Feature Analysis of Chromatic or Achromatic Components based on Tensor Voting and Text Segmentation using Separated Clustering Algorithm

Jonghyun Park, Nguyen Trung Kien, Jaemyeong Yoo, Gueesang Lee

Dept. of Computer Science, Chonnam National University, Korea
jhpark@new21.com, trung_kien_kg@yahoo.com, gslee@chonnam.ac.kr

Abstract

This paper presents a new technique for segmenting corrupted text images on the basis of color feature analysis by second order tensors. It is show how feature analysis can benefit from analyzing features using second order tensor with chromatic and achromatic components. Proposed technique is applied to text images corrupted by manifold types of various noises. Firstly, we decompose an image into chromatic and achromatic components. Secondly, we analyze color features by second order tensors, and remove noises using a vector median. Lastly, mode estimation and segmentation are performed by adaptive mean shift and separated clustering method respectively. The experimental results show that proposed approach is efficient and robust in terms of restoring and segmenting corrupted text images.

Keywords: segmentation, second order tensor, mode detection, region inference.

1. Introduction

The human visual perception system performs extraordinarily well in spotting out and recognizing the miscellaneous objects that natural scene images. Artificial intelligence systems find it not so straightforward to recognize objects in images, even in the case of simple scenes. In general, segmentation refers to the low-level tasks of partitioning an image into disjoint and homogeneous regions which should be meaningful for a given application; this operation is usually preliminary to higher-level tasks such as object recognition, classification, and semantic interpretation. For the last decade, has been a remarkable growth of algorithms for segmentation of color images. Recent research are performed as various techniques: for example, stochastic model based methods [Belongie el. Al. (1998)][Delignon el. Al. (1997)][Panjwani el. Al. (1995)][Wang et. Al. (1998)][Zhu et. Al.], morphological watershed based region growing method [Shafarenko et. Al. (1997)], energy diffusion method [Ma et. Al. (1997)], and graph partitioning method [Shi et. Al. (1997)]. Quantitative evaluation methods have also been suggested [Borsotti et. Al. (1998)]. However, because of difficult nature of the problem, there are few automatic algorithms that can operate

well on corrupted natural images. The problem of segmentation is not simple because of corrupted by noises such as graffito, streaks, corrosion, and so on. If a natural image contains damaged region in some homogeneous color regions, clustering methods in a color space such as [Comaniciu et. Al. (1997)] are not sufficient to handle the problem. In general, natural scene images have various objects, among them; text is important sign since they communicate important information for understanding image. The fact has inspired many efforts on text recognition in static images, as well as video sequences [Zhong et. Al. (1995)][Jain et. Al. (1998)][Haritaoglu (2001)]. In [Ye et. Al. (2004)], Qixiang Ye *et al*. use Gaussian mixture models (GMMs) in HSI color space with spatial connectivity information to segment characters from a complex background. They do not explicitly consider the fact that texts in images can be severely corrupted by noises. In such cases, texts may not be segmented as separate objects due to the corruption of noises which may cause errors when used as input in optical character recognition, as mentioned in the future work in [Lucas et. Al. (2003)]. Hence, we propose restoration and segmentation from corrupted text images using hue and intensity of color components in RGB space. In order to restore, we use tensor voting method using second order tensors for describing tokens as the given input data. Tensor voting have been performed by G.G. Medioni *et al*. in [Medioni et. Al. (2000)][Lee et. Al. (1999)], and has been applied to various application fields such as the inference of object boundaries: as a consequence, its use can explain the presence of noises based on surface saliency in the image feature space. In our paper, noises can be removed by vector median method. Finally, restored images are segmented by separated clustering method.

