14th European Signal Processing Conference (EUSIPCO 2006), Florence, Italy, September 4-8, 2006, copyright by EURASIP

HISTOGRAM-BASED ORIENTATION ANALYSIS FOR JUNCTIONS

Jan Hendrik Moltz[1], *Ingo Stuke*[1], *and Til Aach*[2]

[1]Institute for Signal Processing
University of Lübeck
Ratzeburger Allee 160, 23538 Lübeck, Germany
moltz@informatik.uni-luebeck.de
stuke@isip.uni-luebeck.de

[2]Institute of Imaging and Computer Vision
RWTH Aachen University
52056 Aachen, Germany
til.aach@lfb.rwth-aachen.de

ABSTRACT

This article presents an algorithm for multiple orientation estimation at junctions which can be used as a first step towards a complete description of the junction structure. The algorithm uses the structure tensor approach to determine the orientations of the edges or lines that meet at a junction and then extracts the principal orientations from a histogram of the orientation angles in a circular region around the junction. In contrast to previous solutions it uses only first-order derivatives and is suited for junctions with an arbitrary number of orientations without increasing the runtime.

1. INTRODUCTION

In many image processing applications, the localization and description of junctions is an important subtask. Junctions are intrinsically two-dimensional image features [9] which means that they provide most of the information contained in an image while features of intrinsic dimension one (edges or lines) and zero (flat regions) appear more frequently but are redundant. In practice it is important that intrinsically two-dimensional features are the only ones that do not suffer from the aperture problem and can be recognized within a small window. This property makes them suitable for matching problems where corresponding points are to be identified in a number of images which is an essential task in motion estimation, object tracking, stereo vision, registration, and object recognition.

Junctions are image regions where several oriented structures like edges or lines meet. After a junction has been detected, an important step towards a complete description of its structure is to determine the orientations of the edges and lines. This constitutes a problem of multiple orientation estimation. Solutions to this problem can be based on well-known algorithms that determine single orientations as they appear at edges or in simple oriented patterns. In the next section, the concept of orientation will be introduced formally and a widely used approach for single orientation estimation using the so called structure tensor will be reviewed.

2. SINGLE ORIENTATION ESTIMATION

An image is called ideally oriented if there is an orientation along which the intensities do not change within a small region. This is rarely the case in practice, but it is still possible to determine the orientation with the least intensity variation which can be a meaningful measure for describing image structures like edges or lines and for detecting their intersection points which appear as corners and junctions in the image. Orientations have to be distinguished from directions: While the latter are given in the range $]-180°, 180°]$, the former only have a range of $]-90°, 90°]$ and thus a periodicity of $180°$.

A formal definition of the notion of orientation can only be given for continuous image functions. An image $I : \mathbb{R}^2 \to \mathbb{R}$ is ideally oriented in a region $\Omega \subseteq \mathbb{R}^2$ if there is a unit vector $\mathbf{v} = (\cos\theta, \sin\theta)^T$ satisfying

$$I(\mathbf{x}) = I(\mathbf{x} + k\mathbf{v}) \quad \forall\, \mathbf{x} \in \Omega, k \in \mathbb{R} \quad \text{s.t. } \mathbf{x} + k\mathbf{v} \in \Omega.$$

This is equivalent to

$$\frac{\partial I(\mathbf{x})}{\partial \mathbf{v}} = 0 \quad \forall\, \mathbf{x} \in \Omega \tag{1}$$

where the left hand term denotes the directional derivative of I along $\mathbf{v}$ which can also be written as $\mathbf{v}^T \nabla I(\mathbf{x})$.

