RESEARCH ARTICLE OPEN ACCESS

Cartoonization Using LBG And KPE Vector Quantization

Dr. Archana B. Patankar*, Ms. Purnima Tawde**
*(Associate Professor Computer Department, Thadomal Shahani Engineering College, Mumbai University, Mumbai-50)
** (M.E Student Computer Department, Thadomal Shahani Engineering College, Mumbai University, Mumbai-50)

ABSTRACT
In this paper we introduce a non photorealistic rendering technique that can automatically deliver a cartoon effect stylization of a photograph. The method proposed generates a cartoonized image by using vector quantization and line extraction algorithms. Image cartoonization is achieved by applying LBG (LindeBuzo Gray) and KPE (Kekre's Proportionate Error) vector quantization algorithms along with gradient edge detection method. Our method is simple and easy to implement. Experimental results are presented to show the effectiveness of our method.

Keywords- Cartoon effect, color space conversion, line drawing, non-photorealistic rendering, vector quantization

I. INTRODUCTION

Non-Photorealistic Rendering (NPR) is a part of Image processing and Computer vision it involves working with visual information. NPR in general involves abstraction and stylization of raw images. NPR is an area of computer graphics that focuses on enabling a wide variety of expressive styles for digital art which includes computer generated image shading, animation and painting. In this paper we introduce a framework that uses vector quantization and edge detection method to automatically produce a cartoon like effect on a given true colored image.

Cartoon effect stylized images can be used to generate paintings, drawing, technical illustration, and animated cartoons. Stylized images not only prove useful in the field of art and creativity but also can be beneficial in the field of visualization, architectural illustration and animated explanation of experiments[4] [5]. Movie to comic conversion software and computer games also require cartoonized image to improve their work [2]. These applications require normal camera generated true colored images to be converted to cartoon form.Itcan be obtained by applying method we have presented in this paper.

1.1 Problem Statement

Given a true colored image as input, that contains specifications of the coordinates values andtheir intensities. The expected output is cartoon effect stylization of the input image.

Line Extraction: The task of line extraction is to locate discontinuities in the image intensities. The pixel where brightness of the image changes sharply represent the edges that are to be extracted[8][6][20].

Vector Quantization: Vector quantization (VQ) is a lossy data compression method. VQ is the process to approximate values of intensities of colors to form clusters of different colors[11][12].

The vector quantization when applied on an input image forms the clusters of colors which lead to color reduction. The output produced by vector quantization when mapped with the edges extracted from the input image provides a border like effect on the image, which makes the image look cartoonized.

1.2 Related Work

There has been tremendous research in the area of Non Photorealistic Rendering (NPR). Among the earliest work on image-based NPR was that of Saito and Takahashi, in 1990 they performed image processing operations on data buffers derived from geometric properties of 3D scenes [3]. Several image stylization systems have been proposed, mainly to help artists with labor-intensive procedures**.**The system introduces by DeCarlo and Santella in the year 2000 extended the mean-shift based stylization approach to obtain cartoonized images[4]. Real time image abstraction is also a great work of NPR where HolgerWinnem¨oller, Sven C. Olsen and Bruce Gooch have cartoonized images using bilateral filters and difference of gaussian edge detection filter [1]. In Stylized augmented reality for improved immersion, Fischer has used Canny's edge detection and bilateral filtering to generate NPR result [7].Artistic edge corner preserving and smoothing method have used Kuwahara filter to obtain cartoon stylization of images [9].Our work is also related to the research done in NPR and is based on work done to detect lines in an image and vector quantization of an image. Our work presents a framework to obtain cartoonized images from given true colored images.

II. METHOD FOR CARTOONIZATION

The basic workflow of our framework is shown in (Fig 1). We first apply Kekre's RGB to LUV color space conversion to the input image. The result of first step is given to vector quantization step. LindeBuze Gray (LBG) or Kekre's Proportionate Error (KPE) is used to form clusters of colors in the image. The output obtained in step two is then converted from LUV to RGB color space. The result obtained at step three is then enhanced by adjusting sharpness and contrast. Edge detection is done to obtain prominent boundaries in the image. Gradient edge detection method is applied to obtain the edges. The result of gradient edge detection and step four that is color adjustment is combined to obtain the final cartoonized image stylization.

2.1 Kekre's LUV Color Space

Kekre's LUV Color Space is a special case of Kekre transform. Where L represents luminance, U and V represent chromaticity values in a colored image. Equation (1) shows the RGB to LUV conversion matrix of Kekre's color space.Conversion of Kekre's LUV color space back to RGB color space is done by using matrix given in Equation (2), [16] [17].