2. Separation of color components

In color image processing, color of a pixel is presented as three values corresponding to the tristimuli R (red), G (green), and B (blue). Various kinds of color models such as intensity, saturation, and hue can be computed from the RGB components by using either linear or nonlinear transformations. And then, there are various models used for different purposes or to solve different problems in color image processing, such as HLS, XYZ, Lab, UVW, I1I2I3, and YUV. For color clustering, it is desirable that the selected color features define a space processing uniform characteristic. In general, natural color images are consisted of achromatic and chromatic regions. Using only the monochrome component for segmentation can be unstable in low level computer vision. Therefore, we propose a decision function for separating a pixel as chromatic or achromatic such that the appropriate feature is used in segmentation. Proposed method can separate color components based on the sum of differences by using R, G, and B components that is independent of illumination. The equation of decision function is described as

$$F(x,y) = \frac{|R(x,y)-G(x,y)|+|G(x,y)-B(x,y)|+|R(x,y)-B(x,y)|}{3} . \quad (1)$$

If the F(x,y) value is small, the pixel is the closer to the achromatic component. If it is not, the pixel is chromatic. In algorithm I, leaving a gap between two feature ranges prevents that achromatic and chromatic regions are overlapped during clustering. In addition, the values near 0.6 and 1.0 are clustered as one mode due to the cyclic property of hue component. The final values corresponding to one image in the range [0~1], called as chromaticity labeled image, are applied to the tensor voting in 3D.

ALGORITHM I : SEPARATING COLOR COMPONENTS
Step 1: *Read input image*;
Step 2: *If F(x,y) > TH1, F(x,y)=Hue(x,y);*
Else , F(x,y)=Int(x,y);
Step 3: *Chromatic components are normalized into the range 0.6 to 1.0*;
Achromatic components are normalized into the range 0.0 to 0.4;
Step 4: *Chromaticity labeled image from 0.0 to 1.0*;

3. Second order tensors and voting in 3D

3.1 Second order symmetric tensors in 3D

The second order symmetric tensors are an ellipsoid, which is fully described by its associated eigensystem, with three eigenvectors $(\hat{e}_1, \hat{e}_2, \hat{e}_3)$ and three eigenvalues $(\lambda_1, \lambda_2, \lambda_3)$.

$$\begin{bmatrix} \hat{e}_1 & \hat{e}_2 & \hat{e}_3 \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} \begin{bmatrix} \hat{e}_1^T \\ \hat{e}_2^T \\ \hat{e}_3^T \end{bmatrix} \tag{2}$$

Rearranging the eigensystem, the ellipsoid is given by

$$(\lambda_1 - \lambda_2)S + (\lambda_2 - \lambda_3)P + \lambda_3 B \tag{3}$$

where *S* defines a stick tensor, *P* defines a plate tensor and *B* defines a ball tensor:

$$S = \hat{e}_1\hat{e}_1^T, \; P = \hat{e}_1\hat{e}_1^T + \hat{e}_2\hat{e}_2^T \text{ and } B = \hat{e}_1\hat{e}_1^T + \hat{e}_2\hat{e}_2^T + \hat{e}_3\hat{e}_3^T \tag{4}$$

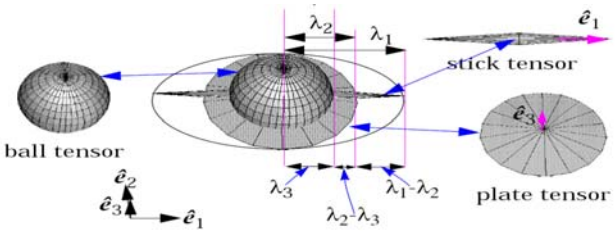


Figure 1. A second order generic tensor and its decomposition into the stick, plate and ball components in 3D.

These tensors define the three basis tensors for any general second order symmetric 3D tensor. For surface inferences, surface saliency is then given by $\lambda_1 - \lambda_2$, with

normal direction estimated as $\hat{e}_1$. Curve saliency is given by $\lambda_2 - \lambda_3$, with tangent direction estimated as $\hat{e}_3$, and junction saliency of curves estimated by the eigenvalue λ_3, which is encoded by the ball tensor. Figure 1 shows a second order symmetric tensor and its decomposition into the stick, plate and ball components in 3D.

