Improved Color Edge Detection by Fusion of Hue, PCA & Hybrid Canny

Gurpreet kaur[1] and Vijay kumar Banga[2]

***Abstract*—** Edge detection is the name for a set of mathematical methods which target at classifying points in a image at which the image intensity varies sharply or, has discontinuities. This paper has focused on detecting the edges for the Hue factor in Human visual system (HSV). Hue is usually represented by position significance, so the existing edge detection techniques has been incapable to correctly perceive edges of hue factor in HSV color plane. As a consequence, the most popular approaches of color edge detection typically neglected the role of hue factor, thus misplaced some edges triggered by hue variations. This research work has proposed a new color edge detection technique based on the mixture of hue factor and principal component analysis to resolve the problems with existing methods. The algorithm has been designed and implemented in MATLAB. The performance metrics has shown the efficiency of the proposed algorithm over the existing ones.

***Keywords*:-** Edge Detection, Canny Edge Detector, PCA

I. INTRODUCTION

Edge detection refers to the process of identifying and locating sharp discontinuities in an image. The discontinuities are abrupt changes in pixel intensity which characterize boundaries of objects in a scene. Edge detection is a terminology in image processing that refers to algorithms which aim at identifying edges in an image. It is encountered in the areas of feature selection and feature extraction in Computer Vision. An edge detector accepts a digital image as input and produces an edge map as output. The edge map of some detectors includes explicit information about the position and strength of the edges and their orientation.

Gurpreet kaur, Dept. of Electronics and communication engineering from Amritsar College of Engineering and Technology,University of PTU, Punjab 143001 (e-mail: gurpreetkaurhohangmail.com)
Dr.Vijay Kumar Banga Principal (Prof. ECE) Amritsar College of Engineering and Tecnology, Punjab 143001 (e-mail:vijaykumar.banga@gmail.com)

The four steps of edge detection

Filtering- Filter image to improve performance of the edge detector w.r.t noise.

Smoothing- suppress as much noise as possible, without destroying the true edges.

Enhancement- applies a filter to enhance the quality of the edges in the image (sharpening).

Detection- determine which edge pixels should be discarded as noise and which should be retained (usually, thresholding provides the criterion used for detection).

Localization- determine the exact location of an edge (*sub-pixel* resolution might be required for some applications, that is, estimate the location of an edge to better than the spacing between pixels). Edge thinning and linking are usually required in this step.

