

Line and Edge Detection by Symmetry Filters

Piet W. Verbeek and Lucas J. van Vliet

Pattern Recognition Group Delft, Faculty of Applied Physics, Delft University of Technology,
Lorentzweg 1, 2628 CJ Delft, The Netherlands

Abstract

Sequential (search) methods for edge and line detection and parallel (filter) methods for edge detection are well-known. Filter methods for line detection are generally based on direction-by-direction filtering: each filter is parallel, but only special hardware architectures allow filtering in different directions simultaneously. We present a class of isotropic edge and line detection filters that work in arbitrary dimension: symmetry filters. The principle of the method is correlation of the filter window content with a mapped version. The mapping may be point inversion, blow or rotation, all relative to the window center. Covariance is used as correlation measure. In this paper only point inversion is considered. Moreover, the filter may contain weighting coefficients e.g. to emphasize contributions from the central part. For point inversion, lines give a positive correlation and edges a negative correlation (twofold symmetry and anti-symmetry).

1. Introduction to line detection

Sequential (search) methods for edge and line detection and parallel (filter) methods for edge detection are well-known [1,2]. Filter methods for line detection (one might even include the Hough transform) are generally based on direction-by-direction filtering: each filter is parallel, but only special hardware architectures allow filtering in different directions simultaneously [3,4]. Why can edges be found by a single filter while lines can not? The main difference between an edge and a line lies in the character of their 1D cross sections. The cross section of an edge is a 1D edge in almost all directions. The cross section of a line is a 1D isolated point in almost all directions. There is little difference with the cross section of a 2D isolated point which is a 1D isolated point in all directions. Known isotropic filters do not react to properties in a specific direction.

Therefore, it is hard to discern a line from a 2D isolated point without a set of directional filters. The present work proposes a class of one-step isotropic nonlinear filters based on the concept of local symmetry that can detect any locally symmetric image parts such as lines, edges, strips, squares, circles, chess boards and crosses.

In section 2 we define the characteristic symmetries of a line. In section 3 we derive a 2D point inversion (PI) symmetry filter for line detection and discuss the complexity of point inversion filters. Section 4 shows image processing results, section 5 summarizes the conclusions.

2. Symmetry based line detection

2.1. Matched filters versus symmetry filters

Conventional linear filters for line detection are essentially of the matched filter (MF) type. This can be visualized as

- 1) taking a typical piece of line,
- 2) assume zero noise,
- 3) subtract the mean,
- 4) normalize by the filter size and
- 5) apply it as a convolution filter.

Each grey profile and each direction needs a separate filter. When the grey profile of the line is unknown an idealized shape such as a block or a Gaussian may replace it. When the line direction is unknown it can be assessed by a set of at least three filters (in 2D, Knutsson [4]) equally spaced in direction. Nonlinear filters of the min-max type also need directional information for line detection [5].

In a 2D image that contains several lines of different directions one would like automatic adaptation of the matched filter while passing from one line to another: space variant convolution. It goes too far if at each filter position the piece of image

covered by the filter is taken to be the convolution filter. Each feature will then match itself, which leads to a local signal power estimator instead of a line detector. However, at positions where a line is present it makes little difference if the piece of image is rotated over 180 degrees and then used as convolution filter: matching will still occur. At other positions rotation of the window over 180 degrees will in general spoil the match: the procedure acts as a line detector. Excluding the window center we can make it ignore single points. Actually, we have something less specific than a line detector, we have a local symmetry detector. It will not only find lines but any piece of image that is invariant for rotation over 180 degrees, i.e. twofold symmetric. The line is detected by virtue of its symmetry. Symmetry is invariant under mapping. The mapping here is rotation over 180 degrees. The method can be generalized to other symmetries. We shall call it symmetry filtering. In the next section we shall see that:

- 1) there are other symmetries besides twofold symmetry that are of interest for line detection,
- 2) there are other features than line pieces that can be detected on the basis of their symmetry,
- 3) the principle works in arbitrary dimension.

2.2. Characteristic symmetries of lines and other features

In order to select the best symmetry to exploit in line detection we make an inventory of line symmetries. We define two classes of symmetries. First, symmetries that contain a preferred orientation and therefore can only lead to line detectors for a specific line orientation. When orientation is unknown such detectors still need the direction-by-direction processing of traditional line detectors, which we try to avoid. We are interested in the second class, symmetries that do not contain a preferred orientation, isotropic symmetries.