3.2 Tensor Voting

The voting field defines the most likely normal by selecting a most likely continuation curve between two points *O* and *P* in Figure 2. The length of the normal vector at P, representing the strength of the vote, is define by the following equation in spherical coordinates:

$$DF(s,k,\sigma) = \exp\left(\frac{s^2 + ck^2}{\sigma^2}\right) \tag{5}$$

where is $s = (l\theta)/\sin(\theta)$ and $k = 2\sin(\theta)/l$. The parameter s is the arc length *OP*, k is the curvature, c is a constant, and σ is the scale of voting field controlling the size of the voting neighborhood and the strength of votes in Figure 2.

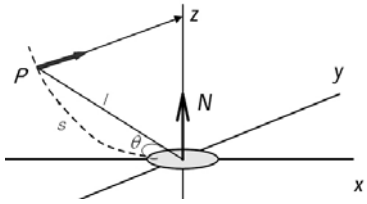


Figure 2. Second order votes cast by a stick tensor located at the origin.

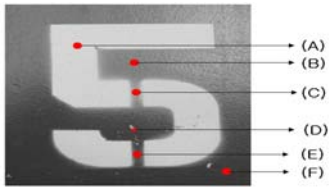
3.3 Feature extraction

Now that the most likely type of feature at each token has been estimated, we want to compute the dens structures that can be inferred from the tokens. This can be achieved by casting votes to all locations, whether they contain a token or not, using the same voting fields and voting mechanism. Then each site contains two features, indicating feature saliency and direction. We can infer a smooth structure that connects the tokens with high feature saliencies computed by using $\lambda_2 - \lambda_3$ for surface saliency. Given this dense information, in 3D we analyze true surface points and connect them. We detail feature analysis based on tensor voting for corrupted text images in next section.

4. Vector Median-based Surface Smoothing in Tensor Voting

In general, text region appears as homogeneous color components. However, the image may also be noisy, as the physical surface of the sign degrades due to corrosion, dirt, intentional defacing, and so on. The noise regions can be presented as severely different values. Though the noise regions appear with similar values, their

regions size is smaller than text regions. Noise regions can be regarded to be smaller than 10×10 and the size of characters by experimental experiences. The same result is achieved by tensor voting. Figure 3 is shown experimental result of surface saliency analyzed by second order tensor. Here, we used voting scale to vote of tokens with $\sigma = 10$. The empirical surface saliency analysis of pixels can be used to select a threshold for determining noise regions. From such an observation and the experimental data in Figure 3, we figure out that the threshold can be selected between [0~2.87] (here we use 1.5) when the maximum value of surface saliency in a given image is 4.29 based on that $\sigma = 10$. In addition, if the value of a selected threshold is close to 0, noise occupying broad homogeneous regions may not be removed clearly and then the remaining noise can be removed through iteration as mentioned above.

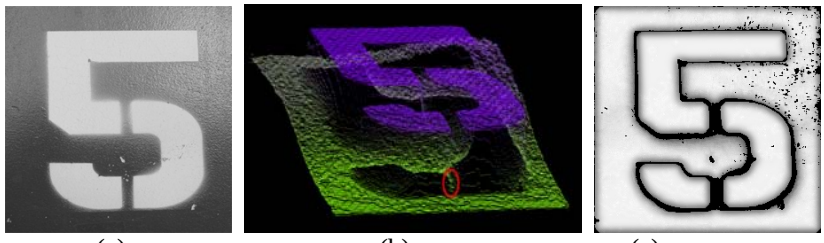


position	surface saliency
A	4.29
B	4.29
C	1.16
D	0.83
E	0.98
F	4.29

(a) (b)

Figure 3. Experimental result of surface saliency: (a) Captured text image, (b) Surface saliency at each position of (a).