II. LITERATURE SURVEY

Wesolkowski et al. (2000) [1] has presented a comprehensive color edge comparison across a variety color spaces. The edge detectors studied are the Sobel operator, the modified Roberts operator, the vector gradient operator, and the 3x3 difference vector operator. Pratt’s Figure of Merit is used for quantitative evaluation. The best quantitative and qualitative results were obtained with the Sobel operator on the hlh2h3 space. The results in the hlh2h3 space were generally better than results obtained in any other color space. Dikbas et al. (2007) [2] has proposed a color edge preserving grayscale conversion algorithm that helps detect color edges using only the luminance component. The algorithm calculates an approximation to the first principal component to form a new set of luminance coefficients instead of using the conventional luminance coefficients. This method can be directly applied to all existing grayscale edge detectors for color edge detection. Processing only one channel instead of three channels results in lower computational complexity compared to other color edge detectors. To save computation time and to design a hardware implementable real-time algorithm, the proposed method avoids eigenvector decomposition by making use of power iteration. The conversion enables the edge detector to detect some edges of the grayscale image that are not detected using regular grayscale image. Chen et al. (2010) [3] to improve the efficiency and the performance of the color edge detection, a novel color edge detection algorithm has proposed. An improved Kuwahara filter is used to smooth the original image first. After edge detection with each channel independently in RGB color space, an adaptive threshold selection method is applied to predict the optimum threshold value and an edge thinning algorithm is used to extract accurate edge. Edge detection is anessential tool in image processing and computer vision. Compared with gray image, color image provides more edge information of objects. However, the current color edge detection algorithms acquired so much time to compute and they are hardly used in real-time system. Technion et al. (2010) [4] has proposed a new type of edges on surfaces, termed relief edges. Intuitively, the surface can be considered as an unknown smooth manifold, on top of which a local height image is placed. Relief edges are the edges of this local image. We show how to compute these edges from the local differential geometric surface properties, by fitting a local edge model to the surface. Relief edges manage to capture the 3D features. They have been utilized to draw edges on scanned objects for artifact illustration in archaeology. Edge detection on surfaces, on the other hand, has received much less attention. The most common edges on surfaces are ridges and valleys, used for processing range images in computer vision, as well as for non-photorealistic rendering in computer graphics. This paper has extended the definition of edges from images to surfaces, for which image edges are a special case. These edges, termed relief curves, use local intrinsic surface properties together with a rough approximation of the base surface to produce superior results. Dezert et al. (2011) [5] has presented an algorithm is based on the fusion of local edge detectors results expressed into basic belief assignments thanks to a flexible modeling, and the proportional conflict redistribution rule developed in DSmT framework. The purpose of DSmT is to overcome the limitations of DST mainly by proposing new underlying models for the frames of discernment in order to fit better with the nature of real problems, and proposing new efficient combination and conditioning rules. The basic belief assignment (bba) associated with the edge of a pixel in each channel of the image is defined according to its gradient magnitude, and one can easily model the uncertainty about our belief it belong or not to an edge. PCR5 and DS rules have been applied in this work to combine these bba's to get the global bba for final decision-making. Other rules of combination of bba's could also have been used instead but they are known to be less efficient than PCR5 or DS rules in high and low conflict cases respectively. XIAO et al. (2011) [6] has proposed a multi-scale edge detection algorithm which took soft threshold method to implement detail enhancement and noise reduction of the true color image. Firstly, obtaining the true color images at different scales through wavelet multi-scale edge detection algorithm, then based on the improved soft threshold filter function, selecting appropriate threshold of the obtained image edges to perform noise reduction while enhance the edge details of the reservation; and finally, carrying out the weighted 2-norm fusion of edges of different-scale-image. Wang and Yan (2012) [7] has presented a new edge detection approach based on vector morphological operators in color image processing. A new vector ordering in RGB color space has proposed. And then by analysing the characteristics of the noise contaminating image, vector morphological operators has proposed and these operators are applied in color edge detection. "Kamboj et al. (2012) [8] has proposed a method for edge detection of color images with automatic threshold detection. The proposed algorithm extracts the edge information of color images in RGB color space with fixed threshold value. The algorithm works on three channels individually and the output is fused to produce one edge map. The algorithm uses the improved Kuwahara filter to smoothen the image, Sobel operator is used for detecting the edge. A new automatic threshold detection method based on histogram data is used for estimating the threshold value. Edge detection reduces the amount of data needed to process by removing unnecessary features. Edge detection in color images is more challenging than edge detection in gray-level images. Compared with gray image, color image provides more edge information of objects. However, the current color edge detection algorithms acquired so much time to compute and they are hardly used in real-time system. In order to improve the efficiency and the performance of the color edge detection. Stock and Koch (2013) [9] has presented the automatic acceleration of a color edge-detection algorithm from a GA description. From a very concise GA description of the color edge detection algorithm, the synthesized accelerator achieved not only a speed-up of 11.8x against highly optimized software running on a current-generation processor, but also beat a programmable GA accelerator ASIC by a speed-up of 350x even when constraining the FPGA to run at the same clock rate. Using the hardware back-end, Gaalop synthesizes an algorithm specific accelerator architecture from the CLU Script DSL. It easily allows experimentation with different number formats, including support for the automated optimization of fixed-point representations. Geometric Algebra (GA) is a branch of mathematics that generalizes complex numbers and quaternions. One of the advantages of the framework is, that it allows intuitive description and manipulation of geometric objects. While even complex operations can be described concisely, the actual evaluation of these GA expressions is extremely compute intensive. However, it has significant fine-grained parallelism, which makes it a profitable target for hardware implementation. Hao et al. (2013) [10] has studied that the premise of obtaining the clear object contour in traditional Canny operator is to set appropriate parameters, does not have the adaptive ability. An adaptive Canny edge-detection method is proposed which Based on Canny theory. Adopt the 3*3 neighborhood instead of canny algorithm in 2*2 neighborhood to calculate the calculation gradient. Then, the maximum between-class variance (Otsu) method is used to obtain the high and low thresholds. Fu et al. (2013) [11] has compared the two improved methods, which are improved Sobel operator and improved wavelet transform using the multi-scale morphological filtering, subjective visual have achieved better results. However there are advantages and disadvantages in objective evaluations. So improvement is further done by using two improved methods with the wavelet transform fusion technology. The experimental results has shown that the fused image has increased significantly in information entropy and the average gradient compared to the improved Sobel operator, and it also has improved the peak signal to noise ratio and the distortion degree compared to the improved wavelet edge detection method. The fused image can concentrate the advantages of the two improved methods together and make complementary advantages. Eventually, the good de-noising effect and complete edge are achieved. Abid et al. (2013) [12] has proposed a new method for image edge detection based on multilayer perceptron (MLP). The method is based on updating a MLP to learn a set of contours drawn on a 3×3 grid and then take advantage of the network generalization capacity to detect different edge details even for very noisy images. The method is applied first to Gray scale images and can be easily extended to color ones. The method works well even for very low contrast images for which other edge operators fail. Franchini et al. (2013) [13] has proposed a hardware implementation of an edge detection method for color images that exploits the definition of geometric product of vectors given in the Clifford algebra framework to extend the convolution operator and the Fourier transform to vector fields. The proposed architecture has been prototyped on the Celoxica Field Programmable Gate Array (FPGA) board. The proposed hardware architecture allows for an average speedup ranging between 6x and 18x for different image sizes against the execution on a conventional general-purpose processor. Clifford algebra based edge detector can be exploited to process not only color images but also multispectral gray-scale images. Lei and Fan (2014) [14] has proposed a novel color edge detection method based on the fusion of hue component and principal component analysis to solve the problems. First, a novel computational method of hue difference is defined, and then it is applied to classical gradient operators to obtain accurate edges for hue component. Moreover, complete object edges can be obtained by using the edge fusion of the first principalcomponent and hue component of color image with low-computational complexity. The proposed gradient operators are found to be very effective to obtain better edge results for color images.

