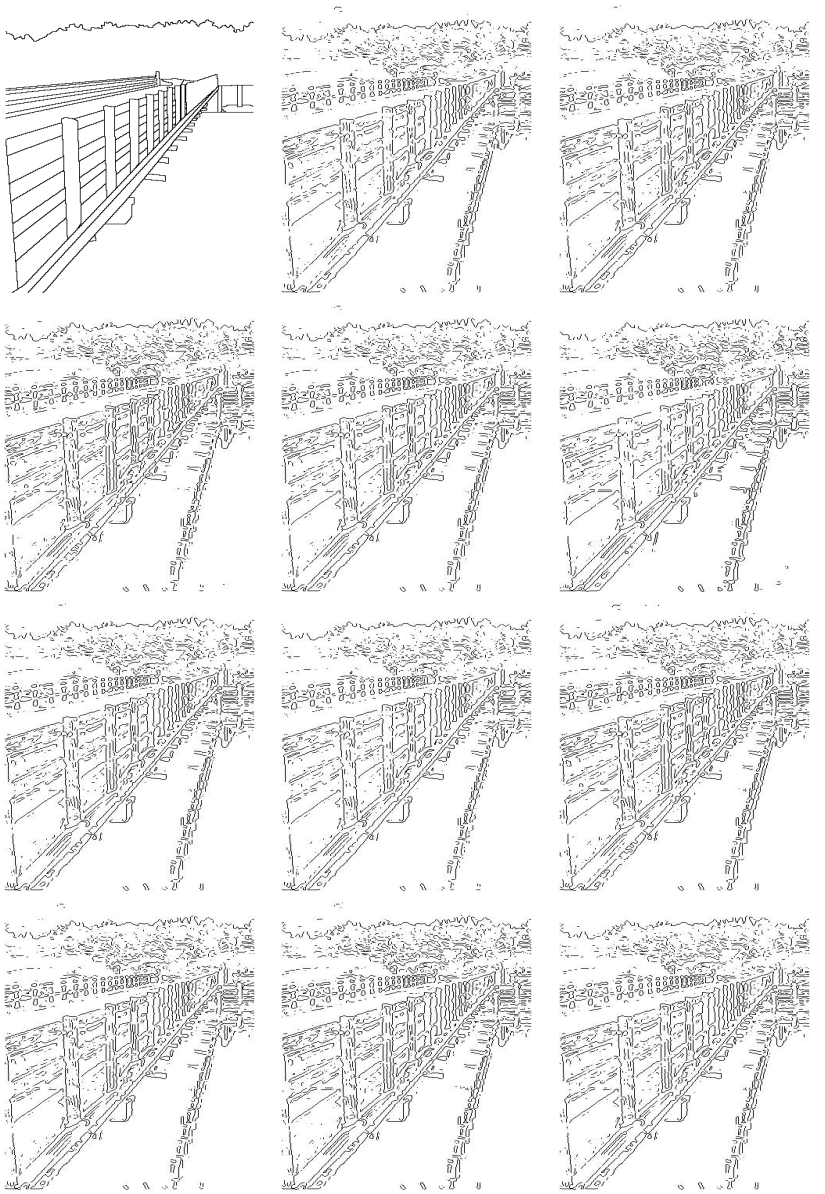
Figure 13	MSE	RMSE	PSNR	MAE	SSIM
b	0.0772	0.2779	59.2557	0.0772	0.5837
c	0.0764	0.2764	59.303	0.0764	0.5832
d	**0.0758**	**0.2752**	**59.3396**	**0.0758**	**0.5876**
e	0.0762	0.276	59.3129	0.0762	0.5858
f	0.0764	0.2764	59.3026	0.0764	0.5525
g	0.0765	0.2766	59.2945	0.0765	0.5848
h	0.0772	0.2778	59.2564	0.0772	0.5827
i	**0.0758**	**0.2752**	59.3381	**0.0758**	0.5869
j	0.0763	0.2762	59.3081	0.0763	0.5849
k	0.0765	0.2765	59.2985	0.0765	0.5846
l	0.084	0.2897	58.893	0.084	0.5443

Table 37. Evaluation of Prewitt edge maps using FoMs given in Table 6.