The orientation specific symmetries of a line are:

- translation symmetry
- reflection symmetry, mirror through the line or perpendicular to the line
- rotation symmetry around axis along the line (in three and more dimensions)
- 180 degrees rotation symmetry around axis perpendicular to the line (in three and more dimensions)

The isotropic symmetries of a line are:

- 180 degrees rotation symmetry, only in 2D
- point inversion symmetry
- blow symmetry

Translation symmetry is traditionally exploited as directional averaging by a 1D uniform filter or in the Hough transform. This shows an alternative to our method of space variant convolution. Our method uses the mapped signal as filter. In directional averaging by a 1D uniform filter the result can also be obtained by averaging a set of mapped versions (translations) of the signal. Noise reduction is proportional to the square root of the number of different versions used. For the symmetries of the second class this number is rather restricted in practical situations: here averaging gives little noise reduction.

Reflection symmetry may select lines (strips) of known orientation but uncertain width. Reflection on a square sampling grid is restricted to sign changes or permutations of coordinates.

Rotation can be over arbitrary angles or over specific angles. Rotation on a square sampling grid is restricted to some discrete rotations ($\pm 90^\circ$ and 180° in 2D, $\pm 90^\circ$, $\pm 120^\circ$ and 180° degrees in 3D), i.e. to permutation and sign change of coordinates.

Our example, 180 degrees rotation symmetry in 2D, is an isotropic symmetry. It is so only in 2D. In three and more dimensions the rotation axis must be specified. Yet, generalization to a more-dimensional isotropic symmetry is still possible. In 2D 180 degrees rotation is identical to point inversion. Point inversion symmetry is isotropic in any dimension.

Blow maps a window to an enlarged copy. In discrete images only integer blow factors can be used. DiZenzo [6] independently devised binary point inversion and blow methods for character recognition. He determines the center of gravity C of a binary object and counts

- The number of object pixel pairs (r_1, r_2) that lie symmetrically around C : $r_1 - C - r_2$
- The number of object pixel pairs (r_1, r_2) collinear with C that have a distance ratio of two: $C - r_1 - r_2$.

In table 1 we show the specificity of symmetry to image features. The many symmetries of a single point are ignored by excluding the window center.

3. Point inversion filtering

3.1. Line detection by point inversion filtering in 2D

As an example of symmetry filtering we consider line detection in 2D by point inversion filtering (PIF).

Table 1. Typical symmetries (+), anti-symmetries (-) and non-symmetries (0) of some local image features.

	point	ball	line	strip	edge	cube	chess	fractal
translation	0	0	+	+	+	0	0	
integer translation	0	0	+	+	+	0	±	
reflection	+	+	+	+	±	+	±	
rotation (2D, ≠90, 180°)	+	+	0	0	0	0	0	
Rotation (3D, ≠90, 180°)	+	+	+	0	+	*	*	
180° rotation (2D)	+	+	+	+	-	+	±	
180° rotation (3D)	+	+	+	+	±	+	±	
90° rotation (2D)	+	+	0	0	0	+	±	
90° rotation (3D)	+	+	+	0	+	+	±	
point inversion	+	+	+	+	-	+	±	
blow	+	0	+	0	+	0	0	+

* special cases such as 120 degrees rotation of cube around diagonal.

Let $s(x_i, y_i)$ be a continuous noise-free 2D grey image that contains a straight symmetric grey line through the origin under an angle ψ with the x_i -axis. It is convenient to use coordinates $a_i = x_i \cos\psi + y_i \sin\psi$ and $b_i = -x_i \sin\psi + y_i \cos\psi$ where the a_i -axis is the line, the b_i -axis is perpendicular to the line (cf. fig. 1). The line has a symmetric profile $s(x_i, y_i) = p(b_i)$, $p(-b_i) = p(b_i)$, $p(0)$ is a maximum.

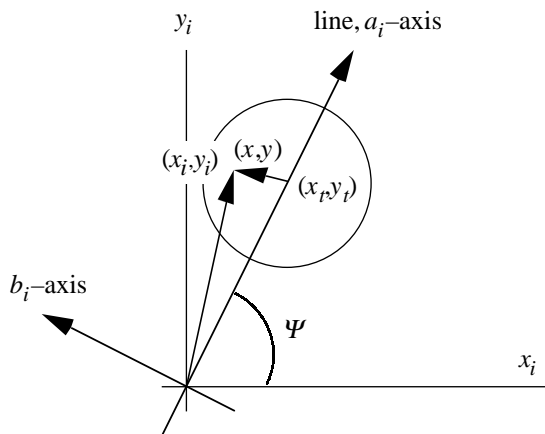


Figure 1: Window over line.