III. PROPOSED METHODOLOGY

A. Input color image

Firstly, we input a coloured image to the proposed algorithm.

B.To Apply dynamic thresholding

The dynamic thresholding method is used for binarizing gray level images. The procedure consists of computing the average gray value of a square of fixed size surrounding the pixel for each pixel of an image. The pixel value is then set to black or white, according to whether its initial value was larger than the surrounding average or not.

Firstly MATLAB's 'rgb2gray' function is used to convert spectrogram RGB image to gray scale image with pixel values ranging from 0 to 255; where 0 represents a fully black pixel and 255 represents white pixel. Then 'im2bw' function is used with a threshold to convert gray scale image to binary image. Dynamic thresholding is calculated by :

t = 0.5*[min(f) + max(f)] …..………..…………..(1)

Where,

t = 0.5, is a threshold value and f is a input image.

C. To Apply dynamic thersholding to get binarized image

Binarized images are images that have been quantised to two values, usually denoted 0 and 1,but often with pixel values 0 and 255, representing black and white. Binary images are used in many applications since they are the simplest to process, but they are such an impoverished representation of the image information that their use is not always possible. However, they are useful where all the information you need can be provided by the silhouette of the object and when you can obtain the silhouette of that object easily. Binary images are typically obtained by thresholding a grey level image. As shown below equation g_1 ,g_2& g_3 is binarized image. Binarized image g_1 is calculated by the given equation:

If $f(i,j)$ is greater then t then,

$g_1(i,j) = f(i,j) + fd^*(u_1(i,j)^{-\frac{1}{fe}} - 1)$ ....(2)

Otherwise,

$g_1(i,j) = f(i,j) - fd^*(u_1(i,j)^{-\frac{1}{fe}} - 1)$ ....(3)

Where ,

$g_1(i,j)$ is binarized image,$f(i,j)$ is input image,fd is maximum dynamic limit,fe shows an error and u_1 ,u_2 & u_3 is the variable of sub image.