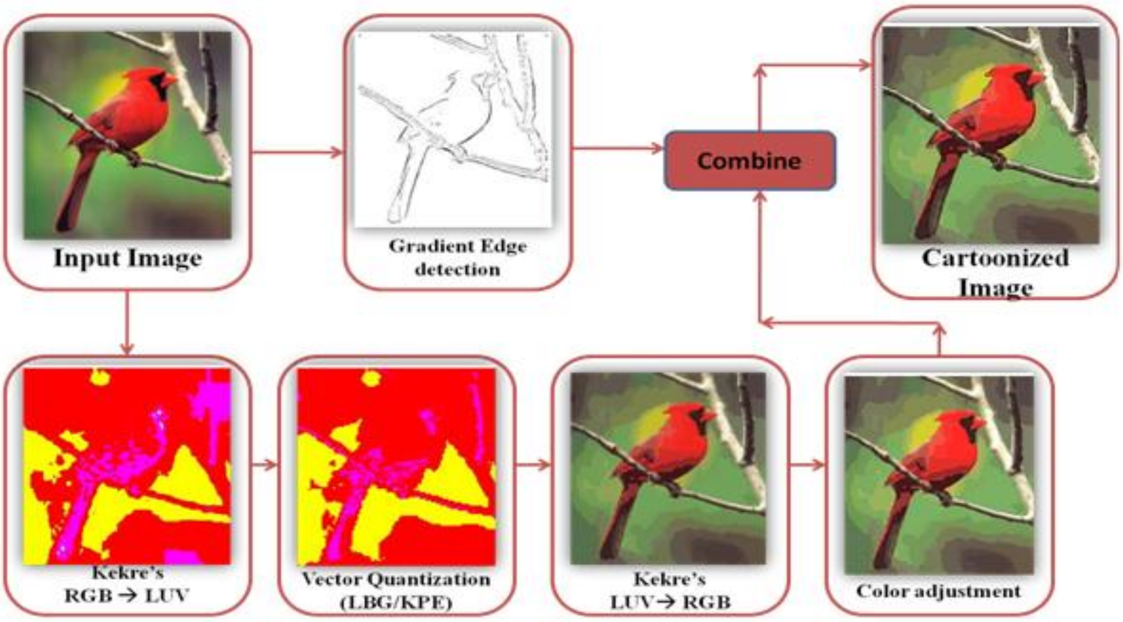

Fig 1: Cartoonization Framework Overview.Each step is the function performed to obtain a cartoonized image. Image in the right top block is the final result of cartoonization process.

RGB to LUV conversion matrix

$$\begin{bmatrix} L \\ U \\ V \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ -2 & 1 & 1 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \qquad (1)$$

LUV to RGB conversion matrix

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 & -2 & 0 \\ 1 & 1 & -1 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} L/3 \\ U/6 \\ V/2 \end{bmatrix} \qquad (2)$$

2.2 Vector Quantization

Vector quantization is a classical quantization technique from signal processing which allows the modeling of probability density functions by the distribution of prototype vectors. It was originally used for data compression. It works by dividing a large set of points (vectors) into groups which have approximately the same number of points closest to them. Each group is represented by its centroid point, as done in k-means and some other clustering algorithms [16] [18].

2.2.1 LindeBuzo Gray (LBG) algorithm

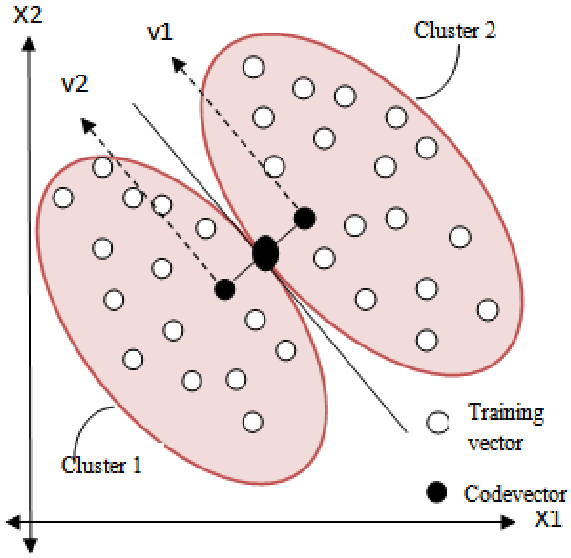

Fig 2: LBG for two clusters

In this algorithm, from the initial feature vector (training set) of an image, centroid is computed as a first code vector. By adding constant error to the code vector two code vectors V1 and V2 are generated Euclidean distance for all training vectors is computed with code vectors V1 and V2. Based on the smallest distance, training vectors are clustered into either V1 or V2. This process is repeated till the codebook of desired size is obtained. In Fig 2, two vectors v1 & v2 are generated by adding constant error to the codevector[11] [13].

