Extracting Lines in Noisy Image Using Directional Information

Jun Zhou and Walter F. Bischof
Department of Computing Science
University of Alberta
Edmonton, Alberta, Canada T6G 2E8
{jzhou,wfb}@cs.ualberta.ca

Arturo Sanchez-Azofeifa
Department of Earth & Atmospheric Sciences
University of Alberta
Edmonton, Alberta, Canada T6G 2E3
arturo.sanchez@ualberta.ca

Abstract

Detection of lines in noisy image is not easy. When using Hough transform, multiple false peaks may be generated from collinear noisy edge points obtained from edge detection, and in turn create false line segments. This paper introduced a method based on directional information to overcome the problem. Directional information is generated by the anisotropic Gaussian filter and Gabor filter, and is used to guide the smoothing in edge detection and peak selection in Hough transform. The experimental results shows the effectiveness of this method.

1. Introduction

Line detection is a fundamental task in computer vision. Hough transform is the most well-known method to detect lines [8, 9]. Normally, Hough transform starts from edge detection based on the local differential property of the image. Then, the edge map is transformed from image space into parameter space. Collinear points in the image spaces correspond to peaks in the parameter space. Thus, we can restore candidate points for line segments in the image space from these peaks. By analyzing and grouping the candidate points, we can obtain the final line segments.

The presentation of noises may affect the line detection in the Hough transform. First, the edge detection is difficult. In noisy images, edges may not be accurately detected and localized. For example, Canny edge detector [1], which often provides the best performance in edge detection, may not find a suitable scale to keep the detail of the edge information while efficiently remove the noises. Second, the noisy edges may generate noisy peaks in the parameter space. These noisy peaks may be mixed with the butterfly pattern of transformed line segments, or generate individual false peaks. Thus, it is difficult to distinguish true peaks from false peaks, and in turn, prohibits correct grouping of candidate points into line segments.

Methods have been proposed to solve this problem, for example, anisotropic filtering in edge detection [10, 6], peak analysis in Hough transformation [11, 5], and grouping of directional features [2].

We propose a new method to effectively detect line segments in noisy images using directional information. First, anisotropic Gaussian filtering is performed in edge detection step. Then orientations extracted from Gabor filter is used in peak analysis. This method has greatly reduce the influence of noises in line extraction.

2. Anisotropic filtering for edge detection

The purpose of anisotropic filtering is to accurately detect edges or lines. In isotropic filtering, noises are smoothed in the same way in all directions. To reduce the side-effect of noises aside the edge or line, one might want to take advantage of the directional property of the linear features. It requires fine tune the filter in the direction of the lines.

2.1. Anisotropic filtering

A fast anisotropic Gaussian filter has been proposed by Geusebroek and Smeulders [6]. The oriented filter in two dimensions is given by the convolution of two Gaussian filters:

$$g(u) = \frac{1}{\sqrt{2\pi}\sigma_u} \exp(-\frac{u^2}{2\sigma_u^2}) \tag{1}$$

and

$$g(v) = \frac{1}{\sqrt{2\pi}\sigma_v} \exp(-\frac{v^2}{2\sigma_v^2}) \tag{2}$$

where $u = x\cos\theta + y\sin\theta$ and $v = -x\sin\theta + y\cos\theta$. Here, θ is the orientation of the anisotropic Gaussian filter, x and y are the cartesian coordinates of the image pixels.

To facilitate the computation, the filter is transformed into image coordinates along the x-direction and t-direction, as shown in figure 1.

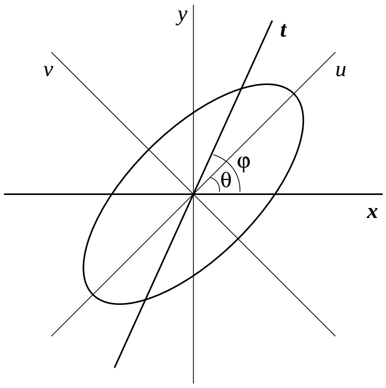


Figure 1. Anisotropic Gaussian filter.