If there is no ideal orientation a minimization problem can be derived from (1)

$$\int_\Omega \left(\frac{\partial I(\mathbf{x})}{\partial \mathbf{v}}\right)^2 \mathrm{d}\mathbf{x} \to \min.$$

This can be rewritten as

$$\int_\Omega (\mathbf{v}^T\, \nabla I(\mathbf{x}))^2\, \mathrm{d}\mathbf{x} = \int_\Omega \mathbf{v}^T\, \nabla I(\mathbf{x})\, (\nabla I(\mathbf{x}))^T\, \mathbf{v}\, \mathrm{d}\mathbf{x}$$

$$= \mathbf{v}^T \int_\Omega \nabla I(\mathbf{x})\, (\nabla I(\mathbf{x}))^T\, \mathrm{d}\mathbf{x}\, \mathbf{v} = \mathbf{v}^T \mathbf{T} \mathbf{v} \to \min$$

where the matrix

$$\mathbf{T} = \int_\Omega \begin{pmatrix} I_x^2(\mathbf{x}) & I_x(\mathbf{x}) I_y(\mathbf{x}) \\ I_x(\mathbf{x}) I_y(\mathbf{x}) & I_y^2(\mathbf{x}) \end{pmatrix} \mathrm{d}\mathbf{x} \tag{2}$$

is called the structure tensor. Including the constraint that $\mathbf{v}$ is a unit vector via a Lagrange multiplicator

$$\mathbf{v}^T \mathbf{T} \mathbf{v} + \lambda(1 - \mathbf{v}^T \mathbf{v}) \to \min$$

and setting the derivative of this term to zero results in $\mathbf{T}\mathbf{v} = \lambda \mathbf{v}$ which means that $\mathbf{v}$ is the eigenvector of $\mathbf{T}$ that corresponds to its lower eigenvalue.

The eigenvalues of $\mathbf{T}$ give an indication of how close the image structure is to an ideal orientation. If both eigenvalues are zero, there is no variation at all and the image has constant intensities in Ω. Neglecting small variations induced by noise, this is often true for background regions or surfaces. On the other hand, both eigenvalues are large if Ω has more than one orientation as at junctions or if there is no structure at all. One small and one large eigenvalue give evidence of an oriented structure and thus high confidence in the computed orientation.

When orientations of an image are analysed, the region Ω is often a small square window that is moved across the image and centred at each pixel in turn. This results in an orientation map for the image where each pixel is assigned an orientation and a measure of confidence based on the eigenvalues which can also be used to identify edge and junction pixels.

The estimation of a single orientation using the structure tensor has been known for 20 years after having been published almost simultaneously by Di Zenzo [4], Bigün and Granlund [2] and by Kass and Witkin [7]. Since then, however, not many efforts have been made to generalize the results in order to estimate multiple orientations as they occur at junctions. The ideas presented so far are summarized in the next section.

3. MULTIPLE ORIENTATION ESTIMATION

3.1 Superposition of Orientations

Iso and Shizawa [6] were the first to address the problem of multiple orientation estimation. The same concept they used for L-, T- and X-junctions was picked up again by Aach et al. [1] who applied it to superimposed oriented patterns. The underlying models assumes the image I to be the sum of two ideally oriented images I_1 and I_2 with orientation vectors $\mathbf{v}_1 = (\cos\theta_1, \sin\theta_1)^T$ and $\mathbf{v}_2 = (\cos\theta_2, \sin\theta_2)^T$ such that

$$\frac{\partial I_1(\mathbf{x})}{\partial \mathbf{v}_1} = \frac{\partial I_2(\mathbf{x})}{\partial \mathbf{v}_2} = 0 \quad \forall\, \mathbf{x} \in \Omega$$

which is equivalent to

$$\frac{\partial^2 I(\mathbf{x})}{\partial \mathbf{v}_1 \partial \mathbf{v}_2} = 0 \quad \forall\, \mathbf{x} \in \Omega \ . \tag{3}$$

In analogy to the single orientation case a minimization problem can be formulated

$$\int_\Omega \left(\frac{\partial^2 I(\mathbf{x})}{\partial \mathbf{v}_1 \partial \mathbf{v}_2} \right)^2 \mathrm{d}\mathbf{x} \to \min. \tag{4}$$