After tensor voting as described in Figure 4(b), a "surface saliency map" defines surface saliency at every pixel in a given image. The map is able to indicate the presence of noise on text as in Figure 4(c) by black regions. The black pixels can be replaced by applying a *vector median filter* to the saliency information.



(a) (b) (c)

Figure 4. (a) original image, (b) the data in the tensor voting where red circle indicates a part of noises, (c) the normalized representation of chromaticity labeled image in the range [0.0-1.0] with black regions denoting noise.

In this Section, we describe a novel approach for smoothing surfaces using vector median from noise regions. Median and Order Statistic-based filters are extensively used in signal processing, especially image processing, due to their ability to reject outliers and preserve features such as edges and monotonic regions. Astola *et al.*

introduced the vector median filter as an extension of the median filter to multivariate data [Shen et. Al. (2004)]. For an observation window $\Omega = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N \in \Re^{\mathbf{m}}\}$, the output of the vector median filter is defined as

$$\mathbf{x}_{\mathbf{VM}} = \arg\min_{\mathbf{x}\in\Omega} \sum_{i=1}^{N} \left\| \mathbf{x} - \mathbf{x}_i \right\|_{\mathbf{p}} \tag{6}$$

where $\|\cdot\|_{\mathbf{p}}$ denotes the L_p norm. The vector median is a suboptimal estimate, in the maximum likelihood sense, of the location parameter of a multivariate Laplacian distribution. To find the vector median, the sum of L_p distances from each sample to all other samples is computed, $\mathbf{d}(\mathbf{x}_j) = \sum_{i=1}^{N} \left\| \mathbf{x}_j - \mathbf{x}_i \right\|_{\mathbf{p}}$, $j = 1,2,\cdots,N$; then, the vector median is set as $\mathbf{x}_{\mathbf{VM}} = \arg\min_{\mathbf{x}_j} \mathbf{d}(\mathbf{x}_j)$. Although this computation has a complexity of $\mathbf{O}(N^2)$, it performs well in practice and is not generally computationally prohibitive as the subwindow size $m \times n$ is usually a small number. In Figure 6, low surface saliency regions, black regions in Figure 6(c), are primarily considered noise and should be replaced with values of the high surface saliency neighbors. If a pixel is judged as noise, neighbors surrounding the pixel in a $(m \times n)$ subwindow are initially examined to find high surface saliency with which to replace the pixel. If the noise region is broad, however, the $(m \times n)$ subwindow may be insufficient to find high surface saliency defined by TH_s. The size of the subwindow $((m+s)\times(n+s))$ is therefore increased until the proper number of high surface saliency pixels is detected. The steps below represent the process of this proposed algorithm. Then, the vector median value among the high surface saliency pixels within the final window is selected. However, some noise may remain after a single pass of the tensor voting and vector median. We therefore repeatedly apply the filter to remove the remaining noise.

Algorithm II. RESTORING CORRUPTED REGION USING VECTOR MEDIAN

//It uses VECTOR MEDIAN to compute the most likely normal region at the token. Then ball tensors are computed by integrating the resulting normal votes cast by voter.//

STEP1: Compute the surface saliency in the $(m+s)\times(n+v)$ window surrounding the noise pixel. $S(x, y)$, $1 \le x \le m+s$,$1 \le y \le n+v$ (As an initial value, $m = n=3$ and $s=v=0$.)

//The number of high surface saliency pixel in the window.//

IF ($S(x, y) \ge TH_s$) count = count+1;

//The window size is changed for ***vector median.***//

IF (Count<TH_c) $s=s+2$, $v=v+2$, **GOTO STEP 1** (here, $TH_c = 8$.)

ELSE **GOTO STEP 2**

STEP2: Enumerate values of pixels corresponding high surface saliency in the increased window: $H(x, y)$, $1 \le x \le m+s$,$1 \le y \le n+v$

Finally, we find the vector median value among the enumerated values and fill the vector median in the noise pixels.