The transformed filters are:

$$g(x) = \frac{1}{\sqrt{2\pi}\sigma_x} \exp(-\frac{x^2}{2\sigma_x^2}) \tag{3}$$

and

$$g(t) = \frac{1}{\sqrt{2\pi}\sigma_t} \exp(-\frac{t^2}{2\sigma_t^2}) \tag{4}$$

where $t = x\cos\phi + y\sin\phi$. The relationship between σ_x, σ_ϕ and σ_u, σ_v, θ is:

$$\sigma_x = \frac{\sigma_u \sigma_v}{\sqrt{\sigma_v^2 \cos^2\theta + \sigma_u^2 \sin^2\theta}} \tag{5}$$

$$\sigma_\phi = \frac{1}{\sin\phi}\sqrt{\sigma_v^2 \cos^2\theta + \sigma_u^2 \sin^2\theta} \tag{6}$$

$$\tan\phi = \frac{\sigma_v^2 \cos^2\theta + \sigma_u^2 \sin^2\theta}{(\sigma_u^2 - \sigma_v^2)\cos\theta\sin\theta} \tag{7}$$

In this way, a 2-D anisotropic Gaussian filter can be implemented by convolving two 1-D Gaussians in the x-direction and ϕ-direction.

2.2. Anisotropic edge detection

The edge detection is performed by Canny edge detector to generate a binary edge map. However, in the Gaussian smoothing step, the traditional isotropic smoothing is replaced by anisotropic smoothing introduced above. To choose the angle in smoothing each pixel, the original image is filtered at 16 orientations. The filtered image takes the maximum response for each pixel over all filters.

3. Line detection using Hough transform

Hough transform is performed on the edge map to transform the $x - y$ coordinates of edge points into $\rho - \theta$ space. We used the normal parameterization proposed by Duda and Hart [4] that a line in an image is given by:

$$\rho = x\cos\theta + y\sin\theta \tag{8}$$

where ρ is the distance of the line to the origin, and θ is the angle between the normal of the line and the $x-$axis, and (x, y) are any point on the line.

In the implementation of the Hough transform, the $\rho - \theta$ space is normally quantized into cells. Assume that a line is mapped to a peak at point (ρ_0, θ_0), the collinear points on the line generate a pattern in the parameter space, and is often mentioned as a *butterfly* [11]. The analysis on the butterfly is supposed to be restricted in a small neighborhood window $(-w_\rho/2, w_\rho/2) \times (-w_\theta/2, w_\theta/2)$ around the peak (ρ_0, θ_0). The peaks in the neighborhood window can be grouped and removed in sequential peak selection [12].

One problem of this peak selection and grouping strategy is that noisy edge points may generate noisy peaks in the parameter space. One example is shown in figure 2 and 3. Figure 3(b) shows the top 20 lines generated from Hough transform peaks (figure 3(a)). Due to the noisy edge points from edge detection (in figure 2(b)), most extracted line are false positive result. Although we can increase the scale of the Gaussian filter to further eliminate the noises in the edge detection, it usually can not remove them completely, and as a side-effect, may also remove correct line points. The noisy peaks can not be excluded without further analysis in the peak selection.

4. Peak selection based on directional information

4.1. Estimate local orientation

By analyzing figure 2(b), we find out that the orientation of the line segment is helpful in peak selection. This is based on the fact that in peaks mapped from lines, the collinear edge points that contribute to the peaks tend to have the same local orientation, while the noisy edge points don't. Also, most of the line segments have parallel edges and similar width, which can be considered as parallel ridges and valleys with similar frequency. Thus, we can remove the noisy edges by tuning a bandpass filter to the appropriate frequency and orientation.

Gabor filter is such a bandpass filter [3]. A 2-D Gabor filter is a Gaussian envelope modulated by a complex sinusoidal carrier, and can be defined as:

$$g(x, y) = h(x, y)s(x, y) \tag{9}$$

where $h(x, y)$ is a Gaussian envelope:

$$h(x, y) = \exp(-(\frac{(x - x_0)^2}{2\sigma_x^2} + \frac{(y - y_0)^2}{2\sigma_y^2})) \tag{10}$$

