OPTOELECTRONICS LETTERS Vol.11 No.1, 1 January 2015

An adaptive tensor voting algorithm combined with texture spectrum*

WANG Gang (王刚)[1,2]**, **SU Qing-tang** (苏庆堂)[1], **LÜ Gao-huan** (吕高焕)[1], **ZHANG Xiao-feng** (张小峰)[1], **LIU Yu-huan** (刘玉环)[1], **and HE An-zhi** (贺安之)[2]

1. School of Information and Electrical Engineering, Ludong University, Yantai 264025, China

2. School of Science, Nanjing University of Science & Technology, Nanjing 210094, China

(Received 28 September 2014)

An adaptive tensor voting algorithm combined with texture spectrum is proposed. The image texture spectrum is used to get the adaptive scale parameter of voting field. Then the texture information modifies both the attenuation coefficient and the attenuation field so that we can use this algorithm to create more significant and correct structures in the original image according to the human visual perception. At the same time, the proposed method can improve the edge extraction quality, which includes decreasing the flocculent region efficiently and making image clear. In the experiment for extracting pavement cracks, the original pavement image is processed by the proposed method which is combined with the significant curve feature threshold procedure, and the resulted image displays the faint crack signals submerged in the complicated background efficiently and clearly.

Document code: A **Article ID:** 1673-1905(2015)01-0073-4

DOI 10.1007/s11801-015-4174-3

Tensor voting algorithm is becoming popular as a method of perceptual reconstruction in the fields of computer vision, machine learning and pattern recognition in recent years. By extrapolating the framework of significant structures, such as breakpoints, endpoints, points of intersection, curves, areas and curved surfaces, from discrete data[1], or by detecting these significant traits from an image constructed by point, curve and surface, the tensor voting algorithm is able to perform the tasks of restoration or estimation[2], therefore it has been applied successfully in segmentation and grouping[3], image repairing and correction[4], curve or curved surface reconstruction and motion estimation, etc[5,6].

We know that the only human-determined parameter in the tensor voting algorithm is called as scale parameter (voting field)[7-10], which can influence the detected structure information, i.e., too small or too large scale parameters will lead to the unstable and uncorrect results. To make the tensor voting algorithm adaptive, we need to use some characteristics of the image to set the scale parameter appropriately, rather than depending on human's subjective judgment or experience judgment. Although it is not likely to guarantee a reconstructed structure 100% the same as the original one, the deviation control and the precision improvement are still badly needed when processing the images, such as pavement image, brain computed tomography (CT) image and city remote sensing satellite image. Through doing experiment, we find that some flocculent polluted detail areas of the resulted image still exist when we set the voting field as a stationary scale. By analyzing, we think this is the result of over-voting and wrong voting, which means that the actual influence of tickets is more than what it should be or the ticket is counted wrongly. As this is the mistake of a wrongly-estimated image, we intend to correct it with the original image itself, and the best is to use the texture spectrum to solve the problems above. We know that the texture spectrum stands for certain geometrical structure in an image, and this geometrical structure can be explained to a mensuration of connecting some information, such as edge position, direction and intension. From this point, the texture is actually the part of significant structure. Nowadays, a formal academic definition for texture is not possible, but the truths that the texture reflects the surface characteristics of image, such as smoothness, roughness and granularity, and people's visual perception about surface characteristics is the result of judging texture information have been generally accepted[11-14]. Even in many cases, we regard that the image texture structure as detail information is worth being protected and this texture information is the significant structure in the image. Therefore, we find the solution to improve the tensor voting algorithm, which is the idea of using texture spectrum.

* This work has been supported by the National Natural Science Foundation of China (No.61471185), the Joint Special Fund of Shandong Province Natural Science Foundation (No.ZR2013FL008), and the Project of Shandong Province Higher Educational Science and Technology Program (No.J14LN20).