2.2.2 Kekre's Proportionate Error (KPE) algorithm

Let T= {X1, X2,…., XM} be the training sequence consisting of M source vector. Assume thatsource vector is of length K, Xm is given as Xm={xm,1, xm,2, …… xm,K} for m=1, 2, ………. , M. In this algorithminitial codevector is computed by taking the mean of all the training vectors Xi for i=1, 2, …, M.Thus initially the codebook contains only one codevector. Then two vectors from the codevectorare computed by adding proportionate error instead of adding constant. From the codevectorproportions between the members of vector is calculated.Let k be the length of codevector, C = {c1, c2, ……..,ck} be the codevector, and E = {e1, e2, ……., ek} be the error vector. cj = min{ci / i= 1,2, …. k} where j is the index of the member of vector whose value is minimumamong the vector members.Then assign ej = 1and if ci / cj≤ 10 then assign ei = ci / cj else assign ei = 10 for i ≠ j and i=1,2,………, k.

Two vectors v1 and v2 are formed, v1 is formed by adding the error vector E to codevector C and v2 is formed bysubtracting the error vector E from codevector C.Euclidean distance between the all the training vectors Xi with v1 and with v2 are computedi.e. d1=‖ v1-Xi ‖2 and d2=‖ v2-Xi ‖2 for i=1, 2, ….., M. If d1 < d2 then Xi is put in cluster1 else Xi is put in cluster2 and two clusters are created. The above process is repeated till codebook of desired size is obtained[14][15][20].

2.3 Edge detection

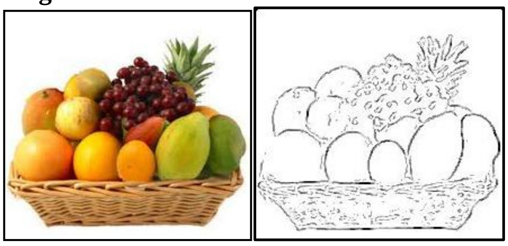

Fig 3 Result of gradient edge detection Method

$$Gx = (\partial xI) \quad (3)$$

$$Gy = (\partial yI) \quad (4)$$

$$G = ((Gx)^2 + (Gy)^2)^{\frac{1}{2}} \quad (5)$$

$$E = \begin{cases} 1 - G & if\ G < T \\ 1 & otherwise \end{cases} \quad (6)$$

In gradient methods we have to first compute directional gradients, Gx and Gy, with respect to the x-axis and y-axis. The x-axis is defined along the columns going right and the y-axis is defined along the rows going down. The gradient magnitude and direction are then computed from their orthogonal components Gx and Gy . Given below in Equation (3), (4)and (5),where I is the grayscale input which is smoothened to be able to extract strong prominent edges and discard weak edges. ∂x and ∂y are gradient operators in two directions, implemented by forward difference. T is a threshold value to obtain continuous edges. $G$is the gradient value calculated using Equation (5). Value of Egiven by Equation (6), is the actual edge detection output to be mapped on result obtained in vector quantization result. The result obtained by gradient edge detection method is sharpened to make the edges look sharp and clear [21].

III. RESULTS

In section III, Fig 4, we have shown the result of applying framework given in Fig 1 on test images TABLE 1, shows the comparison of time taken in seconds to process the given image. Second column of the tablegives time taken by using vector quantization algorithms LindeBuzo Gray (LBG) and Gradient edge detection to cartoonize the test image. Second column showstime taken to cartoonize test image by using Kekre's Proportionate error and third column shows image cartoonization obtained by using Bilateral filter and Difference of Gaussian [1] [10] [19]. Time required to cartoonize image by framework given in Fig 1 is less.