For a circular window W (or any other point inversion symmetric window) of area N centered at (x_i, y_i) the signal covered is $s(x_t+x, y_t+y) = p(b_t+b)$, with $x \equiv x_t-x$, $y \equiv y_t-y$, and $(x, y) \in W$ or $a \equiv a_t-a$ and $b \equiv b_t-b$, and $(a, b) \in W$. As the signal does not depend on a , we divide the window in strips of constant b , with length $a(b)$. Then for functions $f(b)$ that do not depend on a we have

$$\iint_w f(a)dadb = \int f(b)a(b)db$$

In order to construct a matched filter we follow the five points of section 2.1. We take a window around the origin as a typical piece of line. The fixed convolution filter for matched filtering (MF) is

$$\frac{s(x, y) - m(0, 0)}{N} = \frac{p(b) - m(0)}{N}$$

with mean (m)

$$m(x_t, y_t) = \frac{1}{N} \int p(b_t + b)a(b)db \equiv m(b_t)$$

The match filter result is

$$\begin{aligned} MF(x_t, y_t) &= \frac{1}{N} \int (p(b) - m(0))p(b_t + b)a(b)db \\ &= \frac{1}{N} \int (p(b) - m(0))(p(b_t + b) - m(b_t))a(b)db \end{aligned}$$

The space variant convolution filter for matched filtering – an AC power filter (ACF) or local variance filter – is

$$\frac{s(x_t + x, y_t + y) - m(x_t, y_t)}{N} = \frac{p(b_t + b) - m(b_t)}{N}$$

The AC power filter result is

$$\begin{aligned} ACF(x_t, y_t) &= \frac{1}{N} \int (p(b_t + b) - m(b_t))p(b_t + b)a(b)db \\ &= \frac{1}{N} \int (p(b_t + b) - m(b_t))^2 a(b)db \\ &= \frac{1}{N} \int p(b_t + b)^2 a(b)db - m(b_t)^2 \end{aligned}$$

The point inversion changes the space variant convolution filter into

$$\frac{s(x_t - x, y_t - y) - m(x_t, y_t)}{N} = \frac{p(b_t - b) - m(b_t)}{N}$$

The window and the mean do not change under point inversion. The point inversion filter result is

$$PIF(x_t, y_t) = \frac{1}{N} \int (p(b_t - b) - m(b_t))p(b_t + b)a(b)db$$

$$\begin{aligned}
&= \frac{1}{N} \int (p(b_i - b) - m(b_i))(p(b_i + b) - m(b_i))a(b)db \\
&= \frac{1}{N} \int p(b_i - b)p(b_i + b)a(b)db - m(b_i)^2
\end{aligned}$$

In order to see what happens to a window on the line center where we expect a maximum output we consider the window around the origin, where $p(-b) = p(b)$ and

$$\begin{aligned}
MF(0,0) &= ACF(0,0) = PIF(0,0) \\
&= \frac{1}{N} \int p(b)^2 a(b)db - m(0)^2
\end{aligned}$$

At the line center the three filters give the same output. In order to see how sharp the maxima are we compare the second derivatives to b_i , MF'' , ACF'' and PIF'' , at the line center (note that p , m , p'' and m'' are even while p' and m' are odd)

$$\begin{aligned}
MF''(0,0) &= \frac{1}{N} \int p(b)p''(b)a(b)db - m(0)m''(0) \\
ACF''(0,0) &= 2MF''(0,0) + 2GP(0,0) \\
PIF''(0,0) &= 2MF''(0,0) - 2GP(0,0)
\end{aligned}$$

where GP is the gradient power

$$GP(0,0) = \frac{1}{N} \int p'(b_i + b)^2 a(b)db \geq 0$$

A maximum corresponds to a negative second derivative; if MF has a maximum then PIF certainly has (similarly but less interesting: if MF has a minimum, then ACF certainly has). The PIF -peak is more than two times sharper curved than the MF -peak (matched filter) but equally high.

4. Image processing results

In figure 2a a test image is given. It consists of bright fringes of different width, orientation and spacing. Gaussian distributed noise is added of which the amplitude increases linearly from top to bottom in the image (cf. fig 2b). The signal-to-noise ratio decreases from infinity to 0 dB. The result of point inversion filtering with different filter sizes (11, 15, 19 and 23 pixels in diameter) is given in figure 2c-f. We see that all filters improve the SNR with increasing filter size. Positive responses of the PIF are displayed in white, point inversion anti-symmetries are shown in black, while the grey regions indicate no respond at all. The smallest filter has difficulty in finding the widest lines. Here the line profile is wider than the filter size. The largest filter (23) is wider than the dark zones between the fringes in the central top and bottom parts of the image. These dark strips display twofold symmetry and produce positive output.

5. Conclusions

A class of isotropic edge and line detection filters that work in arbitrary dimension has been defined: symmetry filters. The principle of the method is

- mapping of the filter window
- correlation of the filter window content with its mapped version
- using the covariance as a correlation measure

The mapping may be

- point inversion
- blow
- rotation

In this paper only point inversion is considered. For point inversion

- lines give a positive correlation (twofold symmetry)
- edges give a negative correlation (twofold anti-symmetry)

The filter may contain weighting coefficients e.g. to emphasize contributions from the central window part.