Figure 13	FOM_{pr}	FOM_{pin}	D_p
b	0.452	0.9515	0.7089
c	0.4399	0.9272	0.7069
d	0.4259	0.896	0.7039
e	0.4401	0.9249	0.7074
f	0.3196	0.7758	0.668
g	0.442	0.9301	0.7068
h	0.4533	0.9534	0.7104
i	0.4245	0.895	0.7032
j	0.4393	0.9269	0.7067
k	0.4408	0.9288	0.7066
l	**0.5122**	**1.0972**	**0.7195**

Table 38. Evaluation of Prewitt edge maps using Table 6 measures.

Figure 13	Y	H	D^k	$f_2 d_6$	S^k	Ψ
b	**19.1833**	106.527	**5.3744**	**5.3744**	**4.3117**	0.1852
c	20.5803	112.8008	5.6176	5.6176	4.4467	0.1899
d	20.7056	**102.3035**	5.5757	5.5757	4.5252	0.1855
e	19.6992	106.527	5.3869	5.3869	4.3594	0.1824
f	45.2437	161.1987	14.8633	14.8633	8.6126	0.3556
g	19.8549	111.2116	5.4656	5.4656	4.3826	0.1854
h	20.133	112.0045	5.5891	5.5891	4.4041	0.1934
i	20.2218	**102.3035**	5.4802	5.4802	4.478	**0.1815**
j	19.746	106.527	5.4058	5.4058	4.3533	0.1828
k	20.1239	112.0045	5.5242	5.5242	4.4052	0.1869
l	21.4265	113.6002	7.2613	7.2613	5.0858	0.2634



a	b	c
d	e	f
g	h	i
j	k	l

Figure 14. Resultant LoG edge maps (a) Ground Truth. (b)-(l) Edge maps obtained using Laplacian of Gaussian edge detection on grayscale images in Figure 10. (b)-(l).

Table 39. Evaluation of LoG edge maps using Table 5 measures.

Figure 14	BSNR	PM	SSR	Φ	F_α	OVER	UNDER	LOC
b	0.8148	0.9194	0.9774	0.8389	0.8336	0.0672	0.8286	0.11
c	0.8241	0.9225	0.9788	0.8393	0.8346	0.0726	0.8282	0.1151
d	0.8354	0.9245	0.9795	0.8369	0.8329	0.079	0.8246	0.1209
e	0.8266	0.9215	0.9782	0.8365	0.8318	0.0732	0.8251	0.1155
f	0.8274	0.9259	0.9804	**0.8431**	**0.8388**	0.0758	0.8317	0.1184
g	0.8241	0.9212	0.9781	0.8372	0.8325	0.0719	0.8261	0.1144
h	0.8129	**0.9194**	**0.9774**	0.8398	0.8344	**0.0664**	0.8297	**0.1093**
i	0.8377	0.9244	0.9794	0.8354	0.8316	0.08	0.8229	0.1218
j	0.8269	0.9212	0.9781	0.8359	0.8313	0.0732	0.8245	0.1154
k	**0.8399**	0.9239	0.9791	0.8335	0.8298	0.0808	**0.8207**	0.1225
l	0.8252	0.9211	0.978	0.8366	0.8318	0.0723	0.8253	0.1147

Table 40. Evaluation of LoG edge maps using Table 4 measures.