This can be transformed into an eigenvalue problem with the tensor

$$\mathbf{T}_2 = \int_\Omega \begin{pmatrix} I_{xx}(\mathbf{x}) \\ I_{xy}(\mathbf{x}) \\ I_{yy}(\mathbf{x}) \end{pmatrix} \cdot \left(I_{xx}(\mathbf{x}), I_{xy}(\mathbf{x}), I_{yy}(\mathbf{x}) \right) \mathrm{d}\mathbf{x}$$

and eigenvectors of the form

$$(\cos\theta_1 \cos\theta_2, \sin(\theta_1 + \theta_2), \sin\theta_1 \sin\theta_2)^T.$$

These are called mixed orientation vectors and constitute a unique but implicit representation of the two orientation angles. In [6] and [1] different ways are shown how to calculate the angles from the vectors. In principle, this procedure also works for more than two orientations, resulting in larger tensors featuring higher-order derivatives, but in this case no efficient method is known yet for decomposing the mixed orientation vectors. Besides, the algorithm is not able to determine the number of orientations by itself, other than by trying tensors of different size until some confidence criterion based on the eigenvalues is satisfied.

3.2 Vicinity of Orientations

A different model for multiple orientations was proposed by Mota et al. in [8]. In contrast to the superposition approach, it is based on two image regions with different orientations adjoining each other. The region Ω is divided into a subregion P with an orientation vector $\mathbf{v}_1$ and another subregion with $\mathbf{v}_2$. This implies that

$$\frac{\partial I(\mathbf{x})}{\partial \mathbf{v}_1} = 0 \quad \forall\, \mathbf{x} \in P,$$

$$\frac{\partial I(\mathbf{x})}{\partial \mathbf{v}_2} = 0 \quad \forall\, \mathbf{x} \in \Omega \backslash P$$

and thus, as an alternative to (3)

$$\frac{\partial I(\mathbf{x})}{\partial \mathbf{v}_1} \cdot \frac{\partial I(\mathbf{x})}{\partial \mathbf{v}_2} = 0 \quad \forall\, \mathbf{x} \in \Omega.$$

Again this can be turned into a minimization problem

$$\int_\Omega \left(\frac{\partial I(\mathbf{x})}{\partial \mathbf{v}_1} \cdot \frac{\partial I(\mathbf{x})}{\partial \mathbf{v}_2} \right)^2 \mathrm{d}\mathbf{x} \to \min$$

which leads to a tensor

$$\mathbf{T}_3 = \int_\Omega \begin{pmatrix} I_x^2(\mathbf{x}) \\ I_x(\mathbf{x})\, I_y(\mathbf{x}) \\ I_y^2(\mathbf{x}) \end{pmatrix} \cdot \left(I_x^2(\mathbf{x}), I_x(\mathbf{x})\, I_y(\mathbf{x}), I_y^2(\mathbf{x}) \right) \mathrm{d}\mathbf{x}$$

and mixed orientation vectors of the same kind as above. Concerning the generalization to more than two orientations the same holds as for the superposition model. In [8] this method is applied to adjoining oriented patterns and to L- and T-junctions.

4. HISTOGRAM-BASED ORIENTATION ANALYSIS

Junctions are intersection points of several oriented structures like edges or lines. The models of the two methods described above, however, assume the presence of oriented patterns throughout the region Ω which is usually not the case at junctions. The methods often deliver satisfactory results, but they still have the disadvantage of not being able to determine the number of orientations at the junction. Instead, the algorithm has to be given this number as an input. Consequently, an iterative procedure is necessary where the number of orientations is increased until a confidence measure reaches a given threshold value. For Y-junctions which appear frequently in images, three different structure tensors are evaluated because models for one and two orientations are tested first. This increases the runtime of the algorithm considerably, the more so as the tensors become larger during the iteration up to 4×4 for three orientations. Moreover, when the superposition

