Edge and Junction Detection with an Improved Structure Tensor

Ullrich Köthe

Cognitive Systems Group, University of Hamburg,
Vogt-Köln-Str. 30, D-22527 Hamburg, Germany

Abstract. We describe three modifications to the structure tensor approach to low-level feature extraction. We first show that the structure tensor must be represented at a higher resolution than the original image. Second, we propose a non-linear filter for structure tensor computation that avoids undesirable blurring. Third, we introduce a method to simultaneously extract edge and junction information. Examples demonstrate significant improvements in the quality of the extracted features.

1 Introduction

Since the pioneering work of Förstner [3] and Harris and Stevens [5], the structure tensor has become a useful tool for low-level feature analysis. It gained high popularity for corner detection (see [12] for a review), but applications in edge detection [4], texture analysis [10] and optic flow [9] have also been reported.

However, despite the popularity, applications of the structure tensor for edge and junction detection are facing a number of problems. First, we are showing that the standard method for structure tensor calculation violates Shannon's sampling theorem. Thus small features may get lost, and aliasing may occur. Second, to calculate structure tensors from gradient vectors, spatial averaging is performed by means of linear filters (e.g. Gaussians). The resulting blurring is not adapted to the local feature arrangement and orientation, which may cause nearby features to diffuse into each other. Third, cornerness measures derived from the structure tensor have rather low localization accuracy [12].

A more fundamental problem is the integration of edge and junction detection. From topology we know that a complete boundary description must necessarily incorporate both edges and junctions [7]. Usually, edges and corners/junctions are detected independently. This makes the integration problem quite difficult. Attempts to derive edges and junctions simultaneously from the structure tensor [5,4] have not been very successful. The difficulties are partly caused by the other problems mentioned above, but also stem from the lack of a good method for the simultaneous detection of both feature types.

In this paper we propose three improvements to the structure tensor approach that address the aforementioned problems: we use a higher sampling rate to avoid aliasing; we describe a non-linear spatial averaging filter to improve corner localization and to prevent nearby features from merging; and we develop a new method for the integration of corner/junction and edge detection.

2 The Structure Tensor

Given an image$f(x, y)$, the structure tensor is based on the gradient of f, which is usually calculated by means of Gaussian derivative filters:

$$f_x = g_{x,\sigma} \star f, \qquad f_y = g_{y,\sigma} \star f \tag{1}$$

where $\star$ denotes convolution, and $g_{x,\sigma}, g_{y,\sigma}$ are the spatial derivatives in x- and y-direction of a Gaussian with standard deviation σ:

$$g_\sigma(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{2}$$

The gradient tensor Q is obtained by calculating, at each point of the image, the Cartesian product of the gradient vector $(f_x, f_y)^T$ with itself.

$$Q_\sigma = \begin{pmatrix} q_{11} & q_{12} \\ q_{12} & q_{22} \end{pmatrix} = \begin{pmatrix} f_x^2 & f_x f_y \\ f_x f_y & f_y^2 \end{pmatrix} \tag{3}$$

Spatial averaging of the entries of this tensor, usually with a Gaussian filter, then leads to the structure tensor:

$$S_{\sigma',\sigma} = (s_{ij}), \qquad s_{ij} = g_{\sigma'} \star q_{ij} \qquad (i, j \in \{1, 2\}) \tag{4}$$

σ' is the scale of spatial averaging. Averaging is necessary because the plain gradient tensor has only one non-zero eigenvalue and thus represents only intrinsically 1-dimensional features (edges). Spatial averaging distributes this information over a neighborhood, and points that receive contributions from edges with different orientations will have two positive eigenvalues, which allows them to be recognized as intrinsically 2D. Cornerness is then measured by the strength of the intrinsically 2D response, for example:

$$c_1 = \frac{\det(S_{\sigma',\sigma})}{\mathrm{tr}(S_{\sigma',\sigma})} \qquad \text{or} \qquad c_2 = \det(S_{\sigma',\sigma}) - 0.04(\mathrm{tr}(S_{\sigma',\sigma}))^2 \tag{5}$$

The first measure is commonly known as *Förstner's operator* [3], although it was independently proposed by several authors. The second one originates from Harris and Stevens [5] and is called *corner response function*. Rohr [11] later simplified these measures by searching for local maxima of the determinant alone.