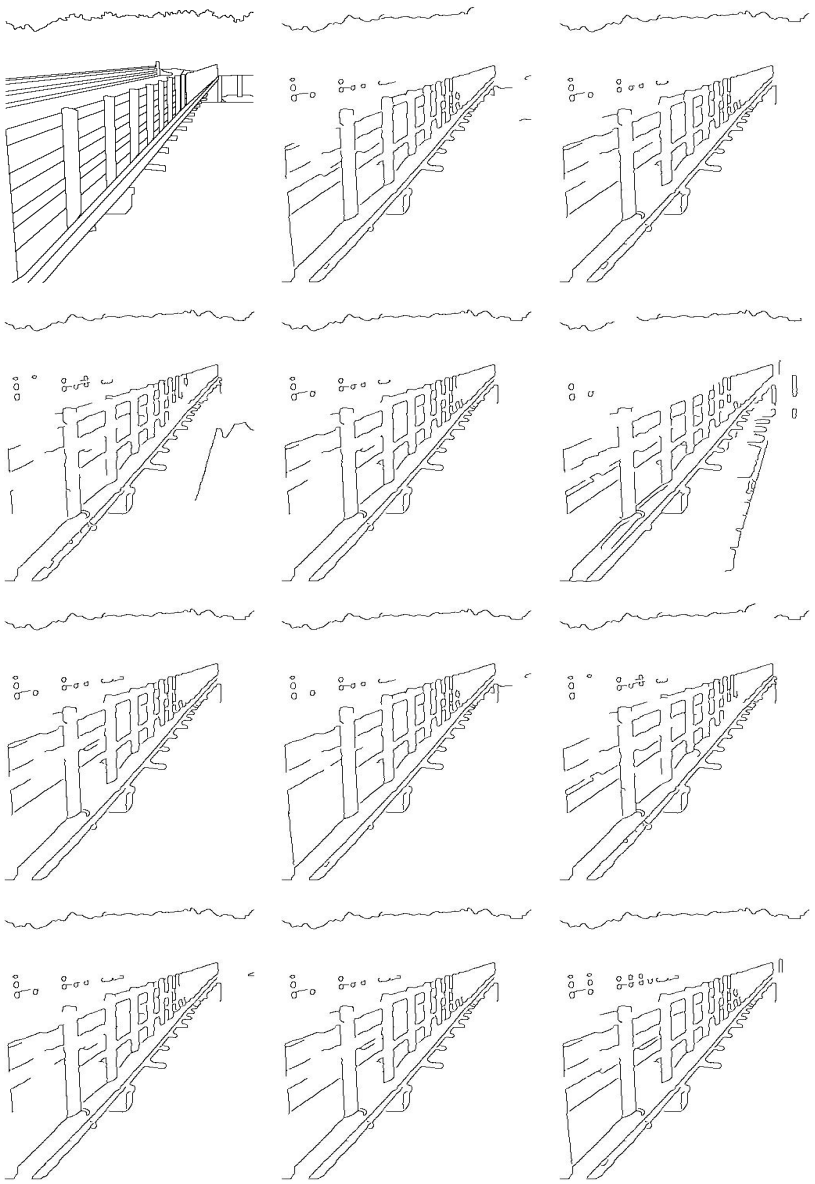
Figure 14	MSE	RMSE	PSNR	MAE	SSIM
b	0.11	0.3317	57.718	0.11	0.4052
c	0.1151	0.3392	57.5227	0.1151	0.394
d	0.1209	0.3477	57.308	0.1209	0.378
e	0.1155	0.3398	57.5071	0.1155	0.3935
f	0.1184	0.344	57.4009	0.1184	0.3809
g	0.1144	0.3382	57.5499	0.1144	0.3968
h	**0.1093**	**0.3306**	**57.7452**	**0.1093**	**0.4122**
i	0.1218	0.3489	57.2769	0.1218	0.3744
j	0.1154	0.3397	57.5093	0.1154	0.3936
k	0.1225	0.3499	57.2525	0.1225	0.3704
l	0.1147	0.3387	57.5369	0.1147	0.395

Table 41. Evaluation of LoG edge maps using FoMs given in Table 6.

Figure 14	FOM_{pr}	FOM_{pin}	D_p
b	0.4884	1.5902	0.7418
c	0.4802	1.683	0.7405
d	0.4772	1.8007	0.7453
e	0.48	1.6938	0.7428
f	0.4473	1.7059	0.7353
g	0.4821	1.673	0.7422
h	0.4977	1.5927	0.7408
i	0.469	1.8049	0.7472
j	0.4821	1.6954	0.7433
k	0.4586	1.8229	0.7445
l	0.4807	1.6801	0.7426

Table 42. Evaluation of LoG edge maps using Table 6 measures.

Figure 14	Y	H	D^k	$f_2 d_6$	S^k	Ψ
b	23.6406	16.6434	12.6281	1.7534	0.8247	0.0553
c	22.3377	16.9706	12.3364	2.0725	2.0494	0.0815
d	20.2098	13.6015	11.6473	1.6918	1.7495	0.0727
e	22.0553	16.9706	12.2558	2.0724	2.0494	0.082
f	24.6368	16.9706	14.3443	1.7112	1.4987	0.0675
g	22.5826	16.9706	12.407	2.0642	2.0492	0.081
h	22.1644	14.8661	11.6687	2.0218	2.0225	0.0747
i	21.3488	16.6434	12.4414	1.6975	1.7785	0.0767
j	21.8819	16.9706	12.1514	2.0684	2.0464	0.082
k	21.4256	14.8661	12.7632	1.5347	1.4164	0.0626
l	22.5368	16.6434	12.414	2.0681	2.049	0.0812



a	b	c
d	e	f
g	h	i
j	k	l

Figure 15. Resultant Canny edge maps (a) Ground Truth. (b)-(l) Edge maps obtained using Canny edge detection on grayscale images in Figure 10. (b)-(l).