To calculate g_2 :

If $f(i,j)$ is greater then t then ,

$g_2(i,j) = f(i,j) + fd * (u_2(i,j)^{-\frac{1}{fe}} - 1)$....................(4)

Otherwise,

$g_2(i,j) = f(i,j) - fd * (u_2(i,j)^{-\frac{1}{fe}} - 1)$....................(5)

To calculate g_3 :

If $f(i,j)$ is greater then t then,

$g_3(i,j) = f(i,j) + fd * (u_3(i,j)^{-\frac{1}{fe}} - 1)$……..….(6)

Otherwise ,

$g_3(i,j) = f(i,j) - fd * (u_3(i,j)^{-\frac{1}{fe}} - 1)$……..…(7)

D. To apply canny on binarized image g_1, g_2 & g_3

$D_1 = [edge\ (g_1, 'canny')]$……………………..(8)

$D_2 = [edge\ (g_2, 'canny')]$……………………..(9)

$D_3 = [edge\ (g_3, 'canny')]$……………….…….(10)

$D_4 = [edge\ (f, '\ canny')]$……………………..(11)

Where,

D_1, D_2 & D_3 is output of the binarized image g_1, g_2 & g_3 from the improved canny edge detector.

E. To Apply edge fusion

Fusion is the process of combine significant information from two or more images into a particular image. The consequential image will be more useful than any of the input images.

Apply improved canny on the step IV all the binarized images edges are detected and in this step apply edge fusion to get fused image by the given equation:

$D = D_1 + D_2 + D_3 + D_4$..(12)

Where ,

D is improved canny to apply on the different binarized image .

F. Then find the Final edge detected image

The output is the final edge detected image. Mathematically, the resultant image is given by :

$final = D + D_4$..(13)

IV. Results And Discussions

The proposed technology has been design and implemented in metlab using image processing tool box. Subsequent section contain experimental result and performance evaluation existence and proposed technique.

V. Experimental Set-up

Here is some representation of proposed algorithm to detect edges in a color image. Firstly input image is shown below:-

Fig (a): shows the input image

Figure (b) shows the hue image of given input image, the hue image describes the true color of an image. From color image to grey-scale image, leads to the outcome that a few edges are missed. In addition, mainly of the missing edges consequence from hue changes. Also the pixel values are same in grey scale images. So to detect missed edges we apply hue analysis to color image.

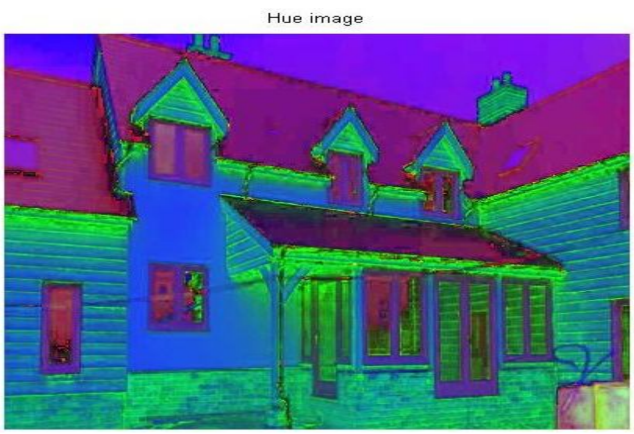


Fig (b): Hue Image

Figure (c) shows the hue detected edge image, we can present a superior edge detection representation for color image once the problem of edge detection of hue component is solved.

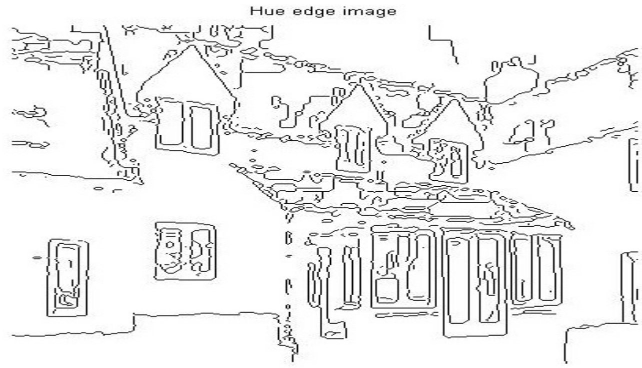


Fig: (c): Hue Detected Edge Image

Figure (d) shows the PCA image of the given input image, the PCA describes the major component information of a part in an image. The three color components are condensed into one containing a main component of information. So to detect missed edges we apply PCA analysis to color image.