3 Improvement I: Correct Sampling

Let us assume that the original image $f(x, y)$ was properly sampled at the Nyquist rate. Setting the pixel distance $\lambda_N = 1$ in the spatial domain, this means that f must be band-limited with cut-off frequency $\omega_N = \pi$:

$$|\omega_1|, |\omega_2| \geq \pi \Rightarrow F(\omega_1, \omega_2) = 0 \tag{6}$$

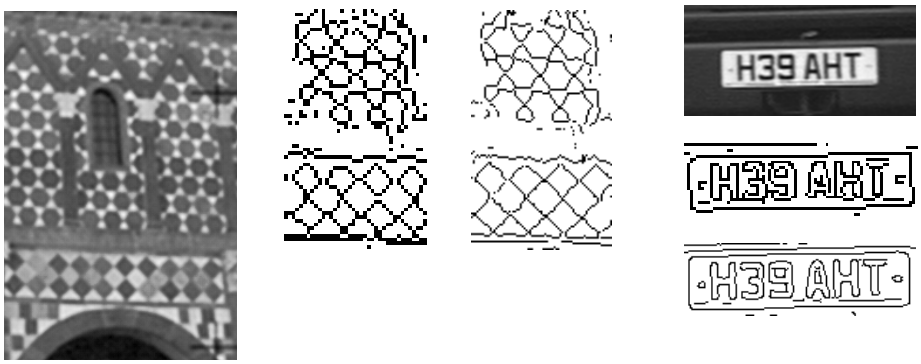


Fig. 1. Original images and their Canny edges at the original and doubled resolutions. The small white tiles in the left image have a diameter of about 3 pixels. Characters in the license plate have a line width of 2 pixels. More examples can be found in [6].

where F denotes the Fourier transform of f, and ω_1, ω_2 are the frequency coordinates. Convolution of f with Gaussian derivative filters corresponds to a multiplication of the spectrum F with the Fourier transforms G_x, G_y of the filters. Since Gaussian derivatives are not band-limited, the derivative images f_x and f_y are still band-limited with $\omega_N = \pi$. Next, we calculate the Cartesian product of the gradient vector with itself. Pointwise multiplication of two functions in the spatial domain corresponds to convolution in the Fourier domain:

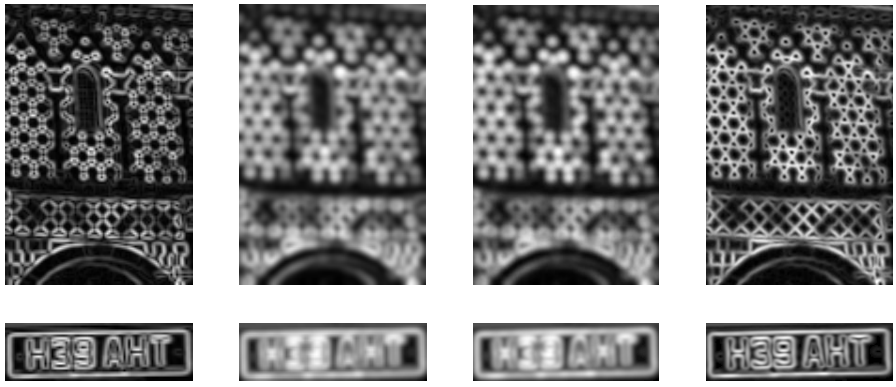
$$f_1 f_2 \quad \circ\!\!-\!\!\bullet \quad F_1 \star F_2 \tag{7}$$

Convolution of two spectra with equal band width *doubles the band width.* Therefore, in order to avoid aliasing and information loss, the elements of the gradient tensor must be represented with *half the sample distance* of the original image. Surprisingly, this important fact has been overlooked so far. As we will see, correct sampling leads to significant improvements in the quality of the edges and corners obtained later on. Oversampling is best realised directly during the calculation of the derivative images. Consider the definition of the convolution of a discrete image f with an analog filter kernel g:

$$(f \star g)(x, y) = \sum_{i,j} f(i, j) g(x - i, y - j) \tag{8}$$

Despite f being discrete, the right-hand side of this equation is an analog function that can be evaluated at arbitrary points (x, y). We obtain an oversampled derivative image by evaluating $f \star g_x$ and $f \star g_y$ at both integer and half-integer positions.

