TEXTURE BOUNDARY TRACKING WITH GABOR PHASE

Stefan Fischer, Josef Bigün

Signal Processing Lab., Swiss Federal Institute of Technology
CH-1015 Lausanne, Switzerland
e-mail: fischer@ltssg3.epfl.ch

Abstract

In this chapter we present a method for texture boundary tracking based on Gabor phase. We show that phase alone can be used as feature for texture boundary discrimination. Classic methods for texture discrimination only use Gabor magnitude and discard phase information in spite of the fact that phase represents important information use for to texture discrimination. Since phase unwrapping in the two-dimensional case is very costly and is not robust enough, we use the phase derivative or phase gradient for texture discrimination. A boundary tracking algorithm that uses a "butterfly" shaped test region is used in order to find the best discrimination between two regions. We show that Gabor phase alone contains enough information for texture boundary tracking and that it can be used to improve and complement texture segmentation methods.

1. Introduction

This chapter attempts to give a role to Gabor phase in form of the phase gradient by using it to track boundaries of textured regions accurately. The reason to examine Gabor phase for texture perception is that the human visual system shows a good performance in distinguishing regions that have the same magnitude but contain patterns with different frequencies or phase shifts. Based on this observation we assume that Gabor phase plays an important role in texture perception. The fact that the human visual system discriminates textures differing in phase information alone, makes it interesting to examine if Gabor phase features can be used in order to track texture boundaries.

The images in Fig. 1–3 intend to demonstrate the significance of orientation, phase and frequency for texture perception. The images show a textured circle on a textured background. The textures are single frequency sinusoids. Although the Gabor amplitude in all regions is the same, the human visual system is able to perceive the form of the circle. It is difficult to build a model for the boundary itself since the two neighboring regions form different patterns depending on the orientation of the border. An attempt to find the boundaries by analyzing the individual magnitude responses results in a discontinuous contour. A collection of Gabor magnitude responses originating from different orientations used in local orientation detection schemes, e.g. [11, 3, 2] can resolve the discontinuity problem on the cost of good localization.

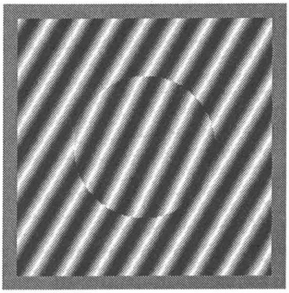

Fig. 1. Rotation.

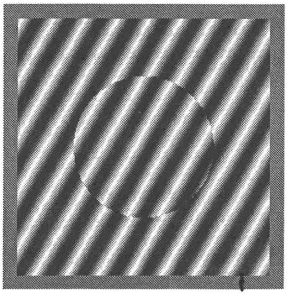

Fig. 2. Phase shift.

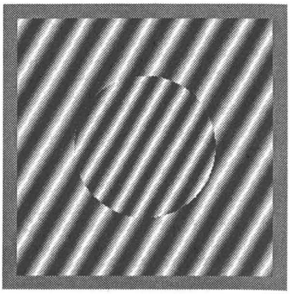

Fig. 3. Frequency deviation.

In [2] a method for texture segmentation based on Gabor magnitude is used. In this approach phase information is discarded. [12] proposes a texture segmentation scheme that is based on quad-tree smoothing followed by a multidimensional clustering. It uses an orientation-adaptive boundary refinement to improve the accuracy of the boundaries. The boundary tracking method proposed in this paper attempts to use Gabor phase for improving such coarse estimations of texture boundaries.

It is generally agreed upon that there is no precise definition of the term *texture* [4, 5]. The definition depends on the type of the images we want to examine. Some images seem to suggest a statistic approach for texture segmentation, other images contain textures with a more regular characteristic. Structural pattern recognition methods consider textures as patterns that are composed of lines and edges. In the case of regular textures geometrical properties could be used to describe and discriminate regions. A texture will be called *regular* if it consists of few distinct fundamental frequencies and thus can be described by means of geometrical primitives.

2. Gabor Decomposition

Gabor filters are oriented, multi-resolution bandpass filters. In the frequency domain the Gabor functions can be represented as two-dimensional Gaussian functions. We use the same Gabor decomposition model described as in [2] where the bandwidth of subsequent filters increase by factor two when the center frequencies increase. The parameters that can be chosen are the minimal and maximal frequency, that is the lowest and highest inflexion points of all Gaussians in the frequency domain, and the number of resolutions and orientations. The filters can be described in the frequency domain as follows:

$$G_{ij}(\omega_r, \omega_\varphi) = \exp\left(\frac{-(\omega_r - \omega_{ri})^2}{2\sigma_{ri}^2}\right) \cdot \exp\left(\frac{-(\omega_\varphi - \omega_{\varphi j})^2}{2\sigma_\varphi^2}\right) \tag{1}$$

i and j refer to the filter positions and the radial and angular bandwidths of the filters.

Important properties of the chosen Gabor decomposition are the orientation resolution and the scale resolution. Other properties are the amount of overlapping and the ratio between orientation sensitivity and scale bandwidth.

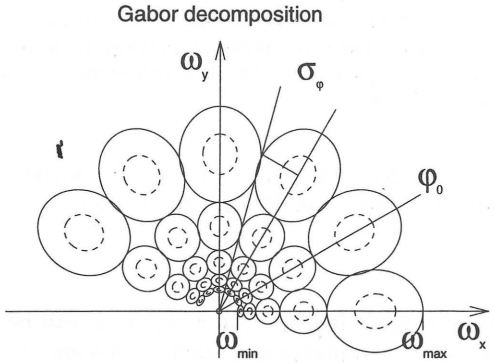

Fig. 4. An example of a Gabor decomposition. The ellipses represent the inflexion points of the two-dimensional Gaussians that are used as bandpass filters. The dashed ellipses show an alternative way of decomposing the frequency plane. In this case the overlapping of the Gaussians is minimal.

The number of resolutions and orientations that one can use is limited mainly because of computational constraints and the image quality. For texture boundary analysis it might be sufficient to use only a few resolutions that correspond to high frequencies. The detection of lines and edges and more complex objects based on stability analysis requires a wider range of frequencies and more resolutions.

We have chosen 6 orientations and 3 resolutions for the texture boundary tracking scheme. For the texture discrimination an overlapping has been chosen where the Gaussians in the frequency domain overlap at their inflexion points. We need the overlapping to make sure that a fundamental frequency between two filter bands is not suppressed. Fig. 4 shows an example for a Gabor decomposition.

The amount of overlapping depends on the application for which we use the Gabor decomposition. Possible criteria for the amount of overlapping are the accuracy of the reconstruction, the accuracy of the detection of lines and edges in scale space or the performance of a particular texture segmentation scheme. Image analysis and

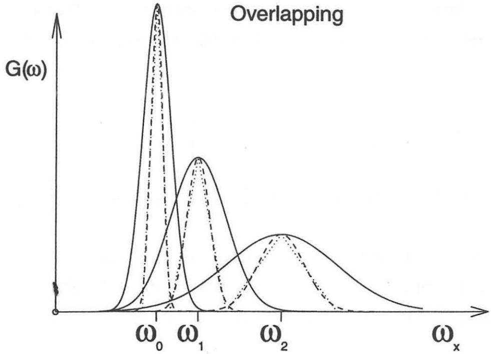

Fig. 5. This figure shows two different methods of decomposing the frequency domain. 1) decomposition with overlapping, 2) decomposition with minimal overlapping. In the second case the Gaussians approximate a triangle in a least square sense. The amplitude at the overlapping point is less then 5 percent of the maximal amplitude.

detection schemes that are based on scale-space stability might require a significant overlapping of the frequency bands in different resolutions.