---

5. Text Image Segmentation

In this section, we briefly review the original mean shift-based density estimation show how mode of clusters is detected by density gradient estimation function [Comaniciu et. Al. (1997)]. And we describe a segmentation algorithm for text images based on separated clustering algorithm.

5.1 Adaptive Mean Shift Procedure

Assume that each data point $\mathbf{x}_i \in \Re^d$, $i = 1,\cdots,n$ is associated with a bandwidth $\mathrm{h}_i > 0$. The sample data estimator

$$\hat{f}_K(\mathbf{x}) = \frac{1}{\mathrm{n}} \sum_{i=1}^{N} \frac{1}{\mathrm{h}_i^d} K\left(\frac{\mathbf{x} - \mathbf{X}_i}{\mathrm{h}_i}\right) \tag{7}$$

based on a spherically symmetric kernel $K(\mathbf{x})$ with bounded support satisfying $K(\mathbf{x}) = c_{K,d} k(\| \mathbf{x} \|^2)$, in which case it suffices to define the function $k(\mathbf{x})$ called the profile of the kernel, only for $x \geq 0$ and $\mathrm{c}_{K,\mathrm{d}}$ is the normalized constant which makes $K(\mathbf{x})$ integrate to one. We define the derivative of the kernel profile as a new function $g(x) = -k'(x)$, and assume that this exists for all $x \geq 0$, except for a finite set of points. Now, if we use a function for profile, the kernel is defined as $G(\mathbf{x}) = c_{Gd} g(\| \mathbf{x} \|^2)$, where $c_{G,d}$ is the corresponding normalization constant. By taking the gradient of (7) the following property can be proven

$$\mathbf{m}_G(\mathbf{x}) = \frac{1}{2} \mathrm{h}^2 c \frac{\nabla \hat{f}_K(\mathbf{x})}{\hat{f}_G(\mathbf{x})} \tag{8}$$

where c is a positive constant and

$$\mathbf{m}_G(\mathbf{x}) = [\Sigma_{i=1}^{\mathrm{N}} \mathbf{x}_i \, \mathrm{g}(\mathbf{v}) / \Sigma_{i=1}^{N} \mathbf{g}(\mathbf{v}^2)] - \mathbf{x}, \mathbf{v} = \left\| \frac{\mathbf{x} - \mathbf{X}_i}{\mathrm{h}} \right\| \tag{9}$$

is called the mean shift vector. The expression (8) shows that at location $\mathbf{x}$ the weighted mean of the data points selected with kernel $G(\mathbf{x})$ is proportional to the normalized density gradient estimate obtained with kernel $K(\mathbf{x})$. The mean shift vector thus points toward the direction of maximum increase in the density. The implication of the mean shift property is that the iterative procedure

$$\mathbf{y}_j = \frac{\Sigma_{i=1}^{\mathrm{N}} g(\mathbf{v}) \cdot \mathbf{X}_i}{\Sigma_{i=1}^{\mathrm{N}} g(\mathbf{v})}, \quad j = 1,2,\ldots \tag{10}$$

This is the weighted mean at $\mathbf{y}_j$ computed with kernel $G(\mathbf{x})$ and $\mathbf{y}_1$ is the center of the initial position of the kernel $\mathbf{x}$. The corresponding sequence of density estimates computed with shadow kernel $K(\mathbf{x})$ is given by $\hat{f}_K(j) = \hat{f}_K(\mathbf{y}_j)$, $j = 1,2,\ldots$. Here, if kernel has a convex and monotonically decreasing profile, two sequences $\{\mathbf{y}_1, \mathbf{y}_2, \ldots\}$

and $\{\hat{f}_K(1), \hat{f}_K(2),\ldots\}$ converge and $\{\hat{f}_K(1), \hat{f}_K(2),\ldots\}$ is monotonically increasing. After that, let us denote by $\mathbf{y}_c$ and $\hat{f}_K^c$ the convergence points of their sequences respectively. Here, we can get two kinds of implications from the convergence result. First, the magnitude of the mean shift vector converges to zero. In fact, the *j*-th mean shift vector is given as