The problem of insufficient sample density is not limited to structure tensor based methods, it affects all algorithms that take products of derivatives. In fig. 1 we compare edges detected with Canny's algorithm [2] at the original and doubled resolutions. The differences in quality are clearly visible. Of course, oversampling is not necessary if the original image does not contain fine scale structure.



(a) (b) (c) (d)

Fig. 2. (a) gradient magnitude at original resolution; (b) trace of structure tensor – original resolution, linear averaging; (c) trace of structure tensor – doubled resolution, linear averaging; (d) trace of structure tensor – doubled resolution, non-linear averaging

4 Improvement II: Non-linear Spatial Integration

The gradient tensor has only one non-zero eigenvalue and thus only represents intrinsically 1-dimensional features (edges). Spatial averaging, usually with Gaussian filters, distributes this information over a neighborhood. Unfortunately, location independent filters do not only perform the desirable integration of multiple edge responses into corner/junction responses, but also lead to undesirable blurring of structure information: If two parallel edges are close to each other, they will be merged into a single edge response, and the narrow region between them is lost. Similarly, edges around small circular regions merge into single blobs, which are erroneously signalled as junctions. Figures 2 b and c demonstrate these undesirable effects.

The reason for the failure of the integration step lies in the linear nature of the averaging: the same rotationally symmetric averaging filter is applied everywhere. This is not what we actually want. Structure information should be distributed only *along* edges, not perpendicular to them. Hence, it is natural to use *non-linear* averaging filters. Such filters were proposed in [9] (based on unisotropic Gaussians) and [13] (based on unisotropic diffusion). In both cases the local filter shape resembles an oriented ellipse whose orientation equals the local edge direction. However, in our experiments elliptic filters did not lead to significant improvements over the traditional isotropic integration.

Therefore, we propose to use oriented filters that are shaped like *hour-glasses* rather than ellipses (fig. 3). This type of filter can be interpreted as encoding the likely continuations of a local piece of edge. Our filter was inspired by methods used in perceptual grouping, e.g. tensor voting [8] and curve indicator random fields [1]. In contrast to those, in our application short-range interaction (over

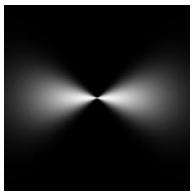


Fig. 3. Hour-glass like filter according to (9), with $\rho = 0.4$ amd $\phi_0 = 0$.

at most a few pixels) is sufficient, and there is no need for separate treatment of straight and curved edges. Thus, we can use a very simple filter design.

We define our non-linear filter kernels as polar separable functions, where the radial part is a Gaussian filter, but the angular part modulates the Gaussian so that it becomes zero perpendicular to the local edge direction $\phi_0(x, y)$:

$$h_{\sigma',\rho}(r, \phi, \phi_0) = \frac{1}{N} e^{-\frac{r^2}{2\sigma'^2}} e^{-\frac{\tan(\phi-\phi_0)^2}{2\rho^2}} \tag{9}$$