Fig (d): PCA Image

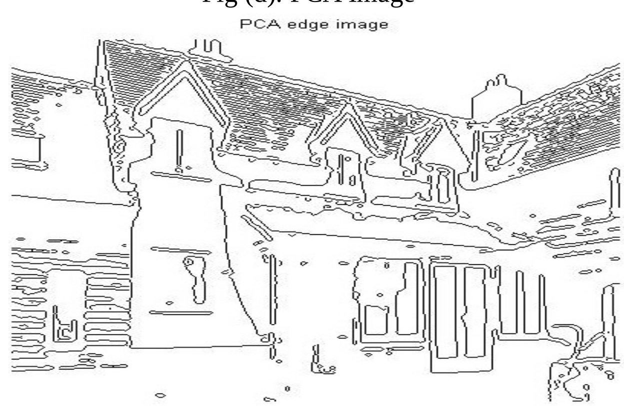


Fig (e): PCA Edge Detected Image

Figure (e) shows the PCA detected edge image, we can present a superior edge detection representation for color image once the problem of edge detection of PCA component is solved.

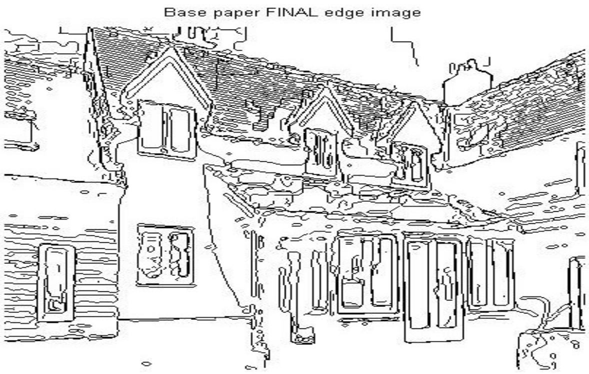


Fig (f): Fusion of Hue and PCA Edge Image

Figure (f) shows fusion of hue and PCA edge image combine significant information from two or more images into a particular image and gets clearer results to detect edges in color image.

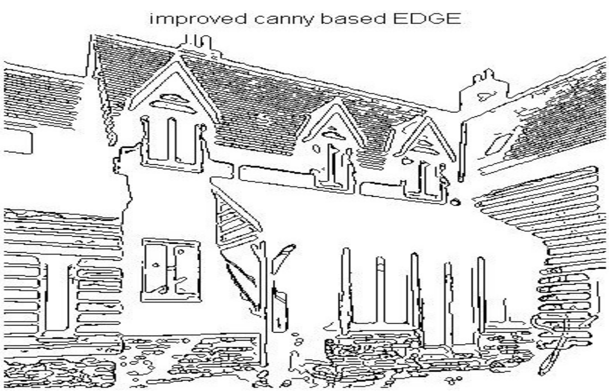


Fig (g): Improved Canny Based Edge Detection.

Figure (g) shows improved canny based edge describes. The improved canny detection detects fine edges in an image. It preserves edges of an image even in low intensity images

Fig (h): Final Edge Detected Image

Figure (h) gives the final result of edge detection in a color image by fusion of hue component, PCA and improved canny based edge detection.

