

# Steerable-Scalable Kernels for Edge Detection and Junction Analysis<sup>\*</sup>

Pietro Perona<sup>1,2</sup>

<sup>1</sup> California Institute of Technology 116-81, Pasadena CA 91125, USA

e-mail: perona@verona.caltech.edu

<sup>2</sup> Università di Padova – DEI, via Gradenigo 6A, 35131 Padova, Italy

**Abstract.** Families of kernels that are useful in a variety of early vision algorithms may be obtained by rotating and scaling in a continuum a ‘template’ kernel. These multi-scale multi-orientation family may be approximated by linear interpolation of a discrete finite set of appropriate ‘basis’ kernels. A scheme for generating such a basis together with the appropriate interpolation weights is described. Unlike previous schemes by Perona, and Simoncelli et al. it is guaranteed to generate the most parsimonious one. Additionally, it is shown how to exploit two symmetries in edge-detection kernels for reducing storage and computational costs and generating simultaneously endstop- and junction-tuned filters for free.

## 1 Introduction

Points, lines, edges, textures, motions are present in almost all images of everyday’s world. These elementary visual structures often encode a great proportion of the information contained in the image, moreover they can be characterized using a small set of parameters that are locally defined: position, orientation, characteristic size or scale, phase, curvature, velocity. It is therefore reasonable to start visual computations with measurements of these parameters. The earliest stage of visual processing, common for all the classical early vision modules, could consist of a collection of operators that calculate one or more dominant orientations, curvatures, scales, velocities at each point of the image or, alternatively, assign an ‘energy’, or ‘probability’, value to points of a position-orientation-phase-scale-etc. space. Ridges and local maxima of this energy would mark special interest loci such as edges and junctions. The idea that biological visual systems might analyze images along dimensions such as orientation and scale dates back to work by Hubel and Wiesel [19, 18] in the 1960’s. In the computational vision literature the idea of analyzing images along multiple orientations appears at the beginning of the seventies with the Binford-Horn linefinder [17, 3] and later work by Granlund [14].

A computational framework that may be used to perform this proto-visual analysis is the convolution of the image with kernels of various shapes, orientations, phases, elongation, scale. This approach is attractive because it is simple to describe, implement and analyze. It has been proposed and demonstrated for a variety of early vision tasks [23, 22, 5, 1, 6, 15, 40, 30, 28, 31, 10, 26, 4, 41, 20, 21, 11, 36, 2]. Various ‘general’ computational justifications have been proposed for basing visual processing on the output of a rich set of linear filters: (a) Koenderink has argued that a structure of this type is an adequate substrate for local geometrical computations [24] on the image brightness, (b) Adelson and Bergen [2] have derived it from the ‘first principle’ that the visual system

---

<sup>\*</sup> This work was partially conducted while at MIT-LIDS with the Center for Intelligent Control Systems sponsored by ARO grant DAAL 03-86-K-0171, .

computes derivatives of the image along the dimensions of wavelength, parallax, position, time, (c) a third point of view is the one of ‘matched filtering’: where the kernels are synthesized to match the visual events that one looks for.

The kernels that have been proposed in the computational literature have typically been chosen according to one or more of three classes of criteria: (a) ‘generic optimality’ (e.g. optimal sampling of space-frequency space), (b) ‘task optimality’ (e.g. signal to noise ratio, localization of edges) (c) emulation of biological mechanisms. While there is no general consensus in the literature on precise kernel shapes, there is convergence on kernels roughly shaped like either Gabor functions, or derivatives or differences of either round or elongated Gaussian functions – all these functions have the advantage that they can be specified and computed easily. A good rule of the thumb in the choice of kernels for early vision tasks is that they should have good localization in space and frequency, and should be roughly tuned to the visual events that one wants to analyze.

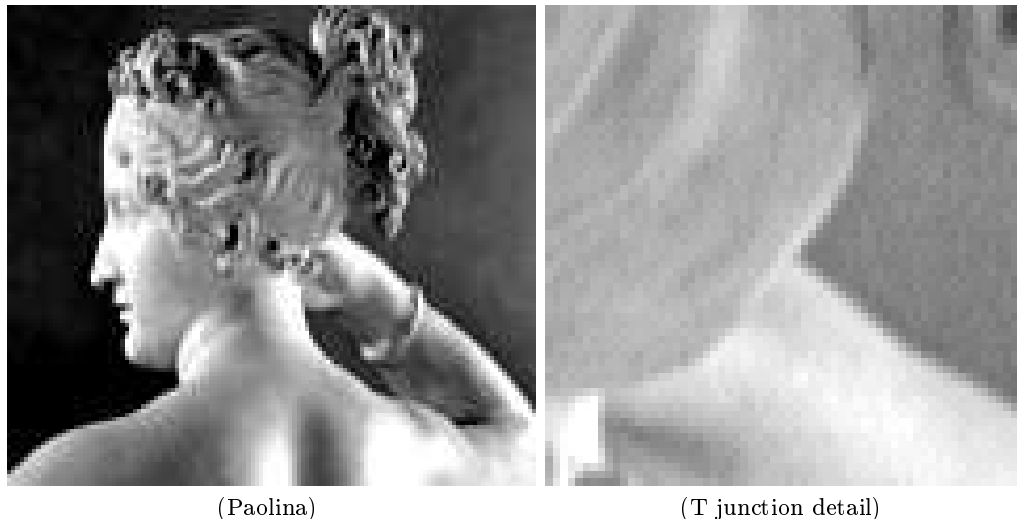
Since points, edges, lines, textures, motions can exist at all possible positions, orientations, scales of resolution, curvatures one would like to be able to use families of filters that are tuned to all orientations, scales and positions. Therefore once a particular convolution kernel has been chosen one would like to convolve the image with deformations (rotations, scalings, stretchings, bendings etc.) of this ‘template’. In reality one can afford only a finite (and small) number of filtering operations, hence the common practice of ‘sampling’ the set of orientations, scales, positions, curvatures, phases<sup>3</sup>. This operation has the strong drawback of introducing anisotropies and algorithmic difficulties in the computational implementations. It would be preferable to keep thinking in terms of a continuum, of angles for example, and be able to localize the orientation of an edge with the maximum accuracy allowed by the filter one has chosen.