A study on symmetry filtering for arbitrary symmetry in arbitrary dimension is under way. The study will include an extensive signal to noise analysis together with an experimental verification.

6. Acknowledgements

This work was partially supported by the Dutch Government as part of the SPIN-FLAIR II program "Delft Intelligent Assembly Cell", the SPIN-3D program for 3D biomedical image analysis and by The Netherlands Foundation for Biomedical Research NWO-MEDIGON, grant 900-538-016.

7. References

- [1] Gerbrands J.J., E. Backer and W.A.G. van der Hoeven, Quantitative Evaluation of Edge Detection by Dynamic Programming, Pattern Recognition in Practice II, E.S. Gelsema and L.N. Kanal, eds., Elsevier (North Holland), 1986, pp. 91-99.
- [2] Verbeek, P.W., H.A. Vrooman and L.J. van Vliet, Low Level Image Processing by Max-Min filters, Signal Processing, Vol.15, no.3, 1988, pp. 249-258.
- [3] Granlund G.H. and J. Arvidsson, The GOP Image Computer, in: Computer Vision, O.D. Faugeras, ed., Cambridge University Press, 1983.
- [4] Knutsson, H., Filtering and Reconstruction in Image Processing, Ph.D.Thesis, Linköping, Sweden, 1982.
- [5] Werman, M. and S. Peleg, Min-Max operators in texture analysis, IEEE PAMI-7, 1985.
- [6] DiZenzo (IBM Rome) private communication, IBM Europe Institute, Garmisch Partenkirchen, 1989.

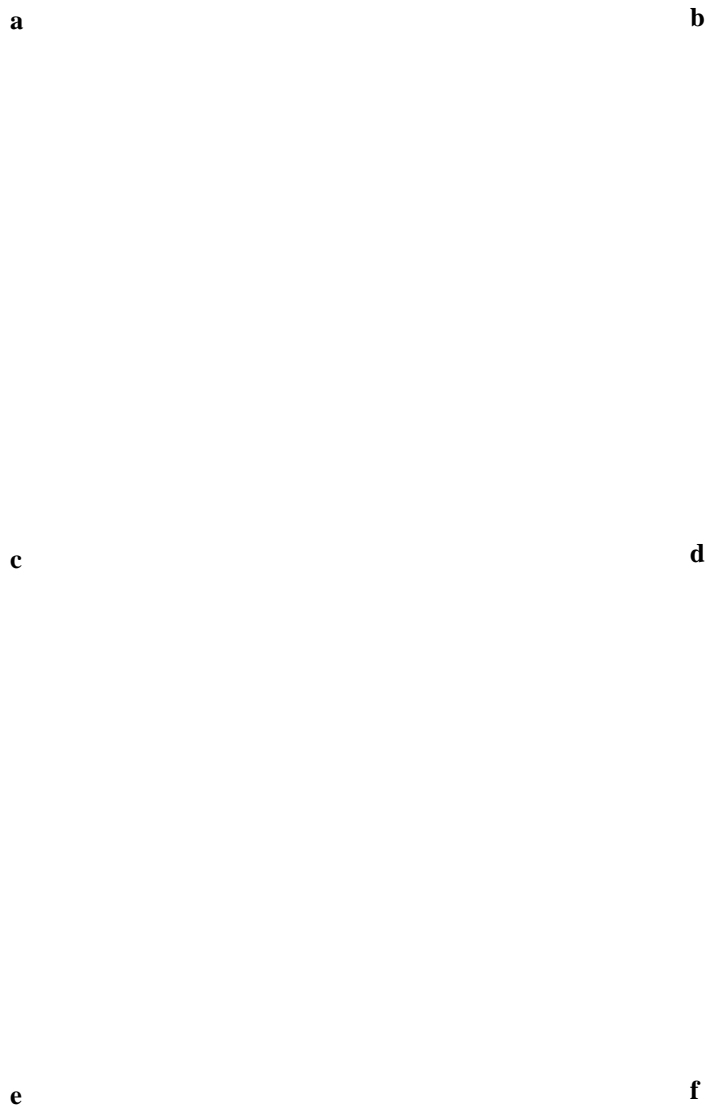


Figure 2: **a)** Image containing bright fringes of different width, orientation and spacing, **b)** Gaussian distributed noise is added with increasing amplitude from top to bottom ($SNR=\infty \rightarrow SNR=0dB$), **c)** result of point inversion symmetry filter (*PIF*) of size 11, **d)** *PIF* of size 15, **e)** *PIF* of size 19, **f)** *PIF* of size 23.