Table 43. Evaluation of Canny edge maps using Table 5 measures.

Figure 15	BSNR	PM	SSR	Φ	F_α	OVER	UNDER	LOC
b	0.6612	0.9012	0.9649	0.8631	0.8534	0.025	0.8598	0.072
c	0.6556	0.905	0.9672	0.8686	0.8591	0.0249	0.8655	0.0722
d	0.6495	0.9117	0.9712	0.8775	0.8686	0.0251	0.8746	0.0729
e	0.6518	0.9054	0.9672	0.8697	0.8603	**0.0244**	0.8667	**0.0718**
f	0.6652	0.927	0.9804	**0.893**	**0.8855**	0.0304	0.8899	0.0787
g	0.6558	0.9047	0.9669	0.8681	0.8587	0.0248	0.865	0.0721
h	0.6596	0.9013	**0.9648**	0.8635	0.8538	0.0248	0.8603	**0.0718**
i	0.652	0.9102	0.9704	0.8752	0.8662	0.0252	0.8723	0.0729
j	0.6543	0.9044	0.9667	0.8681	0.8586	0.0246	0.865	0.0719
k	0.6572	0.9041	0.9666	0.8672	0.8577	0.0249	0.8641	0.0721
l	**0.6764**	**0.9009**	0.9654	0.8594	0.8498	0.0272	**0.8558**	0.0738

Table 44. Evaluation of Canny edge maps using Table 4 measures.

Figure 15	MSE	RMSE	PSNR	MAE	SSIM
b	0.072	0.2682	59.563	0.072	0.643
c	0.0722	0.2686	59.5512	0.0722	0.6427
d	0.0729	0.27	59.5047	0.0729	0.6294
e	**0.0718**	**0.2679**	**59.5743**	**0.0718**	0.6428
f	0.0787	0.2805	59.1734	0.0787	0.5963
g	0.0721	0.2685	59.5551	0.0721	0.6429
h	**0.0718**	**0.2679**	**59.5739**	**0.0718**	**0.6455**
i	0.0729	0.2699	59.5074	0.0729	0.637
j	0.0719	0.268	59.5696	0.0719	0.6429
k	0.0721	0.2685	59.5524	0.0721	0.6428
l	0.0738	0.2717	59.4521	0.0738	0.6403

Table 45. Evaluation of Canny edge maps using FoMs given in Table 6.

Figure 15	FOM_{pr}	FOM_{pin}	D_p
b	0.4191	0.736	0.6768
c	0.4188	0.7382	0.6757
d	0.3907	0.7004	0.6704
e	0.4082	0.7202	0.6749
f	0.3875	0.7211	0.648
g	0.4173	0.7382	0.6763
h	0.4225	0.7462	0.6814
i	0.4029	0.7192	0.6711
j	0.4112	0.7274	0.677
k	0.4197	0.7412	0.677
l	**0.4500**	**0.8100**	**0.6869**

Table 46. Evaluation of Canny edge maps using Table 6 measures.

Figure 15	Y	H	D^k	$f_2 d_6$	S^k	Ψ
b	4.6874	72.5604	1.88	5.9988	4.9695	0.1485
c	**3.7605**	44.1022	**1.7028**	5.7188	4.7002	0.1307
d	15.4101	62.8013	3.9297	6.3354	5.9697	0.1789
e	3.9148	44.1022	1.7580	6.0085	4.9428	0.1363
f	26.3854	115.3777	7.5780	9.1298	9.1603	0.335
g	3.8133	44.1022	1.7219	5.5455	4.5872	0.1259
h	3.7717	**35.0000**	1.7031	5.4156	4.4954	0.1229
i	4.2518	41.0000	1.9337	5.7909	4.8189	0.1333
j	3.9004	39.0000	1.7622	5.639	4.6776	0.1268
k	3.8052	44.1022	1.7199	5.5305	4.5735	0.1259
l	3.8939	41.0000	1.8133	**4.6757**	**3.9596**	**0.1128**