This aim may sometimes be achieved by means of interpolation: one convolves the image with a small set of kernels, say at a number of discrete orientations, and obtains the result of the convolution at *any* orientation by taking linear combinations of the results. Since convolution is a linear operation the interpolation problem may be formulated in terms of the kernels (for the sake of simplicity the case of rotations in the plane is discussed here): Given a kernel  $F : \mathbb{R}^2 \rightarrow \mathbb{C}^1$ , define the family of ‘rotated’ copies of  $F$  as:  $F_\theta = F \circ R_\theta$ ,  $\theta \in \mathbb{S}^1$ , where  $\mathbb{S}^1$  is the circle and  $R_\theta$  is a rotation. Sometimes it is possible to express  $F_\theta$  as

$$F_\theta(\mathbf{x}) = \sum_{i=1}^n \alpha(\theta)_i G_i(\mathbf{x}) \quad \forall \theta \in \mathbb{S}^1, \forall \mathbf{x} \in \mathbb{R}^2 \quad (1)$$

---

<sup>3</sup> Motion flow computation using spatiotemporal filters has been proposed by Adelson and Bergen [1] as a model of human vision and has been demonstrated by Heeger [15] (his implementation had 12 discrete spatio-temporal orientations and 3 scales of resolution). Work on texture with multiple-resolution multiple-orientation kernels is due to Knutsson and Granlund [23] (4 scales, 4 orientations, 2 phases), Turner [40] (4 scales, 4 orientations, 2 phases), Fogel and Sagi [10] (4 scales, 4 orientations, 2 phases), Malik and Perona [26] (11 scales, 6 orientations, 1 phase) and Bovik et al. [4] (n scales, m orientations, l phases). Work on stereo by Kass [22] (12 filters, scales, orientations and phases unspecified) and Jones and Malik [20, 21] (see also the two articles in this book) (6 scales, 2-6 orientations, 2 phases). Work on curved line grouping by Parent and Zucker [31] (1 scale, 8 orientations, 1 phase) and Malik and Giguere [25] (9 curvatures, 1 scale, 18 orientations, 2 phases). Work on brightness edge detection by Binford and Horn [17, 3] (24 orientations), Canny [6] (1-2 scales,  $\infty$ -6 orientations, 1 phase), Morrone, Owens and Burr [30, 28] (1-3 scales, 2-4 orientations,  $\infty$  phases), unpublished work on edge and illusory contour detection by Heitger, Rosenthaler, Kübler and von der Heydt (6 orientations, 1 scale, 2 phases). Image compression by Zhong and Mallat [41] (4 scales, 2 orientations, 1 phase).

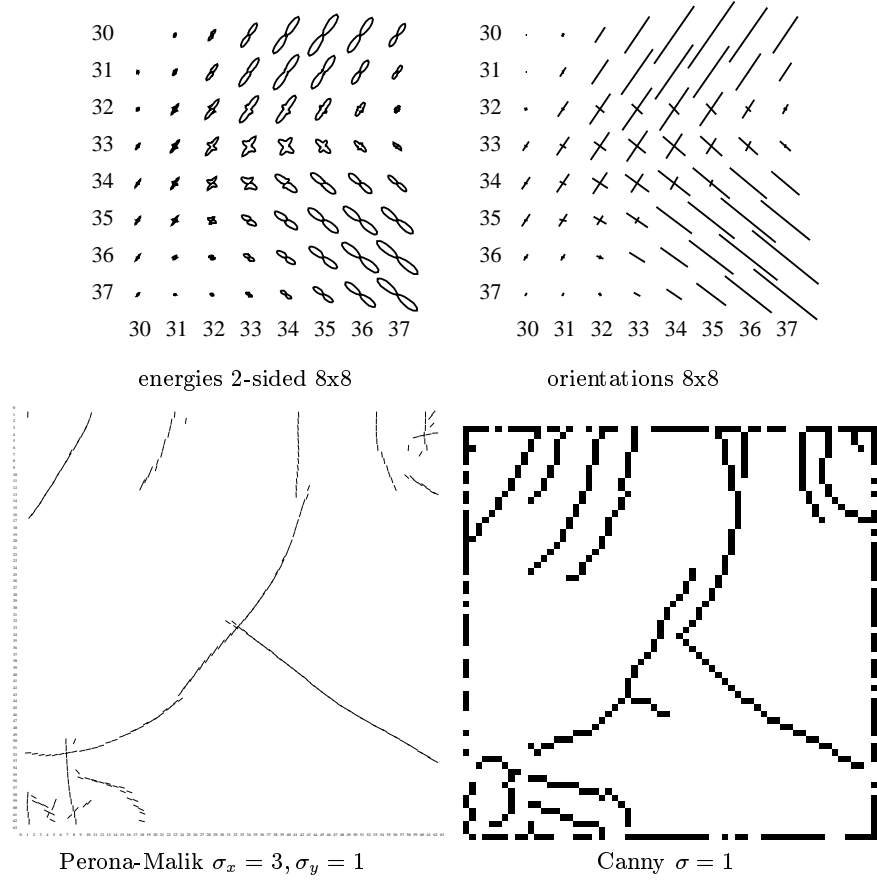


**Fig. 1.**

a *finite* linear combination of functions  $G_i : \mathbb{R}^2 \rightarrow \mathbb{C}^1$ . It must be noted that, at least for positions and phases, the mechanism for realizing this in a systematic way is well understood: in the case of positions the sampling theorem gives conditions and an interpolation technique for calculating the value of the filtered image at any point in a continuum; in the case of phases a pair of filters in quadrature can be used for calculating the response at any phase [1, 29]. Rotation, scalings and other deformations are less well understood.