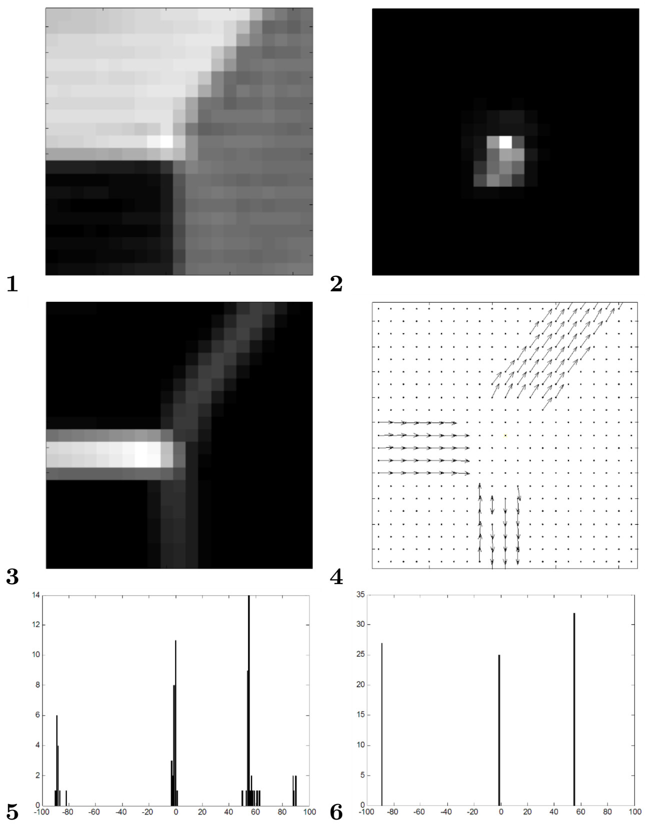

Figure 1: Illustration of the principle of histogram-based orientation analysis. **1** Original image. **2** λ_1 profile. **3** λ_2 profile. **4** Orientation vectors for pixels with $\lambda_1 < 10$ and $\lambda_2 > 50$. **5** Histogram of the orientations within a circular window of radius 9 around the junction centre. **6** Output of the mean shift algorithm for this histogram.

model is used, the structure tensor employs derivatives of increasing order which causes a stronger sensitivity to noise. Finally, the confidence criteria have to be chosen carefully. If two orientations are detected where actually three are present, one cannot assume that the detected two orientations are correct because the underlying model is not suitable for the situation at hand. This generally leads to unusable results.

In the following an algorithm is introduced that is based on an adequate model for junctions and that determines the number of orientations by itself without using an expensive "try and error" procedure. Basically, instead of using a pixel in the centre of the junction as the starting point for determining the orientations of the edges originating from it, it considers the pixels on the edges themselves. For these pixels the orientations can be calculated using the simple structure tensor (2). Edge points are easily recognized by the eigenvalues of this tensor; they fulfill the conditions $\lambda_1 < t_1$ and $\lambda_2 > t_2$ for thresholds that can either be fixed or determined locally by means of a histogram, applying the observation that most of the pixels lie in flat areas and have very small values for λ_1 as well as for λ_2. In practical experiments with 8-bit gray-value images, good results were obtained with $t_1 = 10$ and $t_2 = 50$. Figures 1.2 and 1.3 show the profiles of the two eigenvalues for an example image with a Y-junction. The junction and edge pixels are clearly distinguishable by λ_1 and λ_2, respectively.

In figure 1.4 the orientations computed with the simple structure tensor are plotted for those pixels that have been identified as edge points. In order to get the desired information about the multiple orientation at the junction, it suffices to "collect" the single orientations in a neighbourhood of the central junction pixel. This can be achieved by calculating a histogram of the orientation angles and estimating the principal orientations from it. A suitable bin size is 1°; a finer partition is not reasonable as the usual accuracy of the computed angles is not better than 5°. Figure 1.5 shows the orientation histogram for the example image.