** E-mail: Happy_wg@163.com

Considering texture spectrum is a localization structure, we choose an $n\times n$ matrix centered by a pixel as neighborhood area, and take the grey variance value around this pixel as a texture primitive V_i. Define the texture primitive by grey value changes as

$$V_i=\begin{cases}0 & \left|I_i-I_c\right|\le Th_1\\ 1 & Th_1\le\left|I_i-I_c\right|\le Th_2\\ 2 & \text{Otherwise}\end{cases},\qquad(1)$$

where I_i means the grey values of the regional pixels, I_c means the grey value of center pixel, and Th_1 and Th_2 are threshold constants which are decided by the distribution of the grey level. Define the texture primitive parameter as

$$T_c=\frac{1}{n^2}\sum_{i=1}^{n^2}V_i\ .\qquad(2)$$

For getting the texture spectrum of an area, the average of all texture primitive parameters in the local zone is calculated as

$$T=\frac{1}{ab}\sum_{i=1}^{a}\sum_{j=1}^{b}T_c(i,j)\ ,\qquad(3)$$

where a and b are the sizes of the voting field.

From the discussion above, we know that the texture spectrum can describe the roughness of a certain area. Big spectrum means that the grey variance is violent and the texture information is rich. On the contrary, small spectrum means a smooth grey change and the less texture information. Too much grey variance information in the voting field may bring the voting deviation or the wrong voting result. We can use texture spectrum to adjust the voting field and attenuation function adaptively and improve the voting quality.

Extract the edge from the input image to get sparse input data by using Canny operator. We think that a sparse input image has no curve structure, so all points are coded to ball tensors as

$$\boldsymbol{B}=[\boldsymbol{e}_1\ \boldsymbol{e}_2]\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}\boldsymbol{e}_1^{\mathrm{T}}\\ \boldsymbol{e}_2^{\mathrm{T}}\end{bmatrix}.\qquad(4)$$

Stick tensor is constructed after ball tensor voting, which is defined as

$$\boldsymbol{S}=[\boldsymbol{e}_1\ \boldsymbol{e}_2]\begin{bmatrix}\lambda_1 & 0\\ 0 & \lambda_2\end{bmatrix}\begin{bmatrix}\boldsymbol{e}_1^{\mathrm{T}}\\ \boldsymbol{e}_2^{\mathrm{T}}\end{bmatrix},\qquad(5)$$

where λ_1 and λ_2 are the eigenvalues of the tensors which indicate the semi-major and semi-minor axes, respectively, and $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$ are the corresponding eigenvectors which represent the directions of two axes. In this paper, we use a sparse voting method, in which a feature structure only votes to another feature structure, and it is different from the dense voting method which means that a feature structure votes to point in every position. Therefore, the first voting happens among non-zero ball tensors in the feature set extracted before. The attenuation function DF is defined as

$$DF(s,k,\sigma,T(i,j))=\mathrm{e}^{-\frac{s^2+c_1k^2+c_2T(i,j)}{\sigma^2}}\ ,\qquad(6)$$

where $s=\theta l/\sin\theta$ is arc length from the voter A to the receiver B, $k=2\sin\theta/l$ is the texture adjustment coefficient, θ is the angle between line AB and the tangential line at the position of A, $T(i,j)$ is texture spectrum of the voting field, c_1 is the curvature degeneration control coefficient and $c_1=\frac{-16(\log 0.1)\times(\sigma-1)}{\pi^2}$, $c_2=100T_{\max}$ is texture spectrum coefficient, and $T_{\max}$ is the maximum value in the texture spectrum. When $\theta>\pi/4$, A stops voting, and σ is the voting field parameter which is defined as

$$\sigma=\alpha ab\sum_{i=1}^{a}\sum_{j=1}^{b}\frac{\sin(T(i,j))}{T(i,j)}\ ,\qquad(7)$$

where α is the adjustment coefficient.

In the tensor voting process, ball tensor voting is more complicated because it doesn't have a certain developing direction. However, its ticket is very important, so we define its ticket as

$$\boldsymbol{B}(\theta)\approx\sum_{i=1}^{k}\boldsymbol{R}_\theta^{-1}\boldsymbol{C}(\boldsymbol{R}_\theta)\boldsymbol{R}_\theta^{-\mathrm{T}}\ ,\qquad(8)$$

where $\boldsymbol{R}_\theta$ is rotation matrix that makes voter tensor $\boldsymbol{C}$ rotate to eigenvector direction corresponding to its maximum eigenvalue, and θ is the rotation angle (set positive x axis as reference). In actual operation, we divide the circle into N shares and calculate the unit stick tensor along directions of $2\pi k/N$ (k=0,1,2,⋯, N). The receiver can get a final ticket with definite direction and strength only by calculating the vector product of all non-zero tickets.

The ticket of a stick tensor is more clear. Define the ticket tensor at a receiver position as

$$\boldsymbol{S}(l,\theta,\sigma,T(i,j))=DF(s,k,\sigma,T(i,j))\begin{bmatrix}-\sin(2\theta)\\ \cos(2\theta)\end{bmatrix}[-\sin(2\theta)\ \cos(2\theta)]\ .\qquad(9)$$

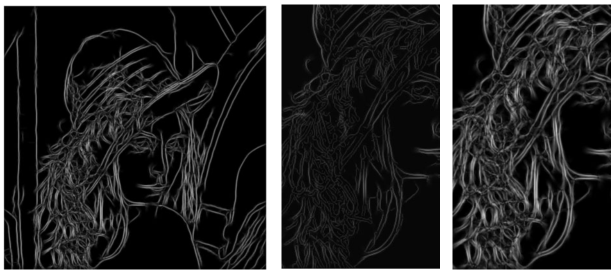
In the whole tensor voting process, the texture spectrum works for the voting control which includes calculating the voting field and the attenuation function. As we said before, the richer texture information equals a bigger texture spectrum value, and a smaller voting field can reduce the richer information. If the texture information is sparse, the spectrum value will be smaller, and a larger voting field is created to increase the lower information.

The tensor decompositions of curve significant image and intersection significant image are got. Through oval representation of a tensor, we can get them conveniently as follows: for curve significant image, $W=\lambda_1/\lambda_2>1$ and $\boldsymbol{w}=\boldsymbol{e}_1$, and for intersection significant image, $W=\lambda_1=\lambda_2$ and $\boldsymbol{w}$ has no certain direction. From significant images, we can know the possibility of a point belonging to a curve or an intersection. If a point is within a curve, its coordinate and direction information $\boldsymbol{w}$ determine its position in this curve, and it contributes to the development of the curve. When we extract the curve structure, we only require those points which have large possibility of belonging to a curve in a zone, which are also the points with the maximum W. Intersection is the cross of two or even more curves. Although they have $\lambda_1=\lambda_2$, these eigenvalues are bigger than those of their neighbors. So we can confirm the intersections if we extract the points with the maximum W in an intersection significant image. After determining curves and intersections, the image characteristics are constructed.

We use the 512×512 Lena image with 256 grey levels. The thresholds of Th_1 and Th_2 are chosen as 64 and 128, respectively, and the texture adjustment coefficient k is 0.1. Fig.1(a) is the original image, and Fig.1(b) shows the texture spectrum from the Lena original image. We can see that Fig.1(b) is able to well describe the gray value changes in the original image. Fig.1(c) is the image processed by classic algorithm in which the stationary voting field parameter σ is 18.25 without the texture spectrum. Apparently, Fig.1(c) almost loses all useful information and becomes unreadable, and the image is badly polluted by strong flocculent regions, which results in a serious distortion. Fig.1(d) is the image processed by tensor voting algorithm combined with texture spectrum in adaptive voting field, and the mean voting field parameter σ is 8.336 3. Fig.1(e) is the result of classic algorithm without texture spectrum, and the stationary voting field parameter σ is 8.336 6. The differences are that we use the adaptive voting field and texture spectrum in Fig.1(d), while in Fig.1(c) and (e) we use the stationary global mean voting field. Fig.1(e) looks similar to Fig.1(d), but in detail areas, they are different. The enlarged images of the regional parts in Fig.(d) and (e) for comparison are shown in Fig.1(f) and (g), respectively.

(a) (b)

(c) (d)



(e) (f) (g)

Fig.1 (a) The original image; (b) The texture spectrum from the original image; Images processed by (c) classic algorithm with σ=18.25, (d) the algorithm using two-threshold texture spectrum and adaptive voting field with σ=8.336 3, and (e) classic algorithm with σ=8.336 6; (f) The enlarged parts from (d); (g) The enlarged parts from (e)

Under the premise of keeping tensor voting with high effect, adding texture information makes Fig.1(f) save more correct detail information, which can be proved in hair terminal. In Fig.1(f), the hair terminals present divergence while they are closed in Fig.1(g). Fig.1(g) can be interpreted as the phenomenon of over-voting. The experimental results show that the participation of texture spectrum affects the preservation of image details, which can also affect both the voting field and the attenuation functions.

In this paper, we apply our tensor voting algorithm to extract the pavement cracks in a set of crack images in Maqun, Huning freeway. The size of each image is 256×256, and 1 mm^2 area is represented by 16×16 pixels. The corresponding results are shown in Fig.2. Fig.2(a) and (b) are the original pavement images which contain the horizontal and vertical cracks, respectively. Owing to the background noise, Fig.2(c) and (d) are the token images containing some scattering noise, whose gray values are equal to the cracks in Fig.2(a) and (b), respectively. Based on the line feature of the cracks in Fig.2(a) and (b), we constrain the curve significant feature W with a threshold. That is to say only if the point in the token image has larger directional tendency, we will add the point within the curve. Otherwise, we will take the point as scattering noise. So the characteristic discriminant criterion can be modified as

$$W=\lambda_1-\lambda_2>\lambda_2, \quad \boldsymbol{w}=\boldsymbol{e}_1 . \tag{10}$$

Because the distribution of gray values in Fig.2(a) and (b) is about from 40 to 120, the thresholds Th_1 and Th_2 are choosen as 6 and 60, respectively. In Fig.2(e) and (f), the texture spectrum is represented by the gray value, i.e., the higher the gray value, the larger the texture spectrum. So we can see that the regions containing cracks have larger texture spectrum. Then a smaller voting field should be choosen in these regions in order to avoid over-voting and wrong voting. With the texture spectrum shown in Fig.2(e) and (f), the resulted images by using our proposed tensor voting method are shown in Fig.2(g) and (h), and the mean voting field parameters are 11.32 and 9.43, respectively. In the resulted images, the broken cracks are connected, which meets the perceived effect of the human brain. At the same time, the scattering noise is effectively suppressed by the significant curve feature constraint.