V. Conclusion

Edge detection is a widely studied area in image processing. Edge detectors have been evaluated on their own and in the presence of noise for better performance. Also, analysis of edge detection evaluators have been done. However, assessment of edge detectors with color-to-grayscale conversion algorithms have not been considered. In this work, I have evaluated the effect of various color-to-grayscale conversion algorithms on the performance of edge detection process. The experimental results show a performance gap between the edge detectors, when a suitable grayscale representation is considered. The first order derivative edge detectors (Sobel, Roberts and Prewitt) performed better on Lightness grayscale representation. While, edge detectors (Laplacian of Gaussian (LoG) and Canny) that incorporate de-noising step, result a better edge map on Gleam grayscale representation. In both cases the gamma corrected grayscale representations performed better than other methods. The results indicate that edge detection is not only dependent on the detector used but also on the color-to-grayscale conversion algorithms.

VI. REFERENCES

1. M. Petrou and C. Petrou, Image processing: The fundamentals, 2nd Edition, John Wiley & Sons, 2010

2. M. Sonka, V. Hlavac, and R. Boyle, Image processing, analysis and machine vision, 2nd Edition

3. R. Jain, R. Kasturi, and B. Schunck, Machine vision, 2nd Edition, McGraw-Hill Book, 1995

4. A.-V. Diaconu, and I. Sima, "Simple, XOR based, image edge detection. Study on black and white / grayscale synthetic images", Int. Conf. on Electronics, Computers and Artificial Intelligence, ECAI'13 IEEE Int. Conf., Piteşti, Romania, 27-29 June, 2013.

5. A.-V. Diaconu V. Ionescu "Simple XOR based image edge detection. study on Natural Grayscale Images" Int. Conf. on Electronics, Computers and Artificial Intelligence, ECAI'13 IEEE Int. Conf., Piteşti, Romania, 27÷29 June, 2013.

6. A. Koschan, M. Abidi, "Detection and classification of edges in color images", IEEE Signal Processing Mag., vol. 22, no. 1, pp. 64-73, Jan. 2005.

7. S.Y. Zhu, K.N. Plataniotis, A.N. Venetsanopoulos Comprehensive analysis of edge detection in color image processing Opt. Eng., 38 (1999), p. 612

8. G. Woods, Dgital Image Processing, 3rd Edition, Pearson Prentice Hall 2010.

9. L. Zhuo, X. Hu, L. Jiang, J. Zhang A color image edge detection algorithm

based on color difference Sensing Imaging, 17 (1) (2016) 1-13

10. S. Xie and Z. Tu. Holistically-nested edge detection. In ICCV, 2015.

11. GrundlandM., Dodgson N. Decolorize: fast, contrast enhancing, color to grayscale conversion. Cambridge, 2009.

12. S. Wesolkowkil M. E. Jernigan R. D. Dony "Comparison of color image edge detectors in multiple color spaces" Proc. IEEE ICIP 2000 pp. 796-799 2000

13. Y. Yitzhaky, E. Peli A method for objective edge detection evaluation and detector parameter selection IEEE Transactions on Pattern Analysis and Machine Intelligence, 25 (8) (2003), pp. 1027-1033

14. k. Panetta, C. Gao, S. Agaian, S. Nercessian, S.: A new reference-based edge map quality measure. IEEE Trans. Syst. Man Cybern. Syst. 46(11), 1505-1517 (2016)

15. S. C. Nercessian, S. S. Agaian, K. A. Panetta, "A new reference-based measure for objective edge map evaluation", Proc. SPIE Def. Security Sens. Conf., 2009.

16. S. Nercessian, K. Panetta, S. Agaian, "A non-reference measure for objective edge map evaluation", Proc. IEEE Int. Conf. Syst. Man Cybern., pp. 4563-4568, 2009.

17. k. Panetta, C. Gao, S. Agaian, S. Nercessian, "Non-reference medical image edge map measure. J Bio Imaging 2014:1-12

18. B. Magnier, Edge detection: a review of dissimilarity evaluations and a proposed normalized measure, Multimedia tools and applications, PP. 1-45, Sep 2017.

19. Bovik, A. (ed.): The Essential Guide to Image Processing. Academic Press, 2009.

20. Shah, M.: Fundamentals of Computer Vision. Chapter 2 (1992)

21. W. K. Pratt, Digital Image Processing, vol. 242. New York, NY, USA: Wiley, 1978

22. A. J. Pinho, L. B. Almeida, "Edge detection filters based on artificial neural networks" in Image Analysis and Processing, Berlin, Germany:Springer, pp. 159-164, 1995.