An example of ‘rotating’ families of kernels that have a finite representation is well known: the first derivative along an arbitrary direction of a round ( $\sigma_x = \sigma_y$ ) Gaussian may be obtained by linear combination of the X- and Y-derivatives of the same. The common implementations of the Canny edge detector [6] are based on this principle. Unfortunately the kernel obtained this way has poor orientation selectivity and therefore it is unsuited for edge detection if one wants to recover edge-junctions (see in Fig. 2 the comparison with a detector that uses narrow orientation-selective filters). Freeman and Adelson have recently proposed [11, 12] to construct orientation-selective kernels that can be exactly rotated by interpolation (they call this property “steerability”) and have shown that higher order derivatives of round Gaussians, indeed all polynomials multiplied by a radially symmetric function are steerable. They have also shown that functions that may be written as finite sums of polar-separable kernels with sinusoidal  $\theta$  component are also steerable. These functions may be designed to have higher orientation selectivity and can be used for contour detection and signal processing [11]. However, one must be aware of the fact that for most kernels  $F$  of interest a finite decomposition of  $F_\theta$  as in Eq. (1) cannot be found. For example the elongated kernels used in edge detection by [35, 36] (see Fig. 2 top right) do not have a finite decomposition as in Eq. (1).

Perona [32, 33] has proposed an approximation technique that, given an  $F_\theta$ , allows one to generate a function  $G_\theta^{[n]}$  which is sufficiently similar to  $F_\theta$  and that is steerable, i.e. can be expressed as a finite sum of  $n$  terms as in (1). This technique is guaranteed to find the most parsimonious steerable approximation to a given kernel  $F_\theta$ , i.e. given a tolerable amount  $\delta$  of error it computes an approximating  $G_\theta^{[n]}$  that has minimum number  $n$  of components and is within a distance  $\delta$  from  $F_\theta$ . Perona [32, 33] and Simoncelli et al. [9]



**Fig. 2.** Example of the use of orientation-selective filtering on a continuum of orientations (see Perona and Malik [35, 36]). Fig. 1 (Left) Original image. Fig. 1 (Right) A T-junction (64x64 pixel detail from a region roughly at the centre of the original image). The kernel of the filter for the edge-detector is elongated to have high orientation selectivity; it is depicted in Fig. 3. (Top-left) Modulus  $R(x, y, \theta)$  of the output of the complex-valued filter (polar plot shown for 8x8 pixels in the region of the T-junction). (Top-right) The local maxima of  $|R(x, y, \theta)|$  with respect to  $\theta$ . Notice that in the region of the junction one finds two local maxima in  $\theta$  corresponding to the orientation of the edges. Searching for local maxima in  $(x, y)$  in a direction orthogonal to the maximizing  $\theta$ 's one can find the edges (Bottom left) with high accuracy (error around 1 degree in orientation and 0.1 pixels in position). (Bottom right) Comparison with the output of a Canny detector using the same kernel width ( $\sigma$  in pixel units).

have proposed non-optimal extensions to the case of joint rotation and scaling.

In this paper the general case of compact deformations is reviewed in section 2. Some results of functional analysis are recalled to formulate the decomposition technique in all generality. The case of rotations is briefly recalled in section 3 to introduce some notation which is used later in the paper. In section 4 it is shown how to generate a steerable and scalable family. Experimental results and implementation issues are presented and discussed. Finally, in section 5 some basic symmetries of edge-detection kernels are studied and their use described in (a) reducing calculations and storage, and (b) implementing filters useful for junction analysis at no extra cost.

## 2 Deformable functions

In order to solve the approximation problem one needs of course to define the ‘quality’ of the approximation  $G_\theta^{[n]} \approx F_\theta$ . There are two reasonable choices: (a) a distance  $D(F_\theta, G_\theta^{[n]})$  in the space  $\mathbb{R}^2 \times \mathbb{S}^1$  where  $F_\theta$  is defined; (b) if  $F_\theta$  is the kernel of some filter one is interested in the worst-case error in the ‘output’ space: the maximum distance  $d(\langle F_\theta, f \rangle, \langle G_\theta^{[n]}, f \rangle)$  over all unit-norm  $f$  defined on  $\mathbb{R}^2$ . The symbols  $\Delta_n$  and  $\delta_n$  will indicate the ‘optimal’ distances, i.e. the minimum possible approximation errors using  $n$  components. These quantities may be defined using the distances induced by the  $L^2$ -norm:

**Definition.**

$$\Delta_n(F_\theta) = \inf_{G_\theta^{[n]}} \|F_\theta - G_\theta^{[n]}\|_{\mathbb{R}^2 \times \mathbb{S}^1}$$

$$\delta_n(F_\theta) = \inf_{G_\theta^{[n]}} \sup_{\|f\|=1} \|\langle F_\theta - G_\theta^{[n]}, f \rangle\|_{\mathbb{R}^2 \times \mathbb{S}^1}$$

The existence of the optimal finite-sum approximation of the kernel  $F_\theta(\mathbf{x})$  as described in the introduction is not peculiar to the case of rotations. This is true in more general circumstances: this section collects a few facts of functional analysis that show that one can compute finite optimal approximations to continuous families of kernels whenever certain ‘compactness’ conditions are met.