$$\mathbf{m}_G(\mathbf{y}_j) = \mathbf{y}_{j+1} - \mathbf{y}_j, \tag{11}$$

and this is equal to zero at the limit point, $\mathbf{y}_c$. In other words, the gradient of the density estimate computed at $\mathbf{y}_c$ is zero. That is, $\nabla\hat{f}_K(\mathbf{y}_c) = 0$. Hence, $\mathbf{y}_c$ is a stationary point of density estimate, $\hat{f}_K(\mathbf{x})$. Second, since $\{\hat{f}_K(1), \hat{f}_K(2),\ldots\}$ is monotonically increasing, the trajectories of mean shift iterations are attracted by local maximum if they are unique stationary points. That is, once y_j gets sufficiently close to a mode of density estimate, it converges to mode. The theoretical results obtained from the above implications suggest a practical algorithm for detecting modes in images: (a) run the mean shift procedure to find the stationary points of density estimates, (b) prune these points by retaining only the local maximum. Therefore, we can automatically determine the number of modes by adaptive mean shift.

5.2 Text Segmentation using Separated Clustering Method

The number and centroid of modes selected in the subsection 5.1 are used as seed values in K-means clustering. Proposed method of separated K-means clustering is then applied to values in the improved image to segment the text. In our case, two different K-means clustering algorithms are performed because intensity values are linear and hue values are characterized with the cyclic property. When the intensity value falls in the range of $[0.0 \sim 0.4]$, the Euclidean distance to seed values in the same range are computed for clustering. When the intensity value falls in the range of $[0.6 \sim 1.0]$, K-means clustering algorithm is performed by considering the cyclic property of Hue component. In that case, the pixel is clustered into the group with a seed closest to the Hue component. Chi Zhang et al. in [19] show that values near the minimum of (0.6) and the maximum (1.0) are clustered as one mode. Two K-means clustering passes are therefore performed while maintaining both the linear property of intensity values in the range $[0.0 \sim 0.4]$ and the cyclic property of hue values in the range $[0.6 \sim 1.0]$.

6. Experimental Results

The proposed method has been applied for segmenting several color images. Using images are corrupted by various noises. In order to perform our experiment, text

regions are manually captured from text images. Examples of representative images from the dataset and the corresponding results are shown in Figure 5. The first and second image is original image which can typically cause problems when using the natural text images for text segmentation. The second image is synthetic image corrupted in PHOTSHOP tool. The results show that proposed approach can restore the corruption parts in corrupted regions as well as removing noises. Figure 6 compares proposed method with conventional segmentation methods. We used EDISON [Comaniciu et. Al. (1997)] as the other segmentation approach. Our approach has performed better segmentation, potentially improving accuracy and reducing computational complexity of OCR algorithms. And then, we performed the processing time of used approaches and the result on three real images is shown in the table 1. The processing time of our proposed method may not be efficient because our approach method has iteration procedure for restoring damaged regions. However, proposed method shows a superior segmentation through reducing the noise remarkably from corrupted text images.

(a)

(b)

Figure 5. Experimental results by proposed method: (a) corrupted text images, (b) segmented images.

(a) (b) (c)

Figure 6. Examples of experimental results primarily containing achromatic regions: (a) corrupted text images, (b) results by proposed method, (c) EDISION method.

Table 1. Comparison results of two approaches with processing time (sec.)

	Proposed method	EDISON
Image 1 (256x256)	10.5(4times)	5.3
Image 2 (256x256)	7.8(3times)	4.6

7. Conclusions

We have presented a new general method for simultaneously restoring and segmenting text image. We derived a set of methods suitable for tasks in image processing and low-level computer vision. The presented methods are associated with the second order tensor and separated clustering method. Data in given image were considered by chromatic and achromatic features. And, selected data were described as tokens by second order tensor, which can detect the presence of noises such as crack or scrawl in corrupted text image. Vector median method then provides proper values to replace the noise values, which are present on texts. Finally, the restored image is clustered by separated k-means method with an adaptive mean shift finding proper modes. Proposed approach can eliminate various noises well while segment text as one object. The result can contribute to restore damaged region as well as eliminating noises in corrupted text images.