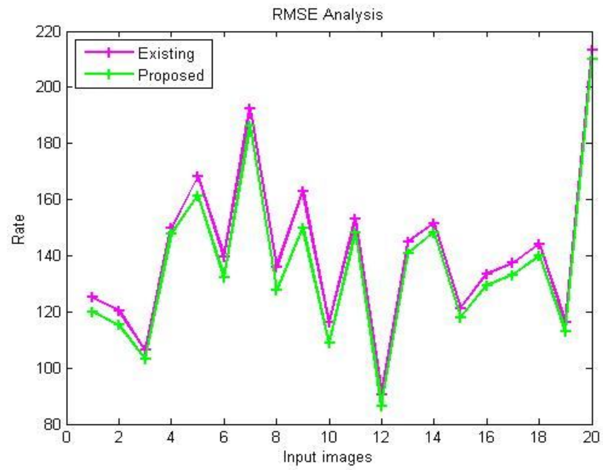
VI. PERFORMANCE EVALUATION

(a) Root Mean Square Evaluation

Table 1 RMSE values of many images

Images	**Existing Edge Detection**	**Proposed Edge Detection**
1. jpg	125.2078	120.0840
2. jpg	120.4907	115.3661
3. jpg	106.3626	103.2256
4. jpg	149.8299	147.6897
5. jpg	168.1220	161.0595
6. jpg	139.4274	132.3054
7. jpg	192.5643	186.4110
8. jpg	135.8713	127.8087
9. jpg	162.9663	149.8312
10. jpg	116.1163	109.0129
11. jpg	153.2123	148.0882
12. jpg	90.2940	86.1887
13. jpg	144.9931	140.8033
14. jpg	151.3968	148.3175
15. jpg	121.0331	117.7808
16. jpg	133.6114	129.4241
17. jpg	137.3244	133.0474
18. jpg	144.2186	139.9236
19. jpg	116.1034	112.9181
20. jpg	213.1525	210.0117

Table 1 shows the result of RMSE for many images, the two corresponding values for each image shows the difference between the existing and proposed Edge Detection. Here in this the values are decreased for each image from existing to proposed Edge Detection.



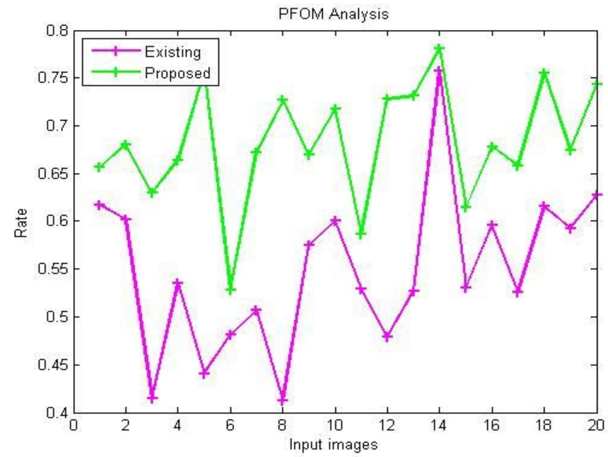
Graph 1: RMSE Evaluation

Graph 1 shows the various images on X-axis and rate on Y-axis, the two graphs indicate the results according to the values of existing and proposed Pratt of previous RMSE table and results shows the Pratt figure of merit for RMSE.

(B) Comparison of PFOM values of edge images by different methods

Table 2: PFOM values of many images

Images	Existing Edge Detection	Proposed Edge Detection
1. jpg	0.61730	0.6563
2. jpg	0.6022	0.6803
3. jpg	0.4145	0.6293
4. jpg	0.5349	0.6639
5. jpg	0.4408	0.7551
6. jpg	0.4811	0.5281
7. jpg	0.5068	0.6725
8. jpg	0.4122	0.7265
9. jpg	0.5743	0.6694
10.jpg	0.6001	0.7173
11.jpg	0.5289	0.5870
12.jpg	0.4786	0.7279
13.jpg	0.5270	0.7313
14. jpg	0.7578	0.7810
15. jpg	0.5306	0.6143
16. jpg	0.5958	0.6777
17. jpg	0.5254	0.6582
18. jpg	0.6154	0.7555
19. jpg	0.5929	0.6738
20. jpg	0.6274	0.7434



Graph 2: PFOM Evaluation

VII. CONCLUSION AND FUTURE SCOPE

Edge detection is a fundamental tool used in most image processing applications to obtain information from the frames as a precursor step to feature extraction and object segmentation. By conducting the survey of different research papers on the new edge detection techniques it has been found that the most of the existing work has been based on gray images so effect of true color has been neglected, moreover the use of hybrid canny for edge detection has also been neglected. The significant improvement of the algorithm over the existing edge detectors has been proved with the help of various parameters. This work has not considered any artificial intelligence based algorithm, so in near future, fuzzy logic, genetic algorithm and particle swarm optimization edge detection can be included.