where ρ determines the strength of orientedness, and N is a normalization constant that makes the kernel integrate to unity. At every point in the image, this kernel is rotated according to the local edge orientation defined by a unit vector $\boldsymbol{n}(x, y) = (\cos(\phi_0), \sin(\phi_0))^T$ which is perpendicular to the gradient. In the local coordinate system defined by $\boldsymbol{n}$ and $\boldsymbol{n}_\perp$, ϕ_0 is zero. Let p, q be Cartesian coordinates in this local coordinate system. Then $r^2 = p^2 + q^2$ and $\tan(\phi - \phi_0) = \frac{q}{p}$. When $r = 0$, we set $\phi := 0$ in order to avoid damping the radial Gaussian at the center of the filter. Given $\boldsymbol{n}$, p and q can be calculated very efficiently from the global coordinates $\boldsymbol{x} = (x, y)$, namely $p = \boldsymbol{n}^T\boldsymbol{x}$ and $q = \boldsymbol{n}_\perp^T\boldsymbol{x}$. In Cartesian coordinates, our kernel thus reads:

$$h_{\sigma',\rho}(\boldsymbol{x}, \boldsymbol{n}) = \frac{1}{N} \begin{cases} e^{-\frac{\boldsymbol{x}^T\boldsymbol{x}}{2\sigma'^2} - \frac{1}{2\rho^2}\left(\frac{\boldsymbol{n}_\perp^T\boldsymbol{x}}{\boldsymbol{n}^T\boldsymbol{x}}\right)^2} & \text{if } \boldsymbol{n}^T\boldsymbol{x} \neq 0 \\ 0 & \text{if } \boldsymbol{n}^T\boldsymbol{x} = 0, \quad \boldsymbol{n}_\perp^T\boldsymbol{x} \neq 0 \\ 1 & \text{otherwise} \end{cases} \tag{10}$$

The nonlinear integration operator $\mathcal{T}$ is defined as:

$$\begin{aligned} T_{\sigma,\sigma',\rho} =& \mathcal{T}_{\sigma',\rho}[Q_\sigma] \\ t_{ij}(x, y) =& \sum_{x',y'} h_{\sigma',\rho}\big(x - x', y - y', \boldsymbol{n}(x', y')\big) q_{ij}(x', y') \quad (i, j \in \{1, 2\}) \end{aligned} \tag{11}$$

Fig. 3 depicts this filter for a horizontal edge. The parameter ρ should be as small as possible in order to obtain pronounced orientedness. We have found experimentally that the filter results are not very sensitive to the choice of ρ – values between 0.3 and 0.7 give essentially the same results. However, for $\rho < 0.3$, the filter becomes susceptible to noise in the estimated direction $\boldsymbol{n}$. For $\rho > 0.7$, undesirable blurring becomes visible again. In the examples, we use $\rho = 0.4$. This means that the kernel amplitude at $\phi = \phi_0 \pm 25°$ is half the maximal amplitude at ϕ_0. The parameter σ' must be large enough to ensure sufficient

overlap between the different edge contributions coming from the neighborhood of a junction. We have found that the averaging scale should be about twice as large as the scale of the gradient filter. Since the structure tensor is represented with doubled resolution, this means that $\sigma' = 4\sigma$. Experiments were done with $\sigma = 0.7$. A theoretical investigation of optimal choices for ρ and σ' will be conducted. The possibility to improve efficiency by means of steerable filters will also be explored.

Fig. 2 shows the trace of the structure tensor obtained by our new filter and compares it with the trace of the structure tensor calculated with linear integration. It can be seen that nearby edges are merged in the linearly smoothed version, which causes small and narrow regions to disappear. This does not happen with non-linear averaging.

5 Improvement III: Integrated Edge and Junction Detection

In many cases the structure tensor is subsequently used to derive a cornerness measure, e.g. c_1 or c_2 in (5). Since a complete boundary description needs both edges and corners, edges are then detected with another algorithm, such as Canny's [2]. This poses a difficult integration problem of edge and corner responses into a single boundary response. Displacements of the detected corners from their true locations and erroneous edge responses near corners and junctions often lead to topologically incorrect boundaries (gaps, isolated "junction" points etc.). These problems have to be repaired by means of heuristics or dealt with by robust high level algorithms. Obviously, it were better if the errors would be avoided rather than repaired. This should be possible if edges and junctions arose from a unified, integrated process. However, this is not straightforward. For example, Förstner [4] tried to derive edge information from the structure tensor as well, but reliablility was not really satisfying.