23. W.A. Yasnoff, W. Galbraith and J.W. Bacus, Error measures for objective assessment of scene segmentation algorithms, Analytical and Quantitative Cytology: Vol. 1(2), pp. 107-121, 1978.

24. D. P. Huttenlocher and W. J. Rucklidge, A multi-resolution technique for comparing images using the Hausdorff distance, IEEE Trans. on Pattern Analysis and Machine Intel.: Vol. 15(9), pp. 850-863, 1993.

25. C. Lopez-Molina, B. De Baets and H. Bustince, Quantitative error measures for edge detection, Pattern Recognition: Vol. 46(4), pp. 1125-1139, 2013.

26. M.P. Dubuisson, A.K. Jain, A modified Hausdorff distance for object

matching, 12th IAPR Int. Conf. on Pattern Recognition: Vol. 1, pp.566-568, 1994.

27. C. Grigorescu, N. Petkov and M.A. Westenberg, Contour detection based on nonclassical receptive field inhibition, IEEE Trans. on Image Proc.: Vol. 12(7), pp. 729-739, 2003.

28. R. Usamentiaga, D. F. Garc´ıa, C. Lopez, and D. Gonz´alez, ´ A method for assessment of segmentation success considering uncertainty in the edge positions, EURASIP J. on Applied Signal Proc., Vol. 2006, pp.207-207, 2006.

29. S. Venkatesh and P.L. Rosin, Dynamic threshold determination by local and global edge evaluation, Comp. Vision, Graphics, and Image Proc.:Vol. 57(2), pp. 146-160, 1995.

30. D. R. Martin, C. C. Fowlkes and J. Malik, Learning to detect natural image boundaries using local brightness, color, and texture cues, IEEE Trans. on Pattern Analysis and Machine Intel.: Vol. 26(5), pp. 530-549, 2004

31. A. J. Baddeley, An error metric for binary images, Robust Comp. Vision: Quality of Vision Algorithms: pp. 59-78, 1992

32. S. Chabrier, H. Laurent, C. Rosenberger, and B. Emile, Comparative study of contour detection evaluation criteria based on dissimilarity measures, EURASIP J. on Image and Video Proc.: Vol. 2008, pp. 1-13, 2008.

33. [SSIM] Zhou, W., 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, Issue 4, April 2004, pp. 600-612.

34. (2011) Mean Squared Error. In: Sammut C., Webb G.I. (eds) Encyclopedia of Machine Learning. Springer, Boston, MA

35. (2017) Root-Mean-Square Error. In: Shekhar S., Xiong H., Zhou X. (eds) Encyclopedia of GIS. Springer, Cham

36. (2011) Mean Absolute Error. In: Sammut C., Webb G.I. (eds) Encyclopedia of Machine Learning. Springer, Boston, MA

37. Wilson, D. L., Baddeley, A. J. and Owens, R. A. (1997) A new metric for grey-scale image comparison. Int. J.Comput. Vis., 24, 5-17

38. Kanan C, Cottrell GW (2012) Color-to-Grayscale: Does the Method Matter in Image Recognition? PLoS ONE 7(1): e29740

39. P. Arbelaez, M. Maire, C. Fowlkes and J. Malik, 'Contour Detection and Hierarchical Image Segmentation', IEEE TPAMI, Vol. 33, No. 5, pp. 898-916, May 2011.

VII. ACKNOWLEDGEMENTS

My sincerest gratitude to my advisor and supervisor, Prof. Seokjoo Shin for his support and encouragement. He has been an inspiration for me during my master studies. His comments and suggestions have improved my work, and helped me to make my findings understandable and presentable. I am also thankful to Prof Inkyu Moon for his support and giving me a chance to start master's degree in Korea. I am also thankful to Prof. Moonsoo Kang and Prof. Kwon Goorak for their suggestions on my thesis work.

I am very grateful to my family who supported me with unconditional love and understanding. I may not thanked them enough for their patience during my stay in Korea. I am very thankful to Kim Hwan for making my life easy in Korea. I am also thankful to my friend Inayat Ullah for his fruitful discussions, comments and reviews on my work. Finally, thank you to my three best friends who made difficult decisions easy for me. Lillian C. Mutalemwa for her jokes to cheer me up, her fruitful discussions on my work and in general; Chua Bih Lii for his encouragement and for giving me a new understanding to things; and Thuy Lam Giang for staying with me in my ups and downs, supporting me emotionally and always been there for me.