Consider a parametrized family of kernels  $F(\mathbf{x}; \theta)$  where  $\mathbf{x} \in X$  now indicates a generic vector of variables in a set  $X$  and  $\theta \in T$  a vector of parameters in a set  $T$ . (The notation is changed slightly from the previous section.) Consider the sets  $A$  and  $B$  of continuous functions from  $X$  and  $T$  to the complex numbers, call  $a(\mathbf{x})$  and  $b(\theta)$  the generic elements of these two sets. Consider the operator  $L : A \rightarrow B$  defined by  $F$  as:

$$(La(\cdot))(\theta) = \langle F(\cdot; \theta), a(\cdot) \rangle_A \quad (2)$$

A first theorem says that if the kernel  $F$  has bounded norm then the associated operator  $L$  is compact (see [7] pag. 316):

**Theorem 1.** *Let  $X$  and  $T$  be locally compact Hausdorff spaces and  $F \in L_2(X \times T)$ . Then  $L$  is well defined and is a compact operator.*

Such a kernel is commonly called a *Hilbert-Schmidt kernel*.

A second result tells us that if a linear operator is compact, then it has a discrete spectrum (see [8] pag. 323):

**Theorem 2.** *Let  $L$  be a compact operator on (complex) normed spaces, then the spectrum  $S$  of  $L$  is at most denumerable.*

A third result says that if  $L$  is continuous and operates on Hilbert spaces then the compactness property transfers to the adjoint of  $L$  (see [8] pag. 329):

**Theorem 3.** *Let  $L$  be a compact operator on Hilbert spaces, then the adjoint  $L^*$  is compact.*

Trivially, the composition of two compact operators is compact, so the operators  $LL^*$  and  $L^*L$  are compact and have a discrete spectrum as guaranteed by theorem 2. The singular value decomposition (SVD) of the operator  $L$  can therefore be computed as the collection of triples  $(\sigma_i, a_i, b_i), i = 0, \dots$  where the  $\sigma_i$  constitute the spectra of both  $LL^*$  and  $L^*L$  and the  $a_i$  and  $b_i$  are the corresponding eigenvectors.

The last result can now be enunciated (see [37] Chap.IV, Theorem 2.2):

**Theorem 4.** *Let  $L : A \rightarrow B$  be a linear compact operator between two Hilbert spaces. Let  $a_i, b_i, \sigma_i$  be the singular value decomposition of  $L$ , where the  $\sigma_i$  are in decreasing order of magnitude. Then*

1. *An optimal  $n$ -dimensional approximation to  $L$  is  $L_n = \sum_{i=1}^n \sigma_i a_i b_i$*
2. *The approximation errors are  $d_n(L) = \sigma_{n+1}$ , and  $\Delta_n^2(L) = \sum_{i=n+1}^N \sigma_i^2$*

As a result we know that when our original template kernel  $F(\mathbf{x})$  and the chosen family of deformations  $R(\theta)$  define a Hilbert-Schmidt kernel  $F(\mathbf{x}; \theta) = (F \circ R(\theta))(\mathbf{x})$  then it is possible to compute a finite discrete approximation as for the case of 2D rotations.

Are the families of kernels  $F(\mathbf{x}; \theta)$  of interest in vision Hilbert-Schmidt kernels? In the cases of interest for vision applications the ‘template’ kernel  $F(\mathbf{x})$  typically has a finite norm, i.e. it belongs to  $L_2(X)$  (all kernels used in vision are bounded compact-support kernels such as Gaussian derivatives, Gabors etc.). However, this is not a sufficient condition for the family  $F(\mathbf{x}; \theta) = F \circ R(\theta)(\mathbf{x})$  obtained composing  $F(\mathbf{x})$  with deformations  $R(\theta)$  (rotations, scalings) to be a Hilbert-Schmidt kernel: the norm of  $F(\mathbf{x}; \theta)$  could be unbounded (e.g. if the deformation is a scaling in the unbounded interval  $(0, \infty)$ ). A sufficient condition for the associated family  $F(\mathbf{x}; \theta)$  to be a Hilbert-Schmidt kernel is that the inverse of the Jacobian of the transformation  $R$ ,  $|JR|^{-1}$  belongs to  $L_2(T)$  (see [34]).

A typical condition in which this arises is when the transformation  $R$  is unitary, e.g. a rotation, translation, or an appropriately normalized scaling, and the set  $T$  is bounded. In that case the norm of  $\|JR\|^{-1}$  is equal to the measure of  $T$ . The following sections in this paper will illustrate the power of these results by applying them to the decomposition of rotating rotating and scaled kernels.

A useful subclass of kernels  $F$  for which the finite orthonormal approximation can be in part explicitly computed is obtained by composing a template function with transformations  $T_\theta$  belonging to a compact group. This situation arises in the case of  $n$ -dimensional rotations and is useful for edge detection in tomographic data and spatiotemporal filtering. It is discussed in [32, 33, 34].