(a) **(b)** **(c)**

(d) **(e)** **(f)**

(g) **(h)** **(i)**

(j) (k) (l)

Fig 4: Original Images (a)(d)(g)(j) , Cartoonization using LBG (b)(e)(h)(k), KPE(c),(f),(i),(l)

Table 1 : Comparison of time taken for Cartoonization of given Image by LBG, KPE and Bilateral filter (BL)

Test Image	LBG (seconds)	KPE (seconds)	BL (seconds)
Sid	5	5	9
Forest	4.8	4.7	7
Cardinal	6	6	8.9
Cake	5.4	5.2	7
Lena	6.2	6	8
fruits	6.5	6.6	8.2

IV. CONCLUSION

The method given by us in this paper represents a simple andeffective framework that produces painting effect on the true colored image making it look like a cartoon. It converts the image in cartoon effect at the same time it retains all the important information in the image. It can automatically abstract a given image and can be beneficial in the field of Non photorealistic rendering. Considering the results obtained, cartoonization framework using LBG and KPE produces a good cartoon stylization effect on the test images given. From details given in Table 1, it can be concluded that our method takes less time to automatically generate a cartoonized image. The result obtained by our method can be used for benefit of artist to produce cartoonized images and for other work related to NPR.

REFERNCES

[1] HolgerWinnemoller, Seven C Oslen and Bruce Gooch, Real time video abstraction, *Proc. ACM SIGGRAPH'06,* pp. 1221-1226, 2006.

[2] Richard Hong, Xiao-Tong Yuan, Shuicheng Yan and Tat Seng Chua, Movie2Comic: Towards lively video content presentation. *IEEE trans. Multimedia 14(3),* 2014.

[3] Saito,Tand Takahashi.T, Comprehensible rendering of 3D shapes, *Proc. ACM SIGGRAPH'90,* pp. 197-206, 1990.

[4] Santella.A and DeCarlo.D, Visual Interest and NPR: an evaluation manifesto, *Proc. ACM SIGGRAPH'06,* pp. 1221-1226, 2006.

[5] Hertzmann.A, Paint by relaxation, *Computer Graphics Internat*.2001, 47-54.

[6] Marr .D and Hildeth. E.C, Theory of edge detection, *Proc. Royal Soc.London,Bio. Sci.* pp. 187-217, 1980.

[7] Fischer .J, Bartz .D and Strasser .W, Stylized Augmented reality for Improved Immersion, *Proc. IEEE Virtual Reality'06,* pp. 195-205, 2005.

[8] Canny .J.F, A Computational approach to edge detection, *IEEE trans. Pattern Recognition and Machine Intelligence, 8(6), 1986*.

[9] Kyprianidis J E, Kang H, D¨ollner J, Anisotropic Kuwahara filtering on the GPU, *GPU Pro—advanced rendering techniques*, 2010.

[10] Tomasi .C and Manduchi .R, Bilateral filtering for gray and colored images, *Proc. ICCV'98,* pp. 839-842, 1998.

[11] YosephLinde,AandresBuzo, An algorithm for vector quantizer design, *IEEE Trans Communication, 28(1), 1980*.

[12] M. Gray, Vector quantization, *IEEE ASSP Magazine, 28(1), 4-29, 1980*.

[13] Dr. H. B. Kekre, ArchanaPatankar, Hema Ramesh Galiyal, Segmentation of blast using vector quantization technique,*International Journal of Computer Application Security*, 72(15), 2013.

[14] Dr. H.B Kekre, Tanuja K. Sarode, Bhakti Raul, Color Image Segmentation using KPE for vector quantization, *International Journal of Computer Science. 3(4),2008, 287-292*.

[15] Dr. H.B Kekre,Tanuja K. Sarode, New clustering algorithm for vector quantization using rotation of error vector, *International Journal of computer science and information security,7(3),2010*, 159-165.

[16] Dr. H.B Kekre,Dr. SudeepThepade, SanchitKhandelwal, Improved Face Recognition with Multilevel VTC usingKekre's LUV color space, *International Journal of Advanced Computer Science and Application, 3(1),2012*.

[17] H.B.Kekre, Sudeep D. Thepade, Boosting block truncation coding using Kekre‟s LUV Color Space for Image Retrieval, *WASETInternational Journal of Electrical, Computer and System Engineering), 2008, 172-180*.

[18] Dr. H.B Kekre,TanujaSarode, Real time video abstraction, Vector Quantizated Codebook Optimization using K-means, *International Journal of Computer Science and Engineering,2009.*

[19] Henry Kang, Senugyong Lee and Charles K Chui, Flow based Image Abstraction,*IEEE Trans. Visualization and Computer Graphics*, 15(1), 2009.

[20] Dr. H.B Kekre,TanujaSarode,Two –level vector quantization method for codebook generation using KPE algorithm, *International Journal of Image processing*,4(1), 2010.

[21] Cewu Lu ,Li Xu and JiayaJia, Combining sketch and tone for pencil drawing production,*International Symposium on Non-Photorealistic Animation and Rendering (NPAR 2012)*, 2012.