References

Belongie. S. *et. al*.(1998), *Color- and texture-based image segmentation using EM and its application to content-based image retrieval*, Proc. of ICCV, pp. 675-682.

Delignon . Y. et. al. (1997), *Estimation of generalized mixtures and its application in image segmentation*, IEEE Trans. On Image Processing, vol. 6, no. 10, pp. 1364-1376.

Panjwani. D.D. and Healey. G. (1995), *Markov random field models for unsupervised segmentation of textured color image*, IEEE Trans. On PAMI, vol. 17, no. 10, pp. 939-954.

Wang. J.P. (1998), *Stochastic relaxation on partitions with connected components and its application to image segmentation*, IEEE Trans. On PAMI, vol. 20, no. 6, pp. 619-636.

Zhu. S.C. and Yuille. A, *Region competition: unifying snakes, region growing, and Bayes/MDL for multiband image segmentation*, IEEE Trans. On PAMI, vol. 18, no. 9, pp. 884-900.

Shafarenko. L.,Petrou. M., and Kittler. J. (1997), *Automatic watershed segmentation of randomly textured color images*, IEEE Trans. On Image Processing, vol. 6, no. 11, pp. 1530-1544.

Ma. W.Y. and Manjunath. B.S. (1997), *Edge flow: a framework of boundary detection and image segmentation*, Proc. of CVPR, pp. 744-749.

Shi. J. and Malik. J. (1997), *Normalized cuts and image segmentation*, Proc. of CVPR, pp. 731-737.

Borsotti. M., Campadelli. P. and Schettini. R. (1998), *Quantitative evaluation of color image segmentation results*, Pattern Recognition Letters, vol. 19, no. 8, pp. 741-748.

Comaniciu. D. and Meer. P. (1997), *Robust analysis of feature spaces: color image segmentation*, Proc. of IEEE Conf. on Computer Vision and Pattern Recognition, pp. 750-755.

Zhong. Y., Karu. K., Jain. A.K. (1995), *Locating text in complex color images*, Pattern Recognition, vol. 28, pp. 1523-1536.

Jain. K., Yu. B. (1998), *Automatic Text location in images and video frames*, Pattern Recognition, vol. 31, pp. 2055-2076.

Haritaoglu. I. (2001), *Scene text extraction and translation for handheld devices*, IEEE Conference on Computer Vision and Pattern Recognition, pp.408-413.

Ye. Q., Gao. W., Huang. Q. (2004), *Automatic text segmentation from complex background*, IEEE Int. Conference on Image Processing, vol. 5, pp. 2905-2908.

Lucas. S.M., Panaretos. A., Sosa. L, Tang. A., Wong. S. and Young. R. (2003), *ICDAR 2003 robust reading competitions*, IEEE Int. Conference on Document Analysis and Recognition, pp.682-687.

Medioni. G., Lee. M.S., and Tang. C.K. (2000), *A Computational Framework for Segmentation and Grouping*, Elsevier.

Lee. M.S., Medioni. G. (1999), *Grouping .,-,→, ,into regions, curves, and junctions*, Computer Vision and Image Understanding, vol. 76, no. 1, pp.54-69.

Shen. Y. and Barner. K.E. (2004), *Fuzzy vector median-based surface smoothing*, IEEE Trans. On Visualization and Computer Garphics, vol. 10, no. 3, pp. 252-265.

Zhang. C., Wang. P. (2000), *A new method of color image segmentation based on intensity and hue clustering*, IEEE Int. Conference on Pattern Recognition, vol.3, pp. 3617-3621.