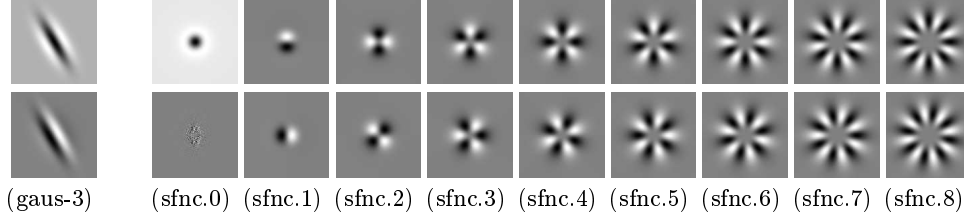
### 3 Rotation

To make the paper self-contained the formula for generating a steerable approximation is recalled here. The  $F_\theta^{[n]}$  which is the best  $n$ -dimensional approximation of  $F_\theta$  is defined as follows:

**Definition.** Call  $F_\theta^{[n]}$  the  $n$ -terms sum:

$$F_\theta^{[n]} = \sum_{i=1}^n \sigma_i a_i(\mathbf{x}) b_i(\theta) \quad (3)$$

with  $\sigma_i, a_i$  and  $b_i$  defined in the following way: let  $\hat{h}(\nu)$  be the (discrete) Fourier transform of the function  $h(\theta)$  defined by:



**Fig. 3.** The decomposition  $(a_i, b_i, \sigma_i)$  of a complex kernel used for brightness-edge detection [36]. (Left) The template function (gaus-3) is shown rotated counterclockwise by  $120^\circ$ . Its real part (above) is the second derivative along the vertical (Y) axis of a Gaussian with  $\sigma_x : \sigma_y$  ratio of 1:3. The imaginary part (below) is the Hilbert transform of the real part along the Y axis. The singular values  $\sigma_i$  (not shown here – see [34]) decay exponentially:  $\sigma_{i+1} \approx 0.75\sigma_i$ . (Right) The functions  $a_i$  (sfnc.i) are shown for  $i = 0 \dots 8$ . The real part is above; the imaginary part below. The functions  $b_i(\theta)$  are complex exponentials (see text) with associated frequencies  $\nu_i = i$ .

$$h(\theta) = \int_{\mathbb{R}^2} F_\theta(\mathbf{x}) \overline{F_{\theta'=0}(\mathbf{x})} d\mathbf{x} \quad (4)$$

and let  $\nu_i$  be the frequencies on which  $\hat{h}(\nu)$  is defined, ordered in such a way that  $\hat{h}(\nu_i) \geq \hat{h}(\nu_j)$  if  $i \leq j$ . Call  $N \leq \infty$  the number of nonzero terms  $\hat{h}(\nu_i)$ . Finally, define the quantities:

$$\sigma_i = \hat{h}(\nu_i)^{1/2} \quad (5)$$

$$b_i(\theta) = e^{j2\pi\nu_i\theta} \quad (6)$$

$$a_i(\mathbf{x}) = \sigma_i^{-1} \int_{\mathbb{S}^1} \overline{F_\theta(\mathbf{x})} e^{j2\pi\nu_i\theta} d\theta \quad (7)$$

See Fig. 3 and [32, 33, 34] for details and a derivation of these formulae.

## 4 Rotation and scale

A number of filter-based early vision and signal processing algorithms analyze the image at multiple scales of resolution. Although most of the algorithms are defined on, and would take advantage of, the availability of a continuum of scales only a discrete and small set of scales is usually employed due to the computational costs involved with filtering and storing images. The problem of multi-scale filtering is somewhat analogue to the multi-orientation filtering problem: given a template function  $F(\mathbf{x})$  and defined  $F_\sigma(\mathbf{x})$  as  $F_\sigma(\mathbf{x}) = \sigma^{1/2} F(\sigma\mathbf{x})$ ,  $\sigma \in (0, \infty)$  one would like to be able to write  $F_\sigma$  as a (small) linear combination:

$$F_\sigma(\mathbf{x}) = \sum_i s_i(\sigma) d_i(\mathbf{x}) \quad \sigma \in (0, \infty) \quad (8)$$

Unfortunately the domain of definition of  $s$  is not bounded (it is the real line) and therefore the kernel  $F_\sigma(\mathbf{x})$  is not Hilbert-Schmidt (it has infinite norm). As a consequence the spectrum of the  $LL^*$  and  $L^*L$  operators is continuous and no discrete approximation may be computed.

One has therefore to renounce to the idea of generating a continuum of scales spanning the whole positive line. This is not a great loss: the range of scales of interest is never the entire real line. An interval of scales  $(\sigma_1, \sigma_2)$ , with  $0 < \sigma_1 \leq \sigma_2 < \infty$  is a very realistic scenario; if one takes the human visual system as an example, the range of frequencies to which it is most sensitive goes from approximately 2 to 16 cycles per degree of visual angle i.e. a range of 3 octaves. In this case the interval of scales is compact and one can apply the results of section 2 and calculate the SVD and therefore an  $L_2$ -optimal finite approximation.

In this section the optimal scheme for doing so is proposed. The problem of simultaneously steering and scaling a given kernel  $F(\mathbf{x})$  generating a family  $F_{(\sigma, \theta)}(\mathbf{x})$  which has a finite approximation will be tackled. Previous non-optimal schemes are due to Perona [32, 33] and Simoncelli et al. [9, 12].

#### 4.1 Polar-separable decomposition

Observe first that the functions  $a_i$  defined in eq.(7) are polar-separable. In fact  $\mathbf{x}$  may be written in polar coordinates as  $\mathbf{x} = \|\mathbf{x}\| R_{\phi(\mathbf{x})} \mathbf{u}$  where  $\mathbf{u}$  is some fixed unit vector (e.g. the 1st coordinate axis versor) and  $\phi(\mathbf{x})$  is the angle between  $\mathbf{x}$  and  $\mathbf{u}$  and  $R_{\phi(\mathbf{x})}$  is a rotation by  $\phi$ . Substituting the definition of  $F_{\theta}$  in (7) we get:

$$\begin{aligned} a_i(\mathbf{x}) &= \sigma_i^{-1} \int_{S^1} \overline{F(\|\mathbf{x}\| R_{\theta+\phi(\mathbf{x})}(\mathbf{u}))} e^{j2\pi\nu_i\theta} d\theta = \\ &= \sigma_i^{-1} e^{-j2\pi\nu_i\phi(\mathbf{x})} \int_{S^1} \overline{F(\|\mathbf{x}\| R_{\psi}(\mathbf{u}))} e^{j2\pi\nu_i\psi} d\psi \end{aligned}$$

so that (3) may be also written as :

$$F_{\theta}(\mathbf{x}) = \sum_{i=1}^N \sigma_i c_i(\|\mathbf{x}\|) e^{j2\pi\nu_i(\theta-\phi(\mathbf{x}))} \quad (9)$$

$$c_i(\|\mathbf{x}\|) = \sigma_i \int_{S^1} \overline{F(\|\mathbf{x}\| R_{\psi}(\mathbf{u}))} e^{j2\pi\nu_i\psi} d\psi \quad (10)$$

#### 4.2 Scaling is a 1D problem

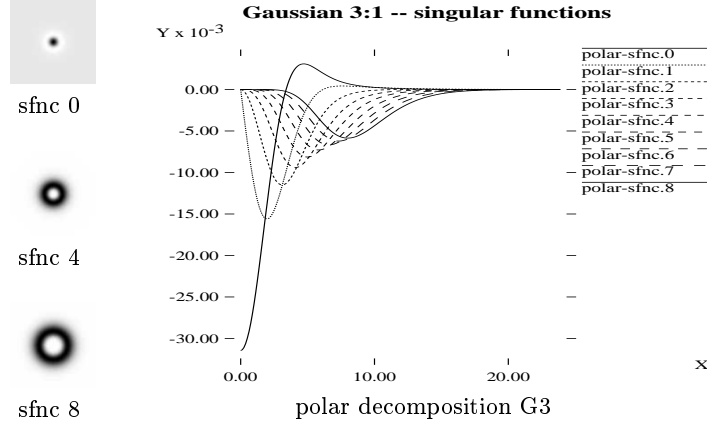
The scaling operation only affects the radial components  $c_i$  and does not affect the angular components. The problem of scaling the kernels  $a_i$ , and therefore  $F_{\theta}$  through its decomposition, is then the problem of finding a finite (approximate) decomposition of continuously scaled versions of functions  $c(\rho)$ :

$$c_{\sigma}(\rho) = \sum_k s_k(\sigma) r_k(\rho) \quad \sigma \in (\sigma_1, \sigma_2) \quad (11)$$

If the scale interval  $(\sigma_1, \sigma_2)$  and the function  $c$  are such that the operator  $L$  associated to  $F$  is compact then we can obtain the optimal finite decomposition via the singular value decomposition. The conditions for compactness of  $L$  are easily met in the cases of practical importance: it is sufficient that the interval  $(\sigma_1, \sigma_2)$  is bounded and that the norm of  $c(\rho)$  is bounded ( $\rho \in \mathbb{R}^+$ ).

Even if these conditions are met, the calculations usually cannot be performed analytically. One can employ a numerical routine (see e.g. [38]) and for each  $c_i$  (below indicated as  $c^i$ ) obtain an SVD expansion of the form:





**Fig. 4.** (Right) The plots of  $c_i(\rho)$ , the radial part of the singular functions  $a_i$  (cfr. eq. 9). The  $\theta$  part is always a complex exponential. The original kernel is the same as in fig. 3. (Left) The 0th, 4th and 8th components  $c_0$ ,  $c_4$  and  $c_8$  represented in two dimensions.

$$c_\sigma^i(\rho) = \sum_k \gamma_k^i s_k^i(\sigma) r_k^i(\rho) \quad (12)$$

As discussed before (Theorem 4) one can calculate the approximation error from the sequence of the singular values  $\gamma_k^i$ . Finally, substituting (12) into (10) the scale-orientation expansion takes the form (see Fig. 6):

$$F_{\theta, \sigma}(\mathbf{x}) = \sum_{i=1}^N \sigma_i e^{j2\pi\nu_i(\theta - \phi(\mathbf{x}))} \sum_{k=1}^{n_i} \gamma_k^i s_k^i(\sigma) r_k^i(\|\mathbf{x}\|) \quad (13)$$

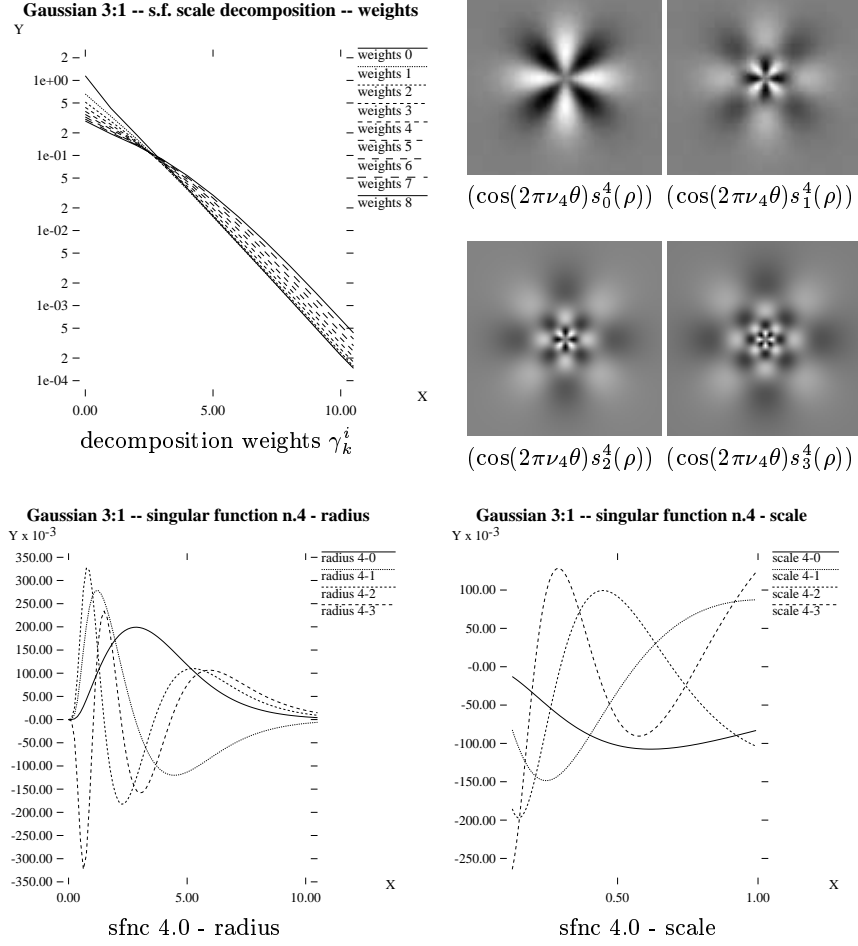
Filtering an image  $I$  with a deformable kernel built this way proceeds as follows: first the image is filtered with kernels  $a_k^i(\mathbf{x}) = \exp(-j2\pi\nu_i\phi(\mathbf{x}))r_k^i(\|\mathbf{x}\|)$ ,  $i = 0, \dots, N$ ,  $k = 0, \dots, n_i$ , the outputs  $I_k^i$  of this operation can be combined as  $I_{\theta, \sigma}(\mathbf{x}) = \sum_{i=1}^N \sigma_i b_i(\theta) \sum_{k=1}^{n_i} \gamma_k^i s_k^i(\sigma) I_k^i(\mathbf{x})$  to yield the result.

### 4.3 Polar-separable decomposition, experimental results

An orientation-scale decomposition was performed on the kernel of Fig. 3 (second derivative of a Gaussian and its Hilbert transform,  $\sigma_x : \sigma_y = 3 : 1$ ). The decomposition recalled in sec. 3 and shown in Fig. 3 was taken as a starting point. The corresponding functions  $c_i(\rho)$  of eq. (9) are shown in Fig. 4.

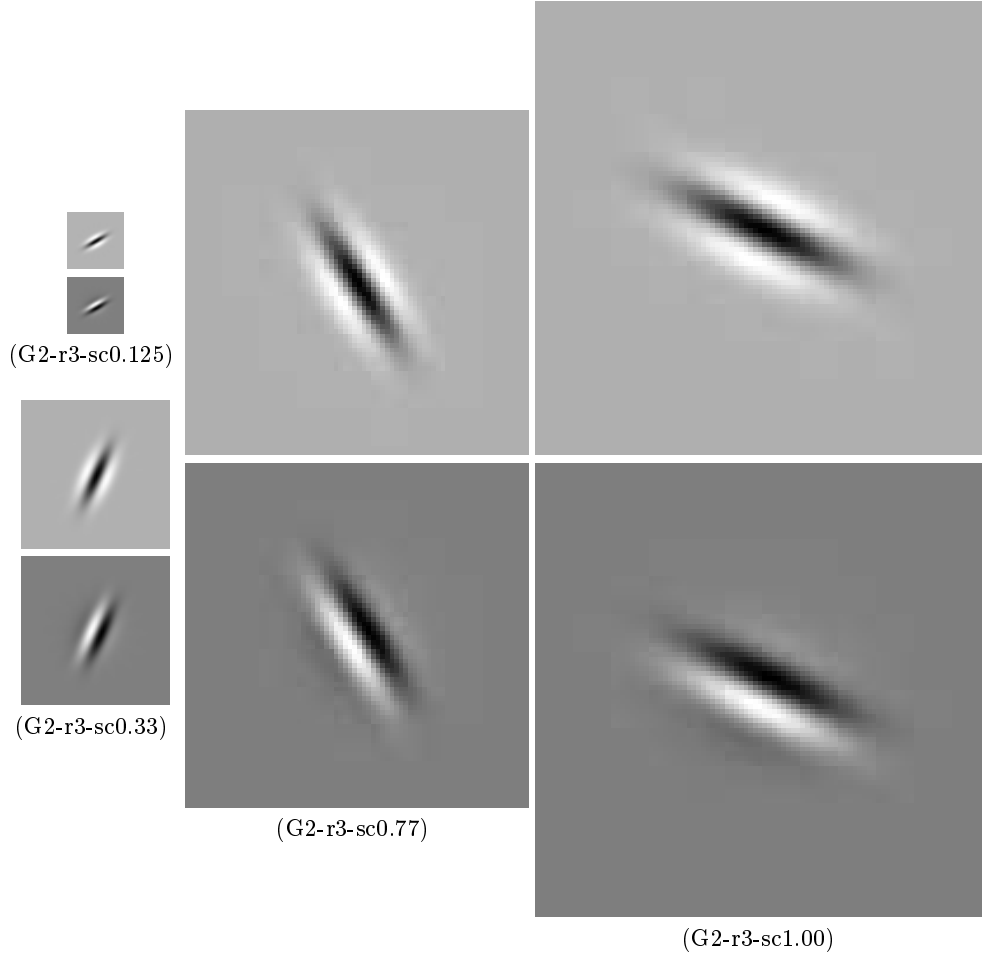
The interval of scales chosen was  $(\sigma_1, \sigma_2)$  s.t.  $\sigma_1 : \sigma_2 = 1 : 8$ , an interval which is ample enough for a wide range of visual tasks.

The range of scales was discretized in 128 samples for computing numerically the singular value decomposition  $(\gamma_k^i, s_k^i, r_k^i)$  of  $c_\sigma^i(\rho)$ . The computed weights  $\gamma_k^i$  are plotted on a logarithmic scale in Fig. 5 (Top). The 'X' axis corresponds to the  $k$  index, each curve is indexed by  $i$ ,  $i = 0, \dots, 8$ . One can see that for all the  $c_i$  the error decreases exponentially at approximately the same rate. The components  $r_k^i(\rho)$  and  $s_k^i(\sigma)$ ,  $i = 4$ ,  $k = 0, \dots, 3$  are shown in the two plots at the bottom of Fig. 5.



**Fig. 5.** Scale-decomposition of the radial component of the functions  $a_i$ . The interval of scales  $\sigma$  is  $\sigma \in (0.125, 1.00)$ . See also Fig. 6. (Top-left) The weights  $\gamma_k^i$  of each polar functions' decomposition ( $i = 0, \dots, 8$ ,  $k$  along the x axis). The decay of the weights is exponential in  $k$ ; 5 to 8 components are needed to achieve 1% error (e.g 5 for the 0th, 7 for the 4th and 8 for the 8th shown in fig 4). (Bottom) The first four radial (left) and scale (right) components of the 5th singular function:  $r_k^4(\rho)$  and  $s_k^4(\sigma)$ ,  $k = 0, \dots, 3$  (see Eq. (12)). (Top-right) The real parts of the first four scale-components of the 5th singular function  $a_5$ :  $\cos(2\pi\nu_4\theta)s_k^4(\rho)$  with  $k = 0, \dots, 3$  (see Eq. (13)).

In figure Fig. 6 reconstructions of the kernel based on a 1% error decomposition are shown for various scales and angles. A maximum of 1% error was imposed on the original steerable decomposition, and again on the scale decomposition of each single  $a_i$ . The measured error was 2.5% independently from angle and scale. The total number of filters required to implement a 3-octave 1% (nominal, 2.5% real) approximation error of the 3:1 Gaussian pair is 16 (rotation) times 8 (scale) = 128. If 10% approximation error is allowed the number of filters decreases by approximately a factor 4 to 32.



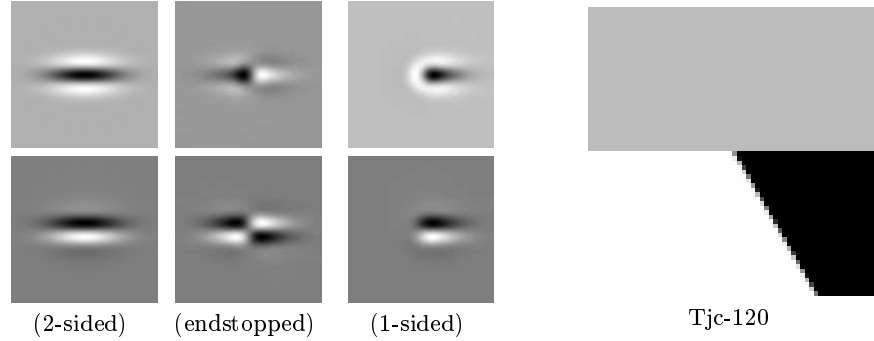
**Fig. 6.** The kernel at different scales and orientations: the scales are (left to right) 0.125, 0.33, 0.77, 1.00. The orientations are (left to right)  $30^\circ$ ,  $66^\circ$ ,  $122^\circ$ ,  $155^\circ$ . The kernels shown here were obtained from the scale-angle decomposition shown in the previous figures.

## 5 Kernel symmetries and junction analysis

The Hilbert-pair kernels used by [27, 11, 36] for edge detection have a number of interesting symmetries that may be exploited to reduce the computational and storage costs by a factor of two. Moreover, these symmetries may be used to reconstruct the response of two associated kernels, endstopped and one-sided, that are useful for the analysis of edge junctions. The kernels of figure 7, are used here as specific examples.

An illustration of the use of these kernels for the analysis of edges and junctions is proposed in Fig. 8 where response maxima w.r. to orientation  $\theta$  as in Fig. 2 are shown for a different image, a synthetic T-junction (Fig. 7, right). The kernels employed for this demonstration have shape as in Fig. 7 and are derived from an elongated Gaussian function of variances  $\sigma_y = 1.2$  pixels and  $\sigma_x : \sigma_y = 3 : 1$ .

From equation (10) one can see that the coefficients  $c_i(\rho)$  (where  $\rho = \|\mathbf{x}\|$ ) are, for each value of  $\rho$ , the Fourier coefficients of  $F_\theta(\mathbf{x})$  along a circular path of radius  $\rho$  and



**Fig. 7.** Three complex-valued kernels used in edge and junction analysis (the real parts are shown above and imaginary parts below). The first one (2-sided) is ‘tuned’ to edges that are combinations of steps and lines (see [36]) – it is the same as in Fig. 3 top left, shown at an orientation of  $0^\circ$ ; the second kernel one (endstopped) is tuned to edge endings and ‘crisscross’ junctions [13, 16, 39]: it is equivalent to a 1st derivative of the 2-sided kernel along its axis direction; the third one (1-sided) may be used to analyze arbitrary junctions. All three kernels may be obtained at any orientation by combining suitably the ‘basis’ kernels  $a_i$  shown in Fig. 3.

center in the origin. The circular path begins and ends at the positive side of the X axis. Consider now such a path for the 2-sided kernel of Fig. 7: observe that for every  $\rho$  we have at least two symmetries.

For the real part:

