Fast Selective Detection of Rotational Symmetries Using Normalized Inhibition

Björn Johansson and Gösta Granlund

Computer Vision Laboratory
Department of Electrical Engineering
Linköping University, SE-581 83 Linköping, Sweden
{bjorn, gosta}@isy.liu.se
http://www.isy.liu.se/cvl/
Phone: +46 13 281302, +46 13 281303, Fax: +46 13 138526

Abstract. Perceptual experiments indicate that corners and curvature are very important features in the process of recognition. This paper presents a new method to efficiently detect rotational symmetries, which describe complex curvature such as corners, circles, star- and spiral patterns. The method is designed to give selective and sparse responses. It works in three steps; first extract local orientation from a gray-scale or color image, second correlate the orientation image with rotational symmetry filters and third let the filter responses inhibit each other in order to get more selective responses. The correlations can be made efficient by separating the 2D-filters into a small number of 1D-filters.
These symmetries can serve as feature points at a high abstraction level for use in hierarchical matching structures for 3D-estimation, object recognition, etc.

1 Introduction

Human vision seems to work in a hierarchical way in that we first extract low level features such as local orientation and color and then higher level features [7]. There also seem to exist lateral interactions between cells, perhaps to make them more selective. No one knows for sure what these high level features are but there are some indications that curvature, circles, spiral- and star patters are among them [10], [18]. And indeed, perceptual experiments indicate that corners and curvature are very important features in the process of recognition and one can often recognize an object from its curvature alone [19], [16]. They have a high degree of specificity and sparsity. As they are point features, they do not suffer from the aperture problem usually encountered for line and edge structures [8]. This paper describes a procedure to detect the features mentioned above. First a local orientation image is calculated. Second this image is correlated with a set of filters to detect complex curvature features. Finally a lateral inhibition procedure is used to make these filter responses more selective and sparse. These symmetries can serve as feature points at a high abstraction level for use in hierarchical matching structure for object recognition and estimation of 3D-structure. One examples is detection of traffic circles, crossroads (star shapes)

D. Vernon (Ed.): ECCV 2000, LNCS 1842, pp. 871–887, 2000.

and bends of the road (high curvature points) in aerial images which is useful in navigation of autonomous aircrafts. Another example concerning autonomous trucks is mentioned in the end of this paper.

2 Notations

$\mathbf{x} = (x_1, x_2)$, $\mathbf{y} = (y_1, y_2)$ denote Cartesian coordinate values, r and φ denote polar coordinate values and $\mathbf{u}$ denote Cartesian coordinates in the frequency domain. Remaining boldface letters, $\mathbf{z}$, $\mathbf{a}$, $\mathbf{b}$ etc., denote real or complex functions. $\mathbf{z}(\mathbf{x})$ means the value at pixel $\mathbf{x}$ and $\mathbf{z}$ means the matrix containing the function values at all pixels.

Products $\mathbf{d} = \mathbf{ab}$ and divisions $\mathbf{e} = \mathbf{a}/\mathbf{b}$ between matrices are exclusively defined to be pointwise operators, i.e. $\mathbf{d}(\mathbf{x}) = \mathbf{a}(\mathbf{x})\mathbf{b}(\mathbf{x})$, $\mathbf{e}(\mathbf{x}) = \mathbf{a}(\mathbf{x})/\mathbf{b}(\mathbf{x})$.

$$\text{Inner scalar product:} \quad \langle \mathbf{b}, \mathbf{f} \rangle = \sum_{\mathbf{y} \in \mathbb{Z}^2} \mathbf{b}^*(\mathbf{y})\mathbf{f}(\mathbf{y}) \tag{1}$$

$$\text{Correlation:} \quad (\mathbf{b} \star \mathbf{f})(\mathbf{x}) = \sum_{\mathbf{y} \in \mathbb{Z}^2} \mathbf{b}^*(\mathbf{y})\mathbf{f}(\mathbf{y} - \mathbf{x}) \tag{2}$$

where * denote complex conjugate. $\mathbf{b} \star \mathbf{f}$ is then also an image (assumed cut to the same size as $\mathbf{f}$). (Do not confuse this with convolution, $\mathbf{b} * \mathbf{f}$, where the filter $\mathbf{b}$ is mirrored instead of conjugated.)

3 Local Orientation

The classical representation of local orientation is simply a 2D-vector pointing in the dominant direction with a magnitude that reflects the orientation dominance (e.g. the energy in the dominant direction). An example of this is the image gradient.

A better representation is the *double angle representation*, where we have a vector pointing in the double angle direction, i.e. if the orientation has the direction θ we represent it with a vector pointing in the 2θ-direction. Figure1 illustrates the idea. The magnitude still represents our confidence in the dominant orientation

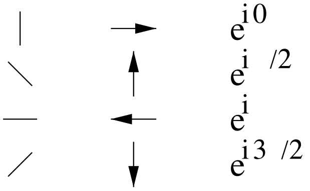


Fig. 1. Local orientations (left) and corresponding double angle descriptors as vectors (middle) and as complex numbers (right)

and that is sometimes referred to as a *certainty measure* (often denoted $\mathbf{c}$). The concept double angle representation was first mentioned in [17]. It has at least two advantages:

- We avoid ambiguities in the representation: It does not matter if we choose to say that the orientation has the direction θ or, equivalently, $\theta + \pi$. In the double angle representation both choices get the same descriptor angle 2θ (modulo 2π).
- Averaging the double angle orientation description field makes sense. One can argue that two orthogonal orientations should have maximally different representations, e.g. vectors that point in opposite directions. This is for instance useful in color images and vector fields when we want to fuse descriptors from several channels into one description.

This descriptor will in this paper be denoted by a complex number $\mathbf{z}$ where $\hat{\mathbf{z}} = \mathbf{z}/|\mathbf{z}|$ points out the dominant orientation and $\mathbf{c} = |\mathbf{z}|$ indicates the certainty or confidence in the orientation identified.
A definition before we continue:

Definition 1. *A* simple signal $I : \mathbb{R}^n \to \mathbb{R}$ *is defined as* $I(\mathbf{x}) = f(\mathbf{x} \cdot \hat{\mathbf{m}})$ *where* $f : \mathbb{R} \to \mathbb{R}$ *and* $\hat{\mathbf{m}}$ *is a fix vector* $\in \mathbb{R}^n$.

Figure 2 shows some examples of simple and non-simple signals. The magnitude of the double angle descriptor is usually chosen to be proportional to the difference between the signal energy in the dominant direction and the energy in the orthogonal direction. In this case the double angle descriptor cannot distinguish between a weak (low energy) simple signal and a strong non-simple signal (e.g. noise with a slightly dominant orientation). We will get a low magnitude $\mathbf{c} = |\mathbf{z}|$ (indicating no dominant orientation) in both cases which sometimes may be undesirable.

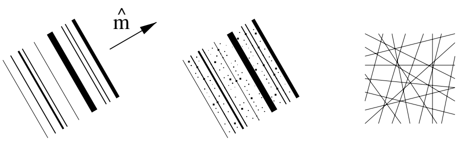


Fig. 2. Examples of a simple signal (left) and non-simple signals (middle and right)

Another, more comprehensive, representation of local orientation is the concept of *tensors* [8]. A 2D tensor is a 2 × 2 tensor (matrix) $\mathbf{T} = \lambda_1 \hat{\mathbf{e}}_1 \hat{\mathbf{e}}_1^T + \lambda_2 \hat{\mathbf{e}}_2 \hat{\mathbf{e}}_2^T$ where λ_1 denote the energy in the dominant orientation direction $\hat{\mathbf{e}}_1$ and λ_2 the energy in the orthogonal direction $\hat{\mathbf{e}}_2$ (by this definition we get $\lambda_1 \geq \lambda_2$ and for a simple signal $\lambda_2 = 0$). From this tensor we can calculate a double angle descriptor e.g. by

$$\mathbf{z} = \left(\frac{\lambda_1 - \lambda_2}{\lambda_1} \right) \left(1 - e^{-\|\mathbf{T}\|/T_0} \right) e^{i2\angle \hat{\mathbf{e}}_1} \tag{3}$$

This detour (gray-level image ⇒ tensor ⇒ double angle descriptor) seems unnecessary, but in this way we can control the selectivity for simple/non-simple signals and the energy sensitivity separately (for instance a weak simple signal can now get a higher magnitude $|\mathbf{z}|$ than a strong non-simple signal). We can also note that the magnitude lies in the interval $[0, 1]$, with $|\mathbf{z}| = 1$ indicating high confidence and $|\mathbf{z}| = 0$ low confidence in the dominant orientation.
There are different ways to compute the tensor in practice. One way is to look at the local energy in different directions using a set of quadrature filters and weigh them together in a proper way [8]. The filtering can be efficiently computed by use of sequential filter trees [4]. Another way to compute a tensor is to approximate (by weighted least square) the local neighborhood $f(\mathbf{x})$ with a polynomial of e.g. second degree

$$f(\mathbf{x}) \sim \mathbf{x}^T\mathbf{A}\mathbf{x} + \mathbf{b}^T\mathbf{x} + c \tag{4}$$

and from the coefficients create a tensor description

$$\mathbf{T} = \mathbf{A}\mathbf{A}^T + \gamma\mathbf{b}\mathbf{b}^T \tag{5}$$

(where γ is a non-negative weight factor between the linear and quadratic terms). This local polynomial approximation can be done by means of convolutions which in turn can be made separable leading to a very efficient algorithm, see [2]. The second method is at present a little faster and is the one used in this paper. To compute all local orientations in 9×9 neighborhoods in a 256×256 gray-level image takes in MATLAB a couple of seconds on a 299MHz SUN Ultra 10.

It is assumed from now on that we have access to orientation transform images in the double angle representation, denoted $\mathbf{z}$.

4 Rotational Symmetries

4.1 Basics of Rotational Symmetries

Figure 3 contain two image patterns together with their corresponding orientation descriptions in double angle representation (the orientation description is only a sketch - the certainty $\mathbf{c}$ is high in all areas where we have an approximately simple signal and fairly high signal energy). The circle can be described in the orientation transform as $\mathbf{z} = \mathbf{c}_{circle}e^{i2\varphi}$ where $\mathbf{c}_{circle} = |\mathbf{z}|$. Thus if we want to detect circles we could simply correlate the orientation image with the filter

$$\mathbf{w}(r,\varphi) = \mathbf{a}(r)e^{i2\varphi} \quad \text{where} \quad \mathbf{a}(r), \text{ e.g.,} = \begin{cases} 1 & \text{if } r < R \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

The correlation at $\mathbf{x} = \mathbf{0}$ becomes

$$(\mathbf{w} \star \mathbf{z}_{circle})(\mathbf{0}) = \langle \mathbf{w}, \mathbf{z}_{circle} \rangle = \langle \mathbf{a}e^{i2\varphi}, \mathbf{c}_{circle}e^{i2\varphi} \rangle = \langle \mathbf{a}, \mathbf{c}_{circle} \rangle \tag{7}$$

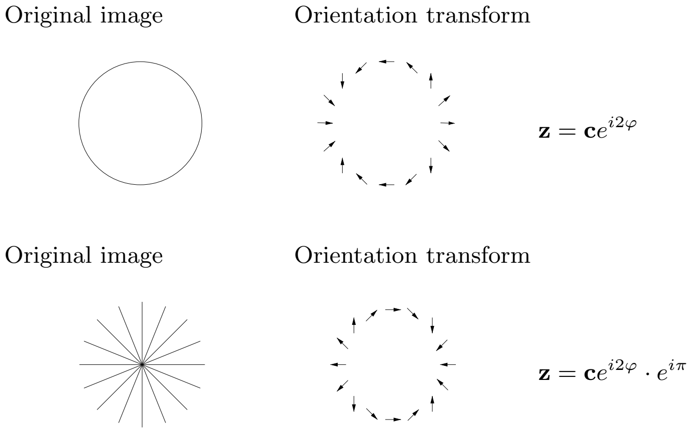

Original image
Orientation transform
$\mathbf{z} = \mathbf{c}e^{i2\varphi}$
Original image
Orientation transform
$\mathbf{z} = \mathbf{c}e^{i2\varphi} \cdot e^{i\pi}$

Fig. 3. Two patterns, a circle and a star, together with their corresponding orientation description in double angle representation

The correlation at $\mathbf{x} \neq \mathbf{0}$ will be complex but give magnitudes less than $\langle \mathbf{a}, \mathbf{c}_{circle} \rangle$. It is interesting to note that the same filter also can be used to detect star patterns. The phase of the double angle description of a circle and a star only differs by a constant π (see Fig. 3). Therefore if we correlate the circle filter with an orientation image describing a star pattern we get

$$(\mathbf{w} \star \mathbf{z}_{star})(\mathbf{0}) = \langle \mathbf{w}, \mathbf{z}_{star} \rangle = \langle \mathbf{a}e^{i2\varphi}, \mathbf{c}_{star}e^{i2\varphi}e^{i\pi} \rangle = e^{i\pi}\langle \mathbf{a}, \mathbf{c}_{star} \rangle \tag{8}$$

($\mathbf{c}_{circle}$ and $\mathbf{c}_{star}$ does not have to be equal).
Consequently the filter can detect, and distinguish, between a whole class of patterns. The phase of $\langle \mathbf{w}, \mathbf{z} \rangle$ represents the class membership and the magnitude indicates the confidence in this membership.

The filter described above is only one of many in a family of filters called *rotational symmetry filters*:

n:th order rotational symmetry filter

$$\mathbf{w}_n(r, \varphi) = \mathbf{a}(r)\mathbf{b}_n(\varphi) \quad \text{where} \quad \mathbf{b}_n(\varphi) = e^{in\varphi} \tag{9}$$

$\mathbf{a}(r)$ is called the *applicability function* and can be thought of as a window for the basis function $\mathbf{b}_n$ which is called *circular harmonic function*. The n:th order filter detects the class of patterns that can be described by $\mathbf{c}e^{i(n\varphi+\alpha)}$ (they are said to have the angular modulation speed n). Each class member is represented by a certain phase α. One can also think of the filter responses as components in a polar Fourier series of local areas in the orientation image.
The curious reader might wonder what these patterns look like. To answer this one has to start with the orientation image $\mathbf{c}e^{i(n\varphi+\alpha)}$ and go backwards to the original image, $\mathbf{I}$, from which the orientation image was calculated. This 'inverse'

is of course not unambiguous, but we get a hint by making the assumption that the image gradient is parallel to the dominant orientation:

$$\nabla\mathbf{I} = \pm|\nabla\mathbf{I}|e^{i(n\varphi+\alpha)/2} \tag{10}$$

(We have to divide the phase by two to get rid of the double angle representation. The price is the direction unambiguity.) This will give a differential equation that can be solved assuming polar separability. The solution (not derived here) becomes:

$$\mathbf{I}(r,\varphi) = \begin{cases} Cr^{1-n/2}\cos((\frac{n}{2}-1)\varphi+\frac{\alpha}{2}) & n \neq 2 \\ Cr^{1-n/2}e^{(1-n/2)\tan(\alpha/2)\varphi} & n = 2 \end{cases} \tag{11}$$

One way to visualize this is to plot level curves $(1+\cos(\omega\mathbf{I}))/2$ (idea from [13]). These are also a solution to the differential equation. Figure 4 contains a sample of these psychedelic patterns. The most useful rotational symmetry filters (i.e. most common visual patterns in our daily lives) are the

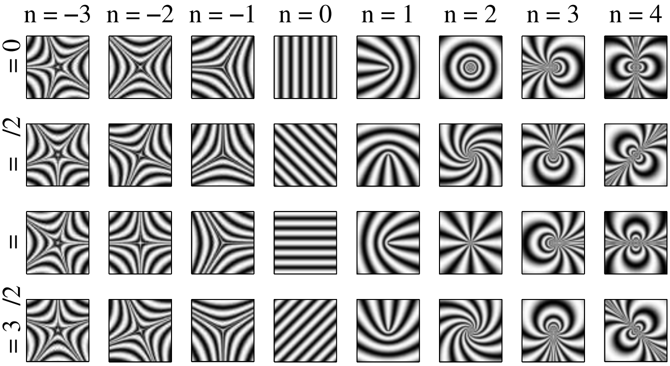


Fig. 4. Corresponding gray-level patterns to some rotational symmetries $e^{i(n\varphi+\alpha)}$

- *zeroth order*, n = 0: Detects lines (simply an averaging of the orientation image).
- *first order*, n = 1 (also called parabolic symmetries, referring to the gray-level patterns the filter are matched to): Detects curvature, corners and line-endings. This is not an optimal corner detector but it is rather robust and will give a high response to a variety of corner angles. The direction of the corner equals the filter output phase.
- *second order*, n = 2 (also called circular symmetries): Detects stars, spiral and circular-like patterns.

The three basis functions $\mathbf{b}_0$, $\mathbf{b}_1$ and $\mathbf{b}_2$ are shown in Fig. 5 below.
The rotational symmetry filters were invented around 1981 by Granlund and Knutsson and have been mentioned previously in literature [8], [12], [15], [13], [6].

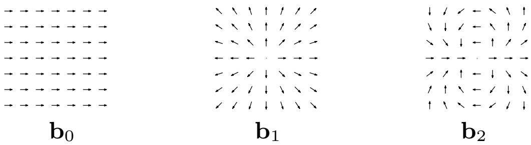


Fig. 5. The three most useful circular harmonic functions, $\mathbf{b}_n = e^{in\varphi}$, plotted in vector representation

4.2 Choice of Applicability Function

The choice of applicability function **a** depends on the application. We can for instance choose a Gaussian function or a table like the one in (6) above. These have one drawback though. In a general image we usually have more than one event (pattern) and we only want to detect one at a time in order to avoid interferences. In the case of rotational symmetries we measure the angular variation. This can differ from one radius to another. One could therefore argue that we should look at one fix radius at a time.
As an example look at the bicycle wheel in Fig. 6. It consists of two events; a star pattern at a small radius and a circle pattern at a larger radius. If we calculate an orientation image of the wheel and apply the circular symmetry filter $\mathbf{w}_2 = \mathbf{a}(r)e^{i2\varphi}$ with the leftmost applicability in Fig. 7 we will get a high filter output at the center of the wheel with the phase π, representing a star shaped pattern. If we instead take the rightmost ring-shaped applicability which mainly 'sees' the circle, we get a high filter output with phase 0. If we on the other hand use the middle applicability which 'sees' both events they will probably cancel each other out somewhat giving a low filter output with random phase. Differently argued one can also say that we are measuring the change in the φ-dimension and should therefore hold the other dimensions (the radius) fairly constant.[1] From this we can conclude that the descriptor is dependent upon scale, like in the description of most other properties. Thus an applicability function localized around a certain radius seems to be a better choice in the case of

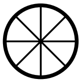


Fig. 6. Bicycle wheel

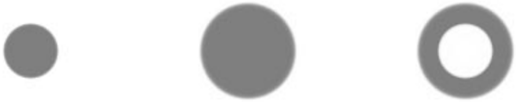


Fig. 7. Three applicabilities $\mathbf{a}(r)$

[1] Also compare with derivation: If you e.g. want to calculate $\partial\mathbf{I}/\partial x$ with a filter it is probably a good idea to make the filter fairly narrow in the y-direction.

multi-events (multi-scales).

What is the best shape of the applicability? We can for instance choose the rightmost ring-shaped applicability in Fig. 7. However, a more fuzzy applicability like the one in Fig. 8, giving smoother and better controlled responses, will probably be a better choice. The applicability in this figure is a simple difference of Gaussians. This applicability has a further advantage in that it can be separated into a few 1D-filters, see Sec. 7.

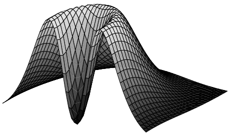
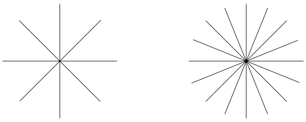


Fig. 8. Cross-section of an smooth applicability localized around a center radius

5 Normalization

Look at the two star patterns in Fig. 9. If we make orientation images for these two stars and correlate with the circular symmetry filter $\mathbf{w}_2 = \mathbf{a}(r)e^{i2\varphi}$ we will in both cases get a high filter response with phase π. The response for the left star will however be lower than the response for the right star simply because we have less orientation information in the first case, i.e. $\langle \mathbf{a}, \mathbf{c}_{star1} \rangle < \langle \mathbf{a}, \mathbf{c}_{star2} \rangle$. This may not be desirable. If we want to detect star shapes we do not care if the star has eight or twenty points, if it is composed of thick or thin lines and so on. Similar problems occur for other patterns.



$$|\langle \mathbf{w}_2, \mathbf{z}_{star1} \rangle| \quad < \quad |\langle \mathbf{w}_2, \mathbf{z}_{star2} \rangle|$$

Fig. 9. Two star patterns

This problem can be solved by *normalization.* We normalize the filter response with the amount of information we have available, weighted with the applicability (remember that the division is a pointwise operator): [2]

[2] Normalization can also be seen as a projection of $\hat{\mathbf{z}}$ onto the basis function $\mathbf{b}_n$ in a Hilbert space with the (weighted) inner product $\langle \mathbf{b}, \hat{\mathbf{z}} \rangle_{\mathbf{ac}} = \sum \mathbf{c}_k \mathbf{a}_k \hat{\mathbf{z}}_k \mathbf{b}^*_{n,k} = \langle \mathbf{a}\mathbf{b}_n, \mathbf{c}\hat{\mathbf{z}} \rangle$

Normalization of filter responses

$$\mathbf{s}_n = \frac{(\mathbf{c}\hat{\mathbf{z}}) \star (\mathbf{a}\mathbf{b}_n)}{\mathbf{c} \star \mathbf{a}} \tag{12}$$

$\mathbf{s}_n(\mathbf{x})$ has the useful property $|\mathbf{s}_n(\mathbf{x})| \leq 1$. If the orientation $\hat{\mathbf{z}}$ is consistent with the filter basis function $\mathbf{b}_n$ we will get $|\mathbf{s}_n| = 1$ independently of the amount of orientation information (except in the degenerated case $\mathbf{c} \equiv 0$). If the total orientation information is too low we cannot rely on the result $\mathbf{s}_n$. In the orientation image we had a certainty measure $\mathbf{c}$ indicating how well we can rely on the orientation information $\hat{\mathbf{z}}$. We can use the same formalism for rotational symmetry images, call it symmetry certainty $\mathbf{c}_s$, indicating how well we can rely on $\mathbf{s}_n$. For example we can use

Symmetry certainty

$$\mathbf{c}_s = m\left(\frac{\mathbf{c} \star \mathbf{a}}{\mathbf{1} \star \mathbf{a}}\right) \tag{13}$$

where $\mathbf{1}$ should be interpreted as an image with the constant value one, i.e. $\mathbf{1} \star \mathbf{a}$ equals the sum of all coefficients in $\mathbf{a}$ except near the image border, and m is a mapping function serving as a fuzzy threshold, e.g.

$$m_{\alpha,\beta}(x) = \frac{((1-\alpha)x)^{\beta}}{((1-\alpha)x)^{\beta} + (\alpha(1-x))^{\beta}} \tag{14}$$

α determines the threshold level and β the 'fuzziness'. Figure 10 illustrates the mapping function for some different values α, β. In this paper however we will use the simple case $\alpha = 0.5$ and $\beta = 1$ giving $m_{\alpha,\beta}(x) = x$ and $\mathbf{c}_s = \frac{\mathbf{c}\star\mathbf{a}}{\mathbf{1}\star\mathbf{a}}$, i.e. $\mathbf{c}_s$ is a weighted averaged input certainty.

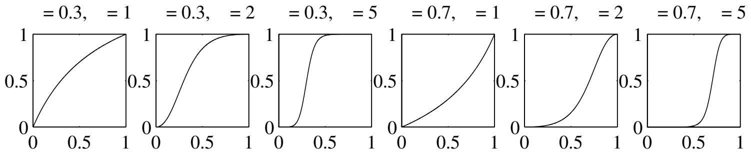


Fig. 10. The mapping function $m_{\alpha,\beta}$ for some values α, β

because $\langle \mathbf{b}_n, \mathbf{b}_n \rangle_{\mathbf{ac}} = \langle \mathbf{a}, \mathbf{c} \rangle$, where $\langle .,. \rangle$ is the normal scalar product defined in Sec. 2 above.

This is actually a special case of normalized convolution [2], [11], [9] where you project $\hat{\mathbf{z}}$ on a subspace spanned by several basis functions (this require knowledge about dual basis theory).

6 Normalized Mutual Inhibition

Figure 11, 12 and 13 illustrates the procedure *gray level image ⇒ orientation image ⇒ rotational symmetry images* (the choice of applicability is not critical in this case, the scale is about the same size as the circle). Figure 11 shows the test image and corresponding orientation image. Figure 12 shows the normalized responses $\mathbf{s}_n$, $n = 0, 1, 2$ and their certainty $\mathbf{c}_s = \frac{\mathbf{c} \star \mathbf{a}}{\mathbf{1} \star \mathbf{a}}$. In Fig. 13 the normalized responses are weighted with the certainty. With this simple choice of $\mathbf{c}_s$ we get

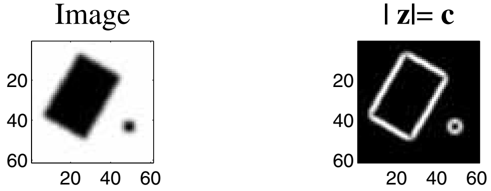


Fig. 11. Left: Gray-level test image; a rectangle and a circle. **Right:** Magnitude of the orientation image (white indicates high magnitude)

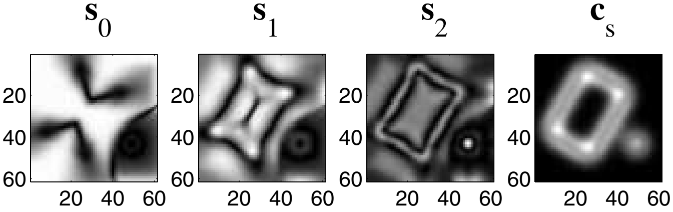


Fig. 12. Magnitude of normalized filter responses $\mathbf{s}_n, n = 0, 1, 2$. **Rightmost:** Symmetry certainty $\mathbf{c}_s = \frac{\mathbf{c} \star \mathbf{a}}{\mathbf{1} \star \mathbf{a}}$

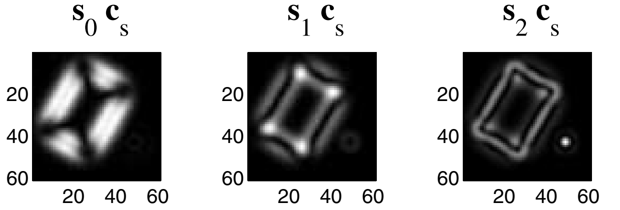


Fig. 13. Magnitude of normalized filter responses $\mathbf{s}_n, n = 0, 1, 2$ weighted with the certainty $\mathbf{c}_s$ (equivalent to $\mathbf{w}_n \star \mathbf{z}$)

$$\mathbf{s}_n \mathbf{c}_s = \frac{(\mathbf{c}\hat{\mathbf{z}}) \star (\mathbf{a}\mathbf{b}_n)}{\mathbf{c} \star \mathbf{a}} \cdot \frac{\mathbf{c} \star \mathbf{a}}{\mathbf{1} \star \mathbf{a}} = \frac{(\mathbf{c}\hat{\mathbf{z}}) \star (\mathbf{a}\mathbf{b}_n)}{\mathbf{1} \star \mathbf{a}} = (\mathbf{c}\hat{\mathbf{z}}) \star \left(\frac{\mathbf{a}}{\mathbf{1} \star \mathbf{a}} \mathbf{b}_n \right) = (\mathbf{c}\hat{\mathbf{z}}) \star (\hat{\mathbf{a}}\mathbf{b}_n) \tag{15}$$

where $\hat{\mathbf{a}} = \frac{\mathbf{a}}{\mathbf{1} \star \mathbf{a}}$ is a normalized applicability. This is the same as if we had filtered without normalization (except for $\hat{\mathbf{a}}$ instead of $\mathbf{a}$). As we soon will see it is still useful to divide the result into the terms $\mathbf{s}_n$ and $\mathbf{c}_s$.

Take one more look at Fig. 13. As you can see the first order symmetry filter $\mathbf{s}_1$ detect corners and the second order filter $\mathbf{s}_2$ detect circles but they also give some response to other patterns than we might wish them to do. This is because we do not have orientation information in the whole applicability window. For instance a line in the outer field of the applicability window might as well be a piece of a circle if you do not have contradictory orientation information in the rest of the window, a corner is almost the same as half a circle etc.

We can make the filters more selective by letting the normalized responses inhibit (or punish) each other (it is much easier to inhibit with the normalized responses compared to the unnormalized because we know that $|\mathbf{s}_n| \leq 1$):

Normalized inhibition

$$\check{\mathbf{s}}_n = \mathbf{s}_n \prod_{k \neq n} (1 - m_k(|\mathbf{s}_k|)) \tag{16}$$

where m_k can be some fuzzy threshold function controlling the power in the inhibition, e.g. the one in (14). If one filter output is high the others will get low. This idea is inspired from lateral inhibition in biological systems and is a well known concept but has never before been used to make rotational symmetries more selective.

Figure 14 illustrates the result after inhibition (m_k is ignored here). The inhibited responses are weighted with the certainty $\mathbf{c}_s$. The inhibited first and second symmetry responses, $\check{\mathbf{s}}_1$ and $\check{\mathbf{s}}_2$, are the most interesting ones (but we still need to calculate $\mathbf{s}_0$ for use in the inhibition procedure).

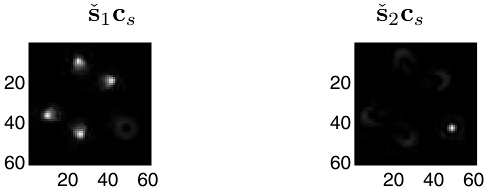


Fig. 14. Inhibited rotational symmetry responses

6.1 Comparison to Previous Methods

A previous solution to achieve better selectivity is described in a patent from 1986 [14] and is used in a commercial image processing program by ContextVision[3]. The patent describes a method termed *consistency operation*. The name refers

[3] http://www.contextvision.se/

to that the operator should only respond to signals that are consistent with the signal model. The method is based on a combination of four correlations:

$$\mathbf{h}_1 = \mathbf{w}_n \star \mathbf{z} \quad , \quad \mathbf{h}_2 = \mathbf{w}_n \star \mathbf{c} \quad , \quad \mathbf{h}_3 = \mathbf{a} \star \mathbf{z} \quad , \quad \mathbf{h}_4 = \mathbf{a} \star \mathbf{c} \tag{17}$$

The filter results are combined into

$$\mathbf{h} = \frac{\mathbf{h}_4\mathbf{h}_1 - \mathbf{h}_2\mathbf{h}_3}{\mathbf{h}_4^{\gamma}} \tag{18}$$

where the denominator is an energy normalization controlling the model versus energy dependence of the algorithm. With $\gamma = 1$ the output magnitude varies linearly with the magnitude of the input signal. Decreasing the value increases the selectivity. The result from this operation is called *divcons* when $n = 1$ and *rotcons* when $n = 2$.
The result when applying this method (with $\gamma = 1$) on the orientation image in Fig. 11 is shown in Fig. 15. The divcons result can be fairly comparable to $\check{\mathbf{s}}_1\mathbf{c}_s$ in this case but the rotcons result is much less selective than $\check{\mathbf{s}}_1\mathbf{c}_s$. The consistency operation has fairly the same behavior as an inhibition with only the zeroth order symmetry response (line patterns). This means for instance that the corners are not inhibited from the second order response but are instead detected as 'half circles'. In general, using normalized inhibition instead of the consistency operation produces more selective and sparse responses.

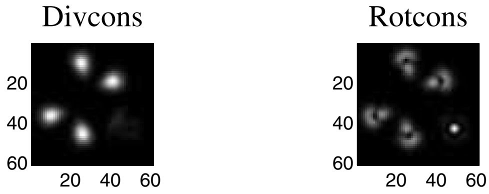


Fig. 15. Increase of selectivity using the consistency operation

7 Separable Filters

On the subject of computational complexity we can note that we have to compute four correlations in the normalized inhibition procedure:

$$(\mathbf{w}_0 \star \mathbf{z}) \quad , \quad (\mathbf{w}_1 \star \mathbf{z}) \quad , \quad (\mathbf{w}_2 \star \mathbf{z}) \quad \text{and} \quad (\mathbf{a} \star |\mathbf{z}|) = (\mathbf{w}_0 \star |\mathbf{z}|) \tag{19}$$

The filters have to be large if the want to find large patterns and the correlations becomes computationally demanding. However there are much redundancy in the filters and they can be separated into a few 1D-filters using SVD (Singular Value Decomposition). A filter kernel $\mathbf{w}$ of size $N \times N$ can be composed into three $N \times N$ matrices, $\mathbf{w} = \mathbf{U\Sigma V}^T = \sum_{k=1}^{N} \sigma_k \mathbf{u}_k \mathbf{v}_k^T$ where $\mathbf{u}_k$ and $\mathbf{v}_k$ are the k:th column in $\mathbf{U}$ resp. $\mathbf{V}$, $\mathbf{UU}^T = \mathbf{VV}^T = \mathbf{I}$ and $\mathbf{\Sigma} = diag(\sigma_1, \sigma_2, ...)$ (where $\sigma_1 \geq \sigma_2 \geq ...$). If only a few σ_k, say M values, is high we can approximate the filter kernel by M terms in the sum, $\mathbf{w} \approx \tilde{\mathbf{w}} = \sum_{k=1}^{M} \sigma_1 \mathbf{u}_k \mathbf{v}_k^T$, which corresponds to $2M$ 1D-filters.

Example 1. Let $N = 21$ and use a difference of Gaussians as applicability (as in Fig. 8):

$$\mathbf{a}(r) = e^{-\frac{r^2}{\sigma^2}} - e^{-\frac{r^2}{\sigma^2 \delta^2}} \quad , \quad \sigma = r_0 \sqrt{\frac{1 - 1/\delta^2}{\ln \delta^2}} \tag{20}$$

where r_0 is the center radius and $0 < \delta < 1$ is a user parameter controlling the thickness of the applicability. If we choose $r_0 = 3$ and $\delta = 0.5$ and compute the SVD for $\mathbf{w}_0 = \mathbf{a}(r)$, $\mathbf{w}_1 = \mathbf{a}(r)e^{i\varphi}$ and $\mathbf{w}_2 = \mathbf{a}(r)e^{2i\varphi}$ respectively we get

$$\begin{aligned} diag(\mathbf{S}_0) &= (3.6124, 0.8469, 0.0000, 0.0000, 0.0000, ...) \\ diag(\mathbf{S}_1) &= (2.6191, 2.6191, 0.1535, 0.1535, 0.0177, ...) \\ diag(\mathbf{S}_2) &= (2.6199, 1.8552, 1.8552, 0.1402, 0.0056, ...) \end{aligned} \tag{21}$$

It is thus sufficient to approximate $\mathbf{w}_0$ with two terms (four 1D-filters), $\mathbf{w}_1$ with two terms (four 1D-filters) and $\mathbf{w}_2$ with three terms (six 1D-filters) and the relative error $\frac{\|w_n - \tilde{w}_n\|}{\|w_n\|}$ is about 5% for both $\mathbf{w}_1$ and $\mathbf{w}_2$. The reader can verify that if we had chosen a Gaussian applicability instead of a difference of Gaussians, we would have to use more terms than above to get a good approximation (except for $\mathbf{w}_0$ which can be separated into only two 1D-filters).

8 Experiments

This section illustrates how the symmetry filters can be use with two examples. Space limitations do not allow any detailed explanations.

8.1 Experiment 1: Autonomous Truck

Figure 16 shows an image of a pallet together with the magnitude (orientation certainty) of its orientation image. This is one type of image used in a robotics project, where the goal is to have an autonomous truck locate the pallet and pick it up. The orientation image was down-sampled before correlation with the symmetry filters (using a Gaussian filter with std = 1.2). The inhibited first and second order symmetries are shown in Fig. 17. The size of the filter kernel is 50×50 with a center radius 6 (the separable filter technique described above was used). The intensity represents the magnitude and the vectors represent the value at the largest maxima points. There are a large number of responses and the output looks complex, which is due to the fact that the input images are indeed complex.

The second order symmetry filter gives characteristic blobs with zero phase on the pallet indicating circular-like patterns (in this case squares). The circle patterns can be extracted from the second order response by e.g. taking the real part and ignoring negative values, see Fig. 18. The pallet has a characteristic signature of three large blobs and some smaller in between, see Fig. 19. The whole procedure from the gray-level image to the circle image took about 6 seconds on a 299MHz SUN Ultra 10. The circle image can be further processed using Hough transform to find 'lines' formed by the blobs and the correct line (the one in Fig. 18) can be found by testing against the characteristic signature hypothesis. In spite of the complexity of the image, the linear set of circular structures turns out to be quite unique. When the pallet is found in the image it is possible to calculate pallet direction, distance and orientation in the real world using the information in the symmetry image and the local orientation image.

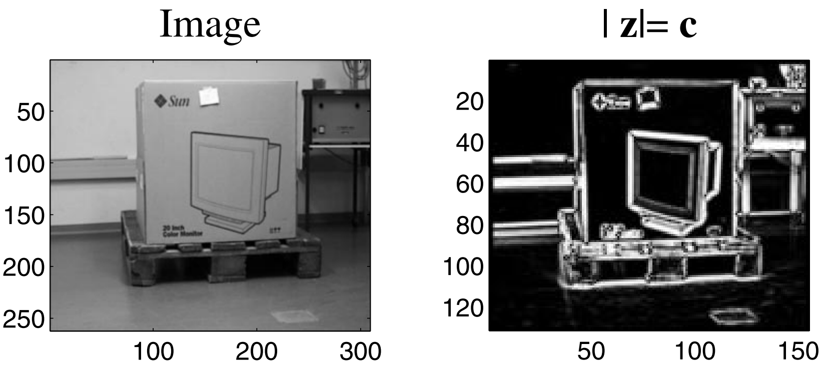
This technique has been tested on 28 images containing pallets. The distance between the camera and the pallet varied in the range of 1 to 16 meters (the symmetries was detected in several scales) and the orientation of the pallet relative to the camera varied between 0 to 40 degrees. The result was promising but the number of test images is too small to reach a final conclusion. Details about this project can be found in [3].

8.2 Experiment 2: Aerial Image

Another example is generation of features for use in navigation of autonomous aircrafts. The features can be used in template matching or local histogram matching to help the aircraft find landmark pbjects to establish its position. Figure 20 contains an aerial image and its orientaton magnitude. The inhibited first order symmetry is shown in Fig. 21 (as before, the magnitude represents the intensity and the vectors represent the complex value at the largest maxima points). The first order symmetry detects the corners in the crossroad and curvature in general. The second order symmetry can be used to find traffic circles, crossroads, houses, and other circular-like patterns. Detected circular-like patterns (c.f. Fig. 18) in two different scales is shown in Fig. 22. The rotational symmetry features is more robust to illumination and seasonal variations than the original gray-level image which makes them suitable in a matching process. In spite of the apparent complexity, the particular pattern of such features turns out to be quite specific. Such a higher level matching is likely to take place in human vision as well. This experiment is part of the WITAS-project, see [1], [5]. Future work includes multiscale detection and testing of robustness and invariance to viewpoint, seasonal variances, illumination etc.

9 Acknowledgment

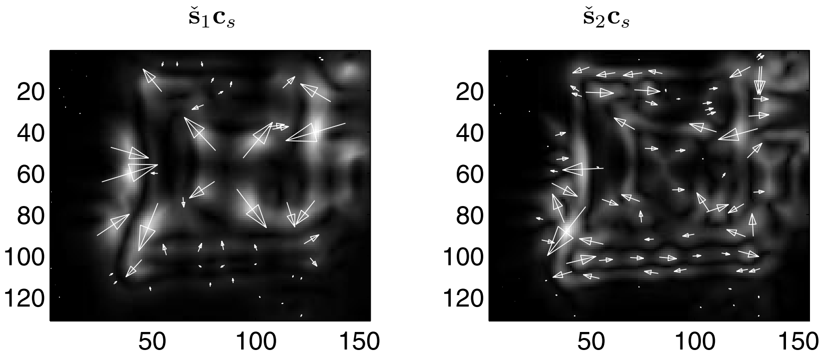
This work was supported by the Foundation for Strategic Research, project VISIT - VIsual Information Technology. The authors would like to thank the staff at Computer Vision Laboratory for constructive discussions.



Image

| z|= c

Fig. 16. An image of a pallet (left) and its local orientation magnitude (right)


$\check{\mathbf{s}}_1\mathbf{c}_s$

$\check{\mathbf{s}}_2\mathbf{c}_s$

Fig. 17. Inhibited rotational symmetry responses. *Left*: first order, *Right*: second order

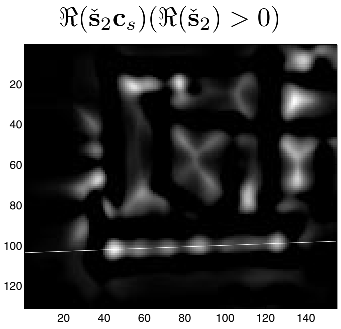

$\Re(\check{\mathbf{s}}_2\mathbf{c}_s)(\Re(\check{\mathbf{s}}_2) > 0)$

Fig. 18. Circular-like patterns extracted from second order symmetry. The circle features forms a line at the pallet position

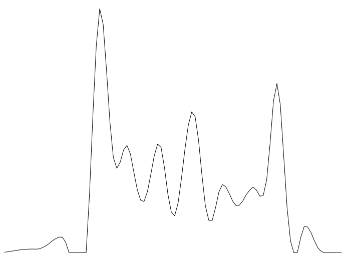


Fig. 19. Characteristics along a line in Fig. 18 fullfilling structural hypothesis for the desired object

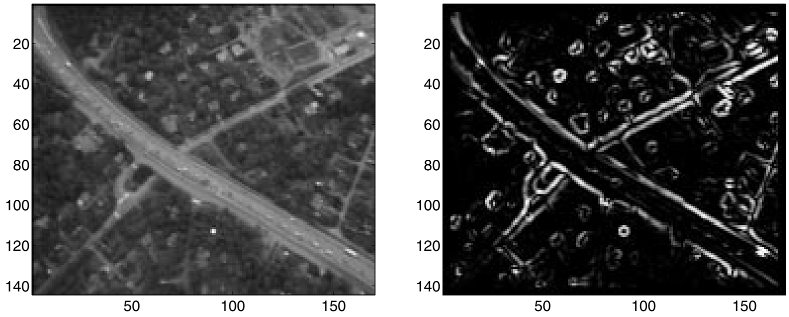


Fig. 20. An aerial image (left) and its local orientation magnitude (right)

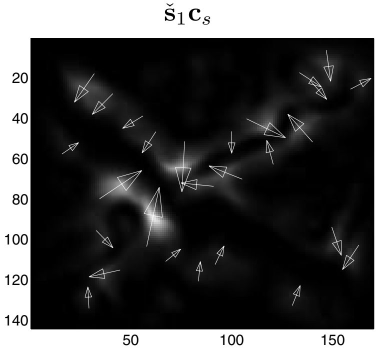

$\check{\mathbf{s}}_1 \mathbf{c}_s$

Fig. 21. Inhibited first order symmetry respons with center radius $r_0 = 15$

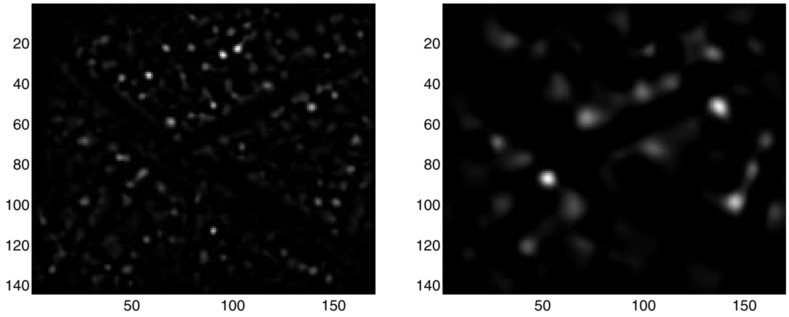


Fig. 22. Circular-like patterns extracted from second order symmetry using center radius $r_0 = 5$ (left) and $r_0 = 15$ (right)

References

[1] WITAS web page: http://www.ida.liu.se/ext/witas/.

[2] Farnebäck, G.: Spatial Domain Methods for Orientation and Velocity Estimation. Lic. Thesis LiU-Tek-Lic-1999:13, Dept. EE, Linköping University, SE-581 83 Linköping, Sweden, March 1999. Thesis No. 755, ISBN 91-7219-441-3.

[3] Roll, Jacob: A System for Visual-Based Automated Storage Robots. Master's thesis, Linköping University, 1999. LiTH-ISY-EX-2053. URL: http://www.isy.liu.se/cvl/ScOut/Masters/Masters.html.

[4] Andersson, M., Wiklund, J., Knutsson, H.: Sequential Filter Trees for Efficient 2D 3D and 4D Orientation Estimation. Report LiTH-ISY-R-2070, ISY, SE-581 83 Linköping, Sweden, November 1998. URL: http://www.isy.liu.se/cvl/ScOut/TechRep/TechRep.html.

[5] Andersson, T., Granlund, G. H., Farnebäck, G., Nordberg, K., Wiklund, J.: WITAS project at Computer Vision Laboratory - A status report (Jan 98). In SSAB 98 Symposium on image analysis, 113–116, Uppsala, Sweden, March 1998.

[6] Bigün, J.: Pattern Recognition in Images by Symmetries and Coordinate Transformations. Computer Vision and Image Understanding, 68(3):290–307, 1997

[7] Bear, M. F., Connors, B. W., Paradiso, M. A.: Neuroscience: Exploring the Brain. Williams & Wilkins, 1996, ISBN 0-683-00488-3.

[8] Granlund, G. H., Knutsson, H.: Signal Processing for Computer Vision. Kluwer Academic Publishers, 1995. ISBN 0-7923-9530-1.

[9] Westin, C-F.: A Tensor Framework for Multidimensional Signal Processing. PhD thesis, Linköping University, SE-581 83 Linköping, Sweden, 1994. Dissertation No 348, ISBN 91-7871-421-4

[10] Gallant, J. L., Braun, J., Van Essen, D. C.: Selectivity for Polar, Hyperbolic, and Cartesian Gratings in Macaque Visual Cortex. Science, 259:100–103, January 1993.

[11] Knutsson, H., Westin, C-F.: Normalized and Differential Convolution: Methods for Interpolation and Filtering of Incomplete and Uncertain Data. CVPR pages 515–523, New York City, USA, June 1993. IEEE

[12] Bårman, H., Granlund, G. H.: Corner Detection Using Local Symmetry. In Proceedings from SSAB Symposium on Picture Processing, Lund University, Sweden, March 1988. Also in Report LiTH–ISY–I–0935, Computer Vision Laboratory, Linköping University, Sweden, 1988.

[13] Bigün, J.: Local Symmetry Features in Image Processing. PhD thesis, Linköping University, Sweden, 1988. Dissertation No 179, ISBN 91-7870-334-4

[14] Knutsson, H., Hedlund, M., Granlund, G. H.: Apparatus for Determining the Degree of Consistency of a Feature in a Region of an Image that is Divided into Discrete Picture Elements. US-Patent 4.747.152, 1988. (Swedish patent 1986).

[15] Bigün, J.: Optimal Orientation Detection of Circular Symmetry. Report LiTH–ISY–I–0871, Computer Vision Laboratory, Linköping University, Sweden, 1987.

[16] Biederman, I.: Recognition-by-Components: A Theory of Human Image Understanding. Psychological Review, 94(2):115–147, 1987.

[17] Granlund, G. H.: In Search of a General Picture Processing Operator. Computer Graphics and Image Processing, 8(2):155–178, 1978.

[18] Oster, G.: Phosphenes. Scientific American, 222(2):82–87, 1970.

[19] Attneave, F.: Some informational aspects of visual perception. Psychological Review, 61:183–193, 1954