3. Gabor Phase

In the ideal case of a single frequency in a region, it can be shown that Gabor phase is linear. We use a one-dimensional example to show the linearity property. The transformation of a cosine function with frequency ω_0 in the frequency domain is shown in Eq. (2). The sign ○—● represents the Fourier transform, $\hat{f}(\omega)$ denotes the function in the Fourier domain. $f(x)$ is the input, $r(x)$ denotes the response.

$$f(x) = \cos(\omega_0 x) \circ\!\!-\!\!\bullet\ \hat{f}(\omega) = \delta(\omega - \omega_0) + \delta(\omega + \omega_0) \tag{2}$$

The Gabor filter selects only a narrow band of the Fourier domain with positive frequency. We see now from Eq. (3) that the phase of the complex response is linear:

$$\hat{f}(\omega) = a\delta(\omega - \omega_0) \bullet\!\!-\!\!\circ\ r(x) = a\frac{1}{2\pi}e^{i\omega_0 x} \tag{3}$$

where a is the amplitude of the Gabor filter at ω_0. It is now clear that a Gabor decomposition with minimal overlapping can suppress fundamental frequencies lying in between the tune-frequencies, and is therefore not suitable for texture segmentation. Thus the phase of each complex response approximates the linear phase. The variation of the phase derivative can be considered as a measure for the linearity of the phase.

Different methods for phase unwrapping have been presented e.g. [13], [6]. These phase unwrapping algorithms perform well on one-dimensional signals but not always well on two-dimensional phase signals. A method to avoid difficulties with phase unwrapping in the first place is to use the phase derivative or phase gradient. Fleet and Jepson [8] use the phase gradient instead of the unwrapped phase for motion estimation. Using the phase gradient instead of phase has the advantage that no unstable phase unwrapping operations are needed. The appendix 1 shows how phase derivatives can be calculated from the derivatives of the real and imaginary parts of the the complex response.

Areas where the magnitude of the complex response are close to zero are called *singularities.* In such regions the phase and the phase gradient looses its significance. [10] and [9] present a method for detecting areas that surround singularities. In [9] this method is used for detecting areas with stable phase with regard to varying scale.

4. Boundary Modeling by Gabor Phase

A texture boundary model is a way of describing and distinguishing different textures. If we can develop a model for texture boundaries, we are able to classify textured regions and we can apply it for texture boundary tracking. The accuracy of the boundary tracking scheme depends on the size of the regions we use to find the model parameters.

An approach to define texture boundaries is to find a model of the phase. Here we define the texture boundaries as areas that separate two regions with different texture models. This approach relies on coherent texture models that describe local properties of the phase within individual texture patches.

Applications of texture models and texture boundary models might also be found in texture coding and texture analysis. For our application we have chosen a linear feature discrimination scheme to find the position that corresponds to the best estimation of a texture boundary. We do not use a descriptive model for textures but search for the best linear discrimination between the features of two regions. In the case of an ideal texture with only one dominant frequency the phase gradient is a constant. We assume that for natural textures the phase gradient is constant for sufficiently homogeneous textures and sufficiently dense Gabor filters.

4.1. Feature Discrimination

The linear Fisher discriminant [7], [14], can be used to find an optimal linear discrimination between two regions that contain a number of features. The Eq. (4) shows the definition of the sample mean vectors μ_1 and μ_2 . n_1 and n_2 are the number of points in region C_1 and C_2.

$$\mu_1 = \frac{1}{n_1} \sum_{x_i \in C_1} x_i, \qquad \mu_2 = \frac{1}{n_2} \sum_{x_i \in C_2} x_i \tag{4}$$

The sample covariance matrices of features in region j are defined in Eq. (5).

$$R_j = \frac{1}{n_j} \sum_{x_i \in C_j} (x_i - \mu_j)(x_i - \mu_j)^T \qquad (5)$$

The linear discriminant is given by $|\nu_1|$. $|\nu_1|$ is a measure of separation:

$$\nu_1 = \alpha^T(\mu_2 - \mu_1) \qquad (6)$$

Where ν_1 shows the best discrimination when α is chosen as in Eq. (7).

$$\alpha = (\mu_2 - \mu_1)^T R^{-1} \qquad (7)$$

Eq. (7) inserted in Eq. (6) yields Eq. (8).

$$\nu_1 = (\mu_2 - \mu_1)^T R^{-1}(\mu_2 - \mu_1) \qquad (8)$$

R in Eq. (7) is the sum of the two covariance matrices $R = R_1 + R_2$. In the case of only one feature we can simplify Eq. (8):

$$\nu_1 = \frac{(\mu_2 - \mu_1)^2}{\sigma_1^2 + \sigma_2^2} \qquad (9)$$

σ_1 and σ_2 are the sample standard deviations of the the two features. The linear discriminant is a dimensionless value that can be used as a measure for the best separation between two classes with a number of features. We assume that the features have approximately the same mean and standard deviation in a region.

4.2. Phase Derivatives and Boundary Tracking

We use a search algorithm to follow the texture boundaries. This procedure allows to reduce the two-dimensional boundary tracking problem to a one-dimensional optimization problem by searching the boundary along a line.

For tracking the boundary we use two test regions. We call these test regions a *sensor* that gives a measure for texture discrimination depending on the position on the image. The proposed algorithm slides the sensor on a line that is perpendicular to the polygon segments that describe the search corridor (see Fig. 6 and Fig. 7). We assume that we have already a rough approximation of the border in form of a polygon. The sensor seeks to maximize a cost function. This cost function is calculated from statistical properties like mean and standard deviation of the phase derivative in the two neighboring regions. Eq. (10) represents a cost function for one resolution and one orientation. σ_i is the standard deviation of the phase derivative in a region i with the mean value μ_i.

$$C = \frac{(\mu_1 - \mu_2)^2}{\sigma_1^2 + \sigma_2^2} \tag{10}$$

A boundary is defined by the properties of the neighboring regions. The phase measurements on the border itself are intrinsically associated with large uncertainties. Consequently the sensor has to be insensitive when its center reaches the boundary. If the region in the center of the sensor is too large, the accuracy of the procedure is affected. Another argument for a small center is the fact that the form of the sensor in the center affects the boundary shape that is detected. The test regions of the sensor could be touched or cut by other parts of the boundary or other boundaries. So the test regions have to be as small as possible to be able to place them completely in a texture region.

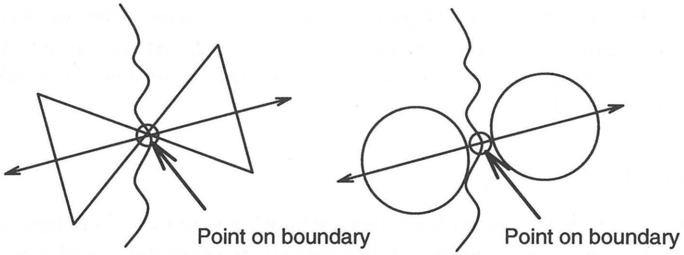

Fig. 6. Texture boundary sensors. This figure shows two different sensors that are insensitive in the center and thus satisfy the requirements for boundary sensors. The sensor on the right hand side can touch other parts of the boundary especially when the orientation of the boundary changes abruptly.

The accuracy and the stability of the search algorithm depend on the size of the sensor wings as well as the shape of the sensor, e.g. the distribution of sensitivity along the search line. Other parameters that can influence the performance of the boundary tracking are the distance between the gravitational centers of the left and right wing and the search strategy and step-size between the search points on the line.

Fig. 6 shows two different sensors with different shapes of the test regions. The butterfly shaped test regions have been chosen to minimize the influence of the region on the border itself. Values of phase derivative on the border are not taken into consideration when the center of the butterfly lies directly on the border. The size of the sensor is linked to the frequency band that is being examined. Lower frequency bands require larger sensors, which have a lower accuracy than a small sensor for high frequency bands. We define a border by the phase derivative values of the two adjoining regions and not by points on the border.

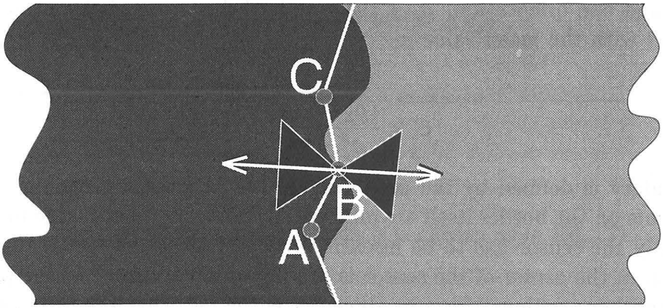

Fig. 7. The values of the phase gradient in the two wings of the butterfly shaped test region are used to compare the features of the regions. The initial estimation of the boundary has the form of a polygon.

Fig. 7 shows the butterfly shaped sensor on the boundary. An initial estimation of the boundaries in form of polygons is used. We assume that we have such an estimation e.g. provided by the method described in [12]. At the same time we make assumptions about the width of a search corridor that describes the possible search region for the butterfly.

4.3. Iterative Improvements

In the case where the initial boundary estimation is not good enough or when the chosen search corridor is too narrow, it is necessary to track the boundaries iteratively. To increase the stability of the iteration, the size of the butterfly regions could be decreased with the number of iterations. The final polygon can be approximated by a spline and smoothness constraints can be introduced to the approximating spline. Especially crossings of border lines e.g. corners with more than two different neighboring textures introduces difficulties. Smoothness constraints might improve the results in such areas where the orientation of the texture is not well defined.

5. Results

The test images contain natural textures that have been separated by a straight line in horizontal direction and by a randomly curved line in vertical direction. The regions have been mean and variance normalized in order to rule out the detection of boundaries by simple edge detection methods. The borders that have been used to separate the patches are the same for the two test images. A Gabor decomposition with 6 orientations and 3 resolutions has been used. Fig. 8 contains natural textures, that are rather inhomogeneous and show few distinct orientations. The image in Fig. 10 has more regular patches with distinct orientations. Fig. 9 shows the result of the texture boundary tracking on the image in Fig. 8. Fig. 11 shows the result of

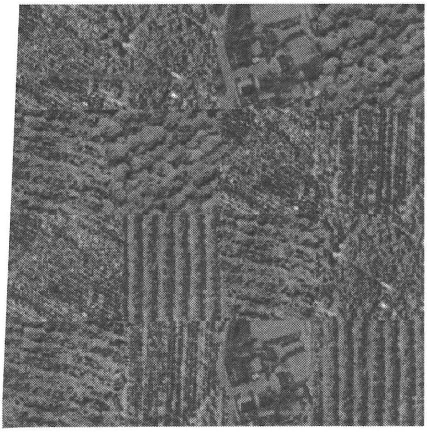

Fig. 8. This image shows the original test image composed of real texture patches.

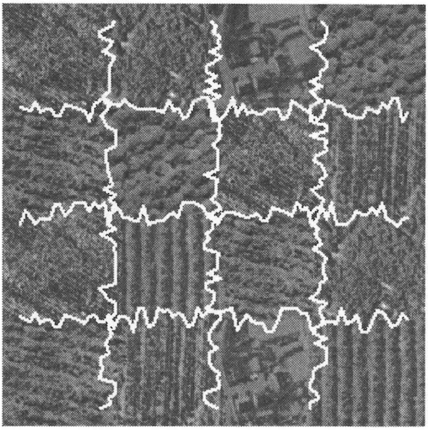

Fig. 9. This image shows the result of the texture boundary tracking for the first test image. The resulting texture boundary is not very stable, the points oscillate around the boundary.

the boundary tracking for the image with more regular textures.

We have assessed the performance of the boundary tracking scheme by comparing its results with the artificially introduced boundaries. However by combining natural textures with synthetic boundaries, artificial boundaries and artifacts might be created that differ from the boundaries used to create the test images. The performance of a texture boundary tracking scheme can only be assessed subjectively. Only for synthetically produced ideal textures we can find an objective performance measure.

6. Discussion

We have presented a method for tracking texture boundaries by using Gabor phase. Our main conclusion is that regular textures can be tracked with high accuracy, by using a simple statistical discrimination method. The best performance can be achieved with textures with few distinct fundamental frequencies, which is in line with previous observations using very simple input images [1]. By using only the best discriminant in all resolutions and orientations, we do not encounter difficulties with phase singularities. The iterative application of the tracking algorithm results in some improvements but at the same time it is not stable in regions where boundaries are crossing. A subsequent smoothing of the polygon or the application of smoothing constraints could improve the result especially in corner areas where more than two neighboring regions meet.

It has been assumed that a rough estimation of the borders is available. This

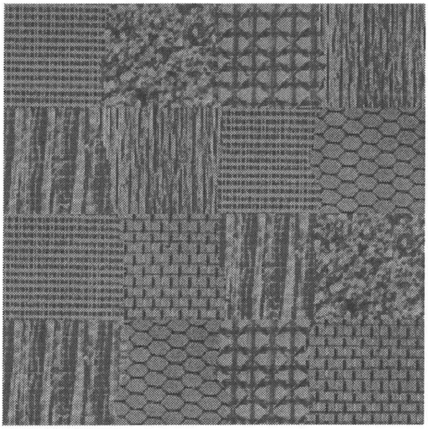

Fig. 10. This figure shows second test image with more regular textures than in the first test image.

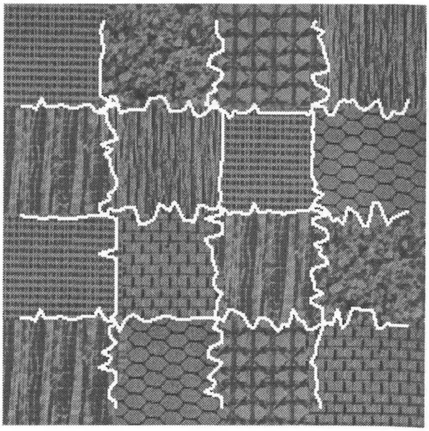

Fig. 11. This image shows the result of the texture boundary tracking. The boundary can not be tracked properly in corner regions where there are more than two neighboring regions.

assumption is reasonable since methods based on smoothed feature clusters yielding approximate boundaries are available [12]. The proposed boundary tracking algorithm provides an analysis tool that supplements other multi-scale detection or classification schemes. It is possible to use the proposed boundary tracking algorithm for unsupervised boundary tracking if the ambiguity problems of boundary crossings are dealt with.

Experimentally we have shown that Gabor phase can be used in texture discrimination, but Gabor magnitude also contains information about the significance of phase which is relevant for the detection of phase singularities.

Therefore we propose to examine texture segmentation procedures that use the complex response in its entirety. For this purpose classification methods capable of dealing with multi dimensional complex feature vectors along with automatic selection of relevant features and feature reduction techniques need to be developed.

Acknowledgments

This project is supported by the Swiss National Science Foundation, grant number 21-33641.92.

Appendix A. Phase Gradient Computation

The phase gradient can be calculated from the spatial derivatives of the response. Differentiation of the complex response can be obtained by a multiplication of the frequency variable with the complex response in the frequency domain.

$\mathbf{x} = (x_0, x_1)$ are the coordinates $R = R(\mathbf{x})$ is the complex valued Gabor filter response. $i = \sqrt{-1}$ is the imaginary unit.

$$\frac{\partial \varphi}{\partial \mathbf{x}} = \left(\frac{\partial \varphi}{\partial x_0}, \frac{\partial \varphi}{\partial x_1}\right) = \nabla \varphi \tag{A.1}$$

$$\frac{\partial R}{\partial \mathbf{x}} = \left(\frac{\partial R}{\partial x_0}, \frac{\partial R}{\partial x_1}\right) = \nabla R \tag{A.2}$$

The complex response and the conjugate complex of the response are expressed in Eq. (A.3) and Eq. (A.4) respectively.

$$R(\mathbf{x}) = \rho \exp(i\varphi) \tag{A.3}$$

$$R^*(\mathbf{x}) = \rho \exp(-i\varphi) \tag{A.4}$$

Eq. (A.5) shows the derivative of the complex response.

$$\frac{\partial R}{\partial \mathbf{x}} = \frac{\partial \rho}{\partial \mathbf{x}} \exp(i\varphi) + i\rho \exp(i\varphi) \frac{\partial \varphi}{\partial \mathbf{x}} \tag{A.5}$$

Multiplying the Eq. (A.5) with the complex conjugate of R we get:

$$R^* \frac{\partial R}{\partial \mathbf{x}} = \rho \frac{\partial \rho}{\partial \mathbf{x}} + i\rho^2 \frac{\partial \varphi}{\partial \mathbf{x}} \tag{A.6}$$

We are only interested in the imaginary parts of both sides of Eq. (A.6). The imaginary part of Eq. (A.6) contains the phase gradient multiplied with ρ^2.

$$\mathrm{Im}\left(R^* \frac{\partial R}{\partial \mathbf{x}}\right) = \rho^2 \frac{\partial \varphi}{\partial \mathbf{x}} \tag{A.7}$$

The phase gradient can be expressed in Eq. (A.8) with the gradient of the complex response and the complex conjugate of the response itself.

$$\frac{\partial \varphi}{\partial \mathbf{x}} = \frac{\mathrm{Im}(R^* \frac{\partial R}{\partial \mathbf{x}})}{\rho^2} = \frac{\mathrm{Im}(\frac{\partial R}{\partial \mathbf{x}})\mathrm{Re}(R) - \mathrm{Re}(\frac{\partial R}{\partial \mathbf{x}})\mathrm{Im}(R)}{\mathrm{Re}(R)^2 + \mathrm{Im}(R)^2} \tag{A.8}$$

An analogous result for 3-D filter responses was used in [8] in order to compute the optical flow in image sequences.

References

1. J. Bigün. Gabor phase in boundary tracking and region segregation. In *Proceedings of the International Conference on Digital Signal Processing*, pages 229–237, Nicosia, Cyprus, July 1993.

2. J. Bigün and J. M. H. du Buf. N-folded symmetries by complex moments in gabor space and their applications to unsupervised texture segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(1):80–87, January 1994.
3. J. Bigün, G. H. Granlund, and J. Wiklund. Multidimensional orientation estimation with applications to texture analysis and optical flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(8):775–790, August 1991.
4. C. H. Chen, L. F. Pau, and P. S. P. Wang, editors. *Handbook of Pattern Recognition & Computer Vision.* World Scientific, Singapore, 1993.
5. J. M. Coggins. *A Framework for Texture Analysis Based on Spatial Filtering.* PhD thesis, Computer Science Department, Michigan University, East Lansing, MI, 1982.
6. J.M.H. du Buf. Gabor phase in texture discrimination. *Signal Processing*, 21(3):221–240, November 1990.
7. R. A. Fisher. The use of multiple measurements in taxonomic problems. *Ann. Eugenics*, 7:179–188, 1936.
8. D. J. Fleet and A. D. Jepson. Computation of component image velocity from local phase information. *International Journal of Computer Vision*, 5(1):77–104, 1990.
9. D. J. Fleet and A. D. Jepson. Stability of phase information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(12):1253–1268, December 1993.
10. A. D. Jepson and D. J. Fleet. Phase singularities in scale-space. *Image and Vision Computing*, 9(5):338–343, October 1991.
11. H. Knutsson. *Filtering and reconstruction in image processing.* PhD thesis, Linköping University, 1982.
12. P. Schroeter and J. Bigün. Hierarchical image segmentation by multi-dimensional clustering and orientation adaptive boundary refinement. *Pattern Recognition*, 28(5):695–709, May 1995.
13. J. M. Tribolet. A new phase unwrapping algorithm. *IEEE Transaction On Acoustics, Speech, And Signal Processing*, ASSP-25(2):170–177, April 1977.
14. T. Y. Young and T. W. Calvert. *Classification, Estimation and Pattern Recognition.* Elsevier, 1974.