In the ideal case, each histogram peak corresponds to an orientation that is present in the neighbourhood of the junction. Sometimes it suffices just to find local maxima in the histogram that exceed a particular threshold. This can, however, lead to misinterpretations: The histogram in figure 1.5 conveys the impression that the orientation $-89°$ appears less frequently than the two other ones. But this does not take into account that there is a certain variation in the histogram due to the inaccuracy of the calculated orientations. This can be compensated for by using the mean shift algorithm to group angles in the histogram that are likely to represent the same orientation in the image.

The mean shift algorithm was mentioned for the first time by Fukunaga and Hostetler [5] as a method for estimating the modes of an unknown probability distribution. Its relevance for image processing was pointed out e. g. by Comaniciu and Meer [3]. Here only the intuitive one-dimensional version of the algorithm is used. Given the histogram $H(\theta)$ for $\theta = -89°, \dots, 90°$, the iteration

$$\theta_0 = \theta, \qquad \theta_{i+1} = \frac{\sum_{\phi=\theta_i-h}^{\theta_i+h} H(\phi) \cdot \phi}{\sum_{\phi=\theta_i-h}^{\theta_i+h} H(\phi)}$$

is performed for each θ until a fixed point is reached. The sums are over all integers between the given limits. This procedure can be thought of as shifting θ to the centre of gravity of its local neighbourhood of radius h in each iteration step. The iteration converges when a local maximum is reached. The choice of h does not seem to play an important role; in practical experiments good results were obtained with $h = 7$. On the average, two iterations were sufficient, so the complexity is moderate.

Compared to a direct maximum search this method has the advantage that the result is not necessarily an integer and that all occurring angles are mapped to a maximum (the one their iteration converged to) which yields the desired grouping. This procedure is a one-dimensional variant of the approach for image segmentation presented in [3]. An example result is shown in figure 1.6. Here three maxima were found that correspond to the three principal orientations of the image. After applying the mean shift algorithm it is still recommendable to sort out orientations with a frequency less than a threshold.

The question remains which size and shape the neighbourhood of a junction pixel should have within which the histogram is calculated. A square window would favour orientations near 45° and −45°, so a circular window is reasonable. Its radius has to be large enough so that sufficiently many edge points are contained for getting a stable histogram, but a window that is too large can contain edges that are not part of the junction that is being considered. A radius of 9 turned out to be a feasible compromise for most cases, but the optimal choice depends on the junction scale and the closeness of adjacent structures. Sometimes the problem arises that not all pixels in the direct neighbourhood of the junction centre are identified by the threshold criterion. To avoid this, the window should be designed as an annulus with an inner radius of about 3.

5. RESULTS

Figures 1 and 2 show some application examples of histogram-based orientation estimation with different junction types. In a photography showing building blocks, junctions were identified and an orientation analysis was performed. Evidently, our algorithm can handle different kinds of junctions and determine the number of orientations correctly. In figures 2.3/2.4 it identifies an erroneously detected junction by finding only one orientation. Figures 2.5/2.6 show a case where the three orientations within the analysis window Ω are correctly estimated. The algorithm is, however, not able to separate the horizontal orientation from the actual junction formed by the other two orientated structures. This problem, though, occurs with the other algorithms as well.

6. DISCUSSION

Our examples confirm that histogram-based orientation estimation is an efficient and robust algorithm for the analysis of junctions that have been found by some corner detector. It is appropriate for an arbitrary number of orientations but uses only first-order derivatives. The results can be used as a starting point to a more precise examination of the junction by testing in which of the possible directions that are implied by the orientations edges are actually present. One obtains a description of the junction structure which can be used as a feature for correspondence search or object recognition.