The improvements to the structure tensor proposed above open up new possibilities for simultaneous edge and junction detection. We base our new edgeness and cornerness measures on the fact that any positive semi-definite second order tensor can be decomposed into two parts, one encoding the intrinsically 1-dimensional properties of the current location (edge strength and orientation), and the other the intrinsically 2D properties:

$$T = T_{\text{edge}} + T_{\text{junction}} = (\mu_1 - \mu_2)\boldsymbol{n}_1\boldsymbol{n}_1^T + \mu_2 I \tag{12}$$

where $\mu_{1,2}$ are the eigenvalues of the tensor, $\boldsymbol{n}_1$ is the unit eigenvector associated with μ_1, and I is the identity tensor. The eigenvalues are calculated as:

$$\mu_{1,2} = \frac{1}{2}\left(t_{11} + t_{22} \pm \sqrt{(t_{11} - t_{22})^2 + 4t_{12}^2}\right) \tag{13}$$

and the eigenvector is

$$\boldsymbol{n}_1 = \begin{pmatrix}\cos(\phi_1)\\ \sin(\phi_1)\end{pmatrix} \qquad \text{with} \qquad \phi_1 = \frac{1}{2}\arctan\left(\frac{2t_{12}}{t_{11} - t_{22}}\right) \tag{14}$$

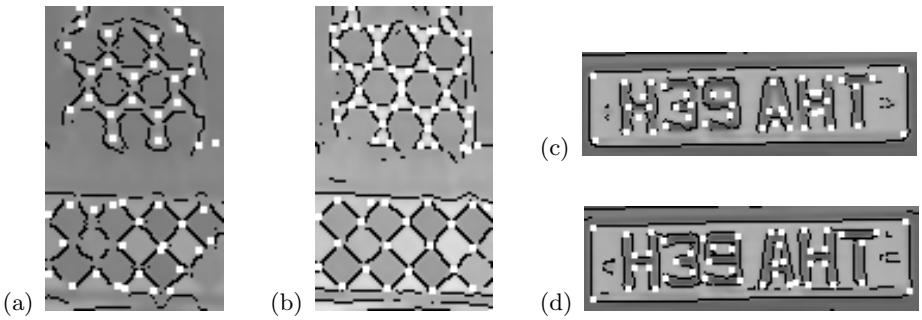


Fig. 4. (a, c) integrated edge and junction detection – linear structure tensor calculation; (b, d) integrated edge and junction detection – non-linear structure tensor calculation. All images are calculated at doubled resolution.

Corners and junctions can now be detected as local maxima of $\mathrm{tr}(T_{\mathrm{junction}})$, whereas T_{edge} can be transformed back into a gradient-like vector $\sqrt{\mu_1 - \mu_2}\,\boldsymbol{n}_1$ that can be fed into Canny's algorithm instead of the normal gradient. Thus, the detected corners/junctions and edges arise from a decomposition of the same original tensor representation which leads to much fewer errors in the resulting boundary.

Fig. 4 compares edges and junctions derived from the standard structure tensor with those from the improved one. This figure reinforces what fig. 1 already demonstrated for edge detection alone: doubling of the resolution and non-linear tensor filtering indeed improve the boundary quality. In fig. 4a, the most severe error is that junctions are hallucinated in the centers of the small triangular regions, because the edges of these regions are merged into a single blob during blurring. Fig. 4c exhibits low quality of the detected edges, again because nearby edges diffuse into each other. Fig. 5 shows an example where the traditional tensor already performs reasonably. But by looking closer one finds the corners to be displaced by 3 pixels from their true locations, whereas the displacement in the non-linearly smoothed tensor is at most 1 pixel.

6 Conclusions

In this paper we improved structure tensor computation in two important ways: increased resolution and non-linear averaging. These improvements allowed us to define a new integrated edge and junction detection method. The experiments clearly indicate that the new method is superior, especially if the image contains small features near the resolution limit, as is typical for natural images.

In order to improve the method further, a better theoretical understanding of the non-linear averaging is required. It should also be investigated if Canny-like non-maxima suppression is optimal for the linear part T_{edge} of our tensor. Furthermore, quantitative comparisons with existing approaches will be conducted.