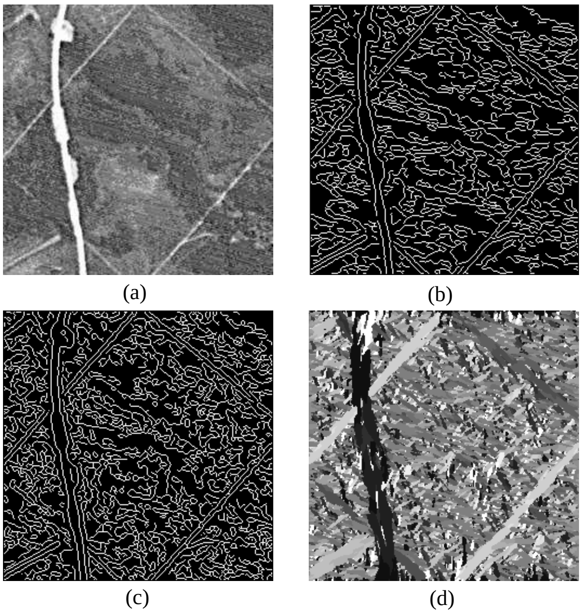


Figure 2. Anisotropic Gaussian filtering, edge detection and orientation map. (a) Original image, (b) anisotropic edge detection, (c) isotropic edge detection, (d) orientation map from Gabor filtering.

and $s(x, y)$ is a sinusoidal carrier:

$$s(x, y) = \exp(-2\pi i(u_0(x - x_0) + v_0(y - y_0))) \quad (11)$$

Here, (x_0, y_0) is the peak of the function, (u_0, v_0) defines the spatial frequency of the sinusoidal carrier, and (σ_x, σ_y) defines the scale of the Gaussian envelope along the x-axis and y-axis respectively.

Based on the above definition, an even-symmetric Gabor filter is given by [7]:

$$g_t(x, y) = \exp(-(\frac{x^2}{2\sigma_x^2} + \frac{y^2}{2\sigma_y^2}))\cos(\frac{2\pi x}{t}) \quad (12)$$

where t defines the wavelength of the Gabor filter.

As in Section 2.1 defining the oriented filter, we may change the orientation of Gabor filter to angle θ, so that

$$g_{t,\theta}(x, y) = g_t(x', y') \quad (13)$$

where $x' = x\cos\theta + y\sin\theta$ and $y' = -x\sin\theta + y\cos\theta$.

To implement this even-symmetric Gabor filter, we need to determine parameter t, σ_x, σ_y, and θ. t is a parameter related to the frequency of the edges, and is set to 5 empirically. σ_x, σ_y are both set to 3. To determine the local orientation, we set θ to 16 directions between $[0, pi)$, uniformly

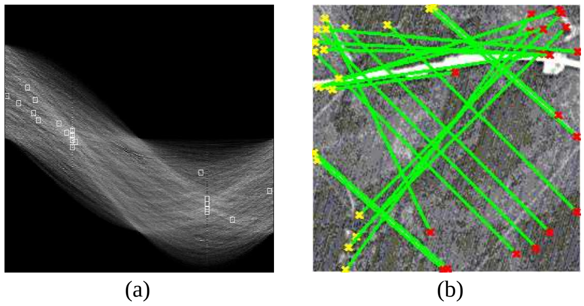


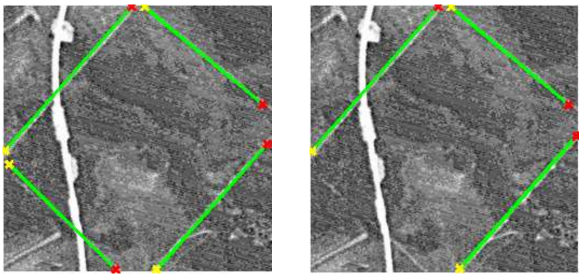
Figure 3. Hough transform. (a) Result of Hough transform on figure 2(b). The white squares are the top 20 peaks. (b) Restored line segments correspond to the top 20 peaks.

distributed. The edge map was filtered iteratively on these directions. The final filtered image takes the maximum response for each pixel over all directions. An example of the orientation map of Figure 2(a) extracted is shown in Figure 2(d). The orientations are displayed in different greylevels. After the filtering, most of the linear features in the image are preserved.

4.2. Peak selection and line grouping

The peak selection is a classification procedure by a decision tree. Each node in the decision tree classifies the lines mapped from a peak in the parameter space on certain feature. These features include the direction of edge points, the distances between edge points, the length of each line segment, and the average length of the line segments in a line.