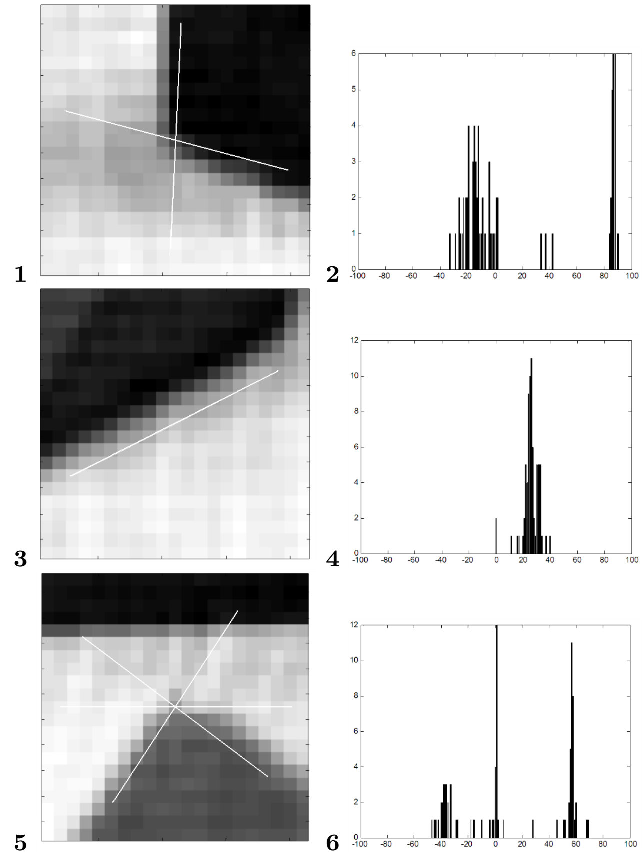

Figure 2: Examples of histogram-based orientation estimation. **1/3/5** Original images with detected orientations, **2/4/6** Orientation histograms before applying the mean shift algorithm. **1/2** L-junction. **3/4** Erroneously detected junction (the error is detected, since the histogram shows only one orientation). **5/6** L-junction and nearby edge. All orientations are correctly estimated, but the junction is erroneously interpreted as having three orientations.

REFERENCES

[1] T. Aach, I. Stuke, C. Mota, and E. Barth, "Estimation of Multiple Local Orientations in Image Signals," *Proc. IEEE ICASSP*, Montréal, 2004, pp. 553–556, 2004.

[2] J. Bigün and G. Granlund, "Optimal Orientation Detection of Linear Symmetry," *Proc. IEEE ICCV*, London, 1987, pp. 433–438.

[3] D. Comaniciu and P. Meer, "Mean Shift Analysis and Applications," *Proc. IEEE ICCV*, Kerkyra, 1999, pp. 1197–1203.

[4] S. Di Zenzo, "A Note on the Gradient of a Multi-Image", *Computer Vision, Graphics, and Image Processing*, vol. 33, pp. 116–125, 1986.

[5] K. Fukunaga and L. Hostetler, "The Estimation of the Gradient of a Density Function, with Applications in Pattern Recognition," *IEEE Transactions on Information Theory*, vol. 21, pp. 32–40, 1975.

[6] T. Iso and M. Shizawa, "Detecting L-, T-, and X-junctions from Low-order Image Derivatives," *SPIE* Vol. 2094, pp. 1185–1197, 1993.

[7] M. Kass and A. Witkin, "Analyzing Oriented Patterns," *Computer Vision, Graphics and Image Processing*, vol. 37, pp. 362–385, 1987.

[8] C. Mota, I. Stuke, T. Aach, and E. Barth, "Estimation of Multiple Orientations at Corners and Junctions," *Proc. DAGM Pattern Recognition Symposium*, Tübingen, 2004, pp. 163–170.

[9] C. Zetzsche and E. Barth, "Fundamental Limits of Linear Filters in the Visual Processing of Two-Dimensional Signals," *Vision Research*, vol. 30, pp. 1111–1117, 1990.