REFERENCES

[1] Wesolkowski, Slawo, M. E. Jernigan, and Robert D. Dony. "Comparison of color image edge detectors in multiple color spaces." In *Image Processing, 2000. Proceedings. 2000 International Conference on*, vol. 2, pp. 796-799. IEEE, 2000.

[2] Dikbas, Salih, Tarik Arici, and Yucel Altunbasak. "Chrominance edge preserving grayscale transformation with approximate first principal component for color edge detection." In *Image Processing, 2007. ICIP 2007. IEEE International Conference on*, vol. 2, pp. II-261. IEEE, 2007.

[3] Chen, Xin, and Houjin Chen. "A novel color edge detection algorithm in RGB color space." In *Signal*

Processing (ICSP), 2010 IEEE 10th International Conference on, pp. 793-796. IEEE, 2010.

[4] Kolomenkin, Michael, Ilan Shimshoni, and Ayellet Tal. "On edge detection on surfaces." In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 2767-2774. IEEE, 2009.

[5] Dezert, Jean, Zhun-ga Liu, and Grégoire Mercier. "Edge detection in color images based on DSmT." In *Information Fusion (FUSION), 2011 Proceedings of the 14th International Conference on*, pp. 1-8. IEEE, 2011.

[6] Xiao, Feng, Mingquan Zhou, and Guohua Geng. "Detail enhancement and noise reduction with true color image edge detection based on wavelet multi-scale." In *Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC), 2011 2nd International Conference on*, pp. 1061-1064. IEEE, 2011.

[7] Wang, Lei, and Lingbo Yan. "Edge detection of color image using vector morphological operators." In *Computer Science and Network Technology (ICCSNT), 2012 2nd International Conference on*, pp. 2211-2215. IEEE, 2012.

[8] Kamboj, Anil, Kavita Grewal, and Ruchi Mittal. "Color Edge Detection in RGB Color Space Using Automatic Threshold Detection." *International Journal of Innovative Technology and Exploring Engineering (IJITEE)* ISSN: 2278-3075, Volume-1, Issue-3, August 2012

[9] Stock, Florian, Andreas Koch, and Dietmar Hildenbrand. "FPGA-accelerated color edge detection using a Geometric-Algebra-to-Verilog compiler." In *System on Chip (SoC), 2013 International Symposium on*, pp. 1-6. IEEE, 2013.

[10] Hao, Geng, Luo Min, and Hu Feng. "Improved Self-Adaptive Edge Detection Method Based on Canny." *In Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2013 5th International Conference on*, vol. 2, pp. 527-530. IEEE, 2013.

[11] Fu, Zuo-Xian, Cai-Xia Deng, and Yuan-Yan Tang. "A fusion edge detection method based on wavelet transform and differential." *InWavelet Analysis and Pattern Recognition (ICWAPR), 2013 International Conference on*, pp. 17-21. IEEE, 2013.

[12] Abid, Sabeur, Farhat Fnaiech, and Ezzeddine Ben Braiek. "A novel neural network approach for image edge detection." In *Electrical Engineering and Software Applications (ICEESA), 2013 International Conference on*, pp. 1-6. IEEE, 2013.

[13] Franchini, Silvia, Antonio Gentile, Giorgio Vassallo, Filippo Sorbello, and Salvatore Vitabile. "A Specialized Architecture for Color Image Edge Detection Based on Clifford Algebra." In *Complex, Intelligent, and Software Intensive Systems (CISIS), 2013 Seventh International Conference on*, pp. 128-135. IEEE, 2013.

[14] Lei, Tao, Yangyu Fan, and Yi Wang. "Colour edge detection based on the fusion of hue component and principal component analysis." *IET Image Processing* 8, no. 1 (2014): 44-55.