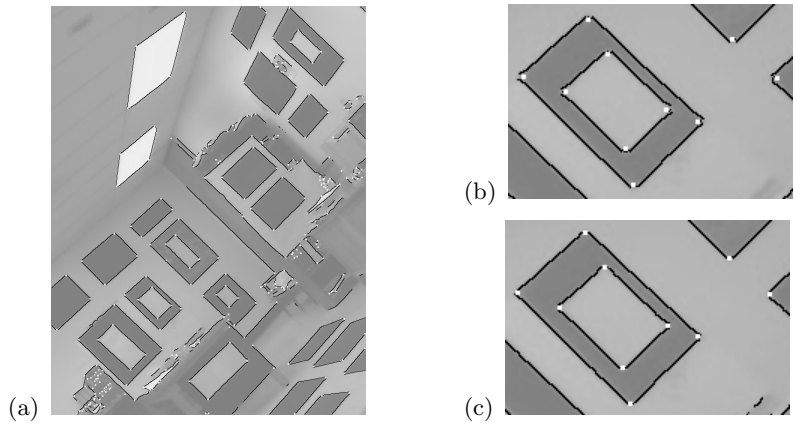


Fig. 5. (a) integrated edge and junction detection in a lab scene; (b) detail of (a) computed with linear tensor averaging; (c) the same region as (b) obtained with nonlinear averaging. Note the corner displacements in (b).

References

1. J. August, S. Zucker: *Sketches with Curvature: The Curve Indicator Random Field and Markov Processes*, IEEE Trans. Patt. Anal. Mach. Intell., 25(4), 387-400, 2003
2. J. Canny: *A Computational Approach to Edge Detection*, IEEE Trans. Patt. Anal. Mach. Intell., 8(6), pp. 679-698, 1986
3. W. Förstner: *A Feature Based Corresponding Algorithm for Image Matching*, Intl. Arch. of Photogrammetry and Remote Sensing, vol. 26, pp. 150-166, 1986
4. W. Förstner: *A Framework for Low Level Feature Extraction*, in: J.-O. Eklundh (Ed.): Computer Vision - ECCV'94, Vol. II. Springer LNCS 801, pp. 383-394, 1994
5. C.G. Harris, M.J. Stevens: *A Combined Corner and Edge Detector*, Proc. of 4th Alvey Vision Conference, 1988
6. U. Köthe: *Gradient-Based Segmentation Requires Doubling of the Sampling Rate*, Univ. Hamburg, Informatics Dept., Tech. Rep. FBI-HH-M-326/03, 2003, subm.
7. V. Kovalevsky: *Finite Topology as Applied to Image Analysis*, Computer Vision, Graphics, and Image Processing, 46(2), pp. 141-161, 1989
8. G. Medioni, M.-S. Lee, C.-K. Tang: *A Computational Framework for Segmentation and Grouping*, Elsevier, 2000
9. H.-H. Nagel, A. Gehrke: *Spatiotemporally adaptive estimation and segmentation of OF-fields*; in: H. Burkhardt and B. Neumann (Eds.): Computer Vision - ECCV'98, Springer LNCS 1407, pp. 86–102, 1998
10. A. R. Rao, B. G. Schunck: *Computing Oriented Texture Fields*, CVGIP: Graphical Models and Image Processing, vol. 53, no. 2, 1991, pp. 157-185
11. K. Rohr: *Modelling and Identification of Characteristic Intensity Variations*, Image and Vision Computing, vol. 10, 66-76, 1992
12. K. Rohr: *Localization Properties of Direct Corner Detectors*, J. of Mathematical Imaging and Vision, 4, pp. 139-150, 1994
13. J. Weickert, T. Brox: *Diffusion and Regularization of Vector- and Matrix-Valued Images*, in: M. Z. Nashed, O. Scherzer (eds.), Inverse Problems, Image Analysis, and Medical Imaging. Contemporary Mathematics, Vol. 313, AMS, 2002