The classification goes through the following steps. First, the peaks in the parameter space are ranked according to their intensity. The highest peak is selected and collinear edge points contribute to the peak are restored through inverse Hough transform. The average deviation of the directions at these points to the directions of corresponding points in the orientation map is calculated. If the deviation is larger than threshold $t1$, the peak, along with peaks in the neighborhood window are removed from the parameter space. This steps enables the peak suppression and removal of most line segments composed of noisy edge points. The second step is the point grouping. When the orientation condition is satisfied, points closer than threshold $t2$ in distance are grouped together to form line segments. The third steps exclude short noisy line segments. If the line segment is shorter than threshold $t3$, the line segment is removed. Finally, the average length of the line segment in a line is calculated. If it is smaller than threshold $t4$, the line is re-



(a) (b)

Figure 4. Line detection results on figure 2(a). (a) with anisotropic filtering, (b) with isotropic filtering.

moved. This final step is to remove the noisy lines that have not been successfully eliminated in the previous steps because such lines tend to be composed of short line segments.

5. Results and discussion

The image used in the experiment is cropped from an aerial photo of 5 meter resolution. The size is $250 * 250$. The image shows straight trails (the thin lines) goes across the forest that forms the noisy background. The purpose of the detection is to find the four thin line segments that form a rectangle.

The system went through all the steps introduced in the previous sections. The standard deviations of the anisotropic Gaussian filter are set to $\sigma_v = 1$ and $\sigma_u = 3\sigma_v$. The parameters of the Hough transform are set to $\Delta\rho = 1$ and $\Delta\theta = \pi/180$. The thresholds in the peak selection and line grouping are set to $t_1 = \pi/9$, $t_2 = 20$, $t_3 = 30$ and $t_4 = 40$, respectively.

The final result of the line detection is shown in figure 4(a). The starting and ending points of the lines are marked by crosses. The results show that all these lines are detected successfully. Figure 4(b) shows the result when isotropic filtering is used instead of anisotropic filtering in the edge detection step. One line is missing in this case. This is due to the high deviation of line points direction to the direction of the corresponding points in the orientation map. When noise is presented in the image, isotropic filtering can not completely preserve the directional information on edges and lines. However, anisotropic filtering method allows high spatial and angular accuracy in the computation. Comparing figure 2(b) and 2(c) with 3(a), we find that the orientation extracted from Gabor filter matches better to the edges directions from anisotropic filtering than those from isotropic filtering.

6. Conclusion

We have presented a line detection method based on directional information and Hough transform. The directional information is extracted and utilized in two stages. First, anisotropic filtering is used in the edge detection step. Each pixel in the image is smoothed according to their local directions extracted from an anisotropic Gaussian filter. Second, after the Hough transform, the peak selection and line segments grouping is guided by the directional information extracted from Gabor filter. This method can successfully extract lines in noisy images.

References

[1] J. Canny. A computational approach to edge detection. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.

[2] M. Carreira, M. Mirmehdi, B. Thomas, and J. Haddon. Grouping of directional features using an extended Hough transform. In *Proceedings of the 15th International Conference on Pattern Recognition*, volume 3, pages 990 – 993, Barcelona, Spain, September 2000.

[3] J. Daugman. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *Journal of the Optical Society of America*, 2(7):1160–1169, 1985.

[4] R. Duda and P. Hart. Use of the Hough transformation to detect lines and curves in pictures. *Communications of the ACM*, 15(1):11–15, 1972.

[5] Y. Furukawa and Y. Shinagawa. Accurate and robust line segment extraction by analyzing distribution around peaks in Hough space. *Computer Vision and Image Understanding*, 92(1):1–25, 2003.

[6] J. Geusebroek, A. Smeulders, and J. Weijer. Fast anisotrpic Gauss filtering. *IEEE Transactions on Image Processing*, 12(8):938–943, 2003.

[7] L. Hong, Y. Wan, and A. Jain. Fingerprint image enhancement: algorithm and performance evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8):777–789, 1998.

[8] J. Illingworth and J. Kittler. A survey of the Hough transform. *Computer Vision, Graphics, and Image Processing*, 44(1):87–116, 1988.

[9] V. Leavers. Survey: Which Hough transform? *Computer Vision, Graphics, and Image Processing*, 58(2):250–264, 1993.

[10] A. Paplinski. Directional filtering in edge detection. *IEEE Transactions on Image Processing*, 7(4):611–615, 1998.

[11] T. Veen and F. Groen. Discretization errors in the Hough transform. *Pattern Recognition*, 14(1-6):137–145, 1981.

[12] X. Zhang and H. Burkhardt. Grouping edge points into line segments by sequential Hough transformation. In *Proceedings of the 15th International Conference on Pattern Recognition*, volume 3, pages 672–675, Barcelona, Spain, September 2000.