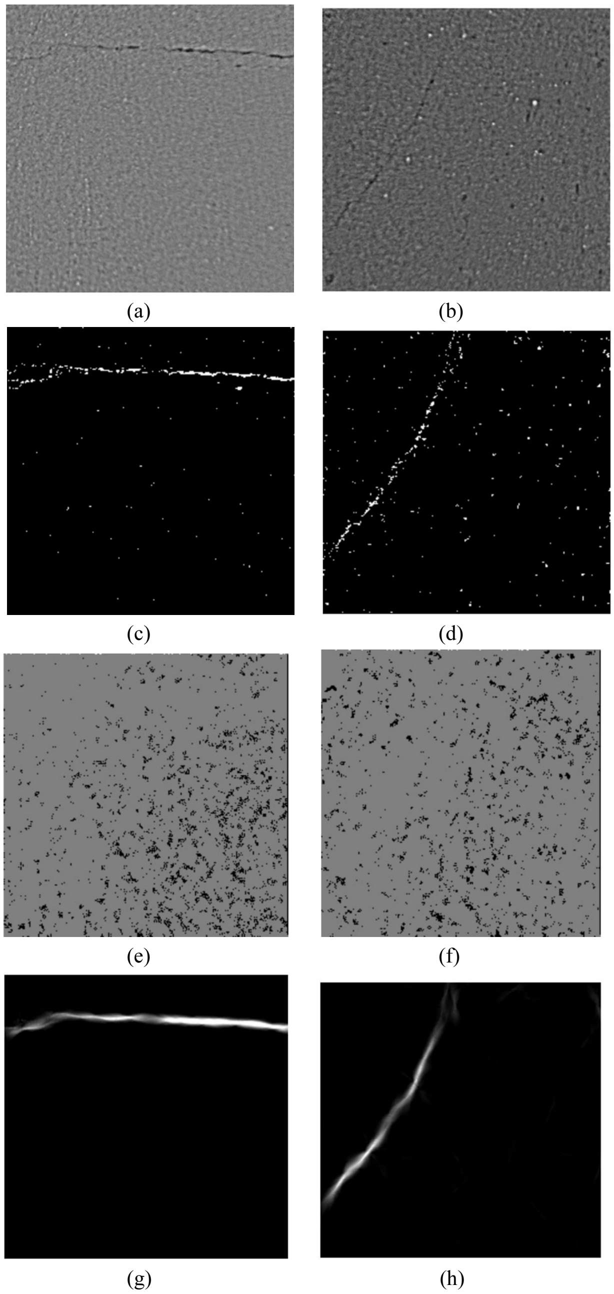


Fig.2 The original pavement images containing (a) horizontal and (b) vertical cracks; (c) and (d) The token crack images from (a) and (b); (e) and (f) The texture spectra computed from (a) and (b); (g) and (h) The resulted images from (a) and (b) by using our proposed tensor voting method

In this paper, we introduce the texture spectrum to tensor voting for improving the visual effect. Using the texture spectrum can adaptively adjust the voting field and the attenuation functions to avoid the phenomenon that too much texture information in the local region disturbs the voting effect. The introduction of texture spectrum is successful for extracting the integral edge in the original image under the complicated background. In the pavement crack extraction experiment, the proposed method can effectively extract the complete crack information and suppress the scattering noise by combining with the significant curve feature threshold. However, the biggest flaw is that the image brightness decreases, so in the future, we plan to design a matching image enhancement algorithm.

References

[1] W. S. Tong, C. K. Tang and G. Medioni, First Order Tensor Voting, and Application to 3-D Scale Analysis, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 175 (2001).

[2] P. Mordohai and G. Medioni, Journal of Machine Learning Research **11**, 411 (2010).

[3] Duan Fen-fang, Shao Feng, Jiang Gang-yi, Yu Mei and Li Fu-cui, Journal of Optoelectronics·Laser **25**, 192 (2014). (in Chinese)

[4] Li Wei-hong, Chen Long and Gong Wei-guo, Journal of Optoelectronics·Laser **25**, 558 (2014). (in Chinese)

[5] M. Kulkarni and A. N. Rajagopalan, Tensor Voting Based Foreground Object Extraction, National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics, 86 (2011).

[6] R. Hariharan and A. N. Rajagopalan, IEEE Transactions on Image Processing **21**, 3323 (2012).

[7] A. Mukherjee, B. Jenkins, C. Fang, R. J. Radke, G. Banker and B. Roysam, Medical Image Analysis **15**, 354 (2011).

[8] Jia-Ya Jia and Chi-Keung Tang, IEEE Transactions on Pattern Analysis and Machine Intelligence **27**, 36 (2005).

[9] R. Hariharan and A. N. Rajagopalan, IEEE Transactions on Image Processing **21**, 3323 (2012).

[10] M. K. Park, S. J. Lee and K. H. Lee, Graphical Models **74**, 197 (2012).

[11] R. Lopes, P. Dubois, I. Bhouri, M. H. Bedoui, S. Maouche and N. Betrouni, Pattern Recognition **44**, 1690 (2011).

[12] Zhenhua Guo and Zhang D., IEEE Transactions on Image Processing **19**, 1657 (2010).

[13] G. H. Liu, Z. Y. Li, L. Zhang and Y. Xu, Pattern Recognition **44**, 2132 (2011).

[14] M. Sezgin and B. Sankur, Journal of Electronic Imaging **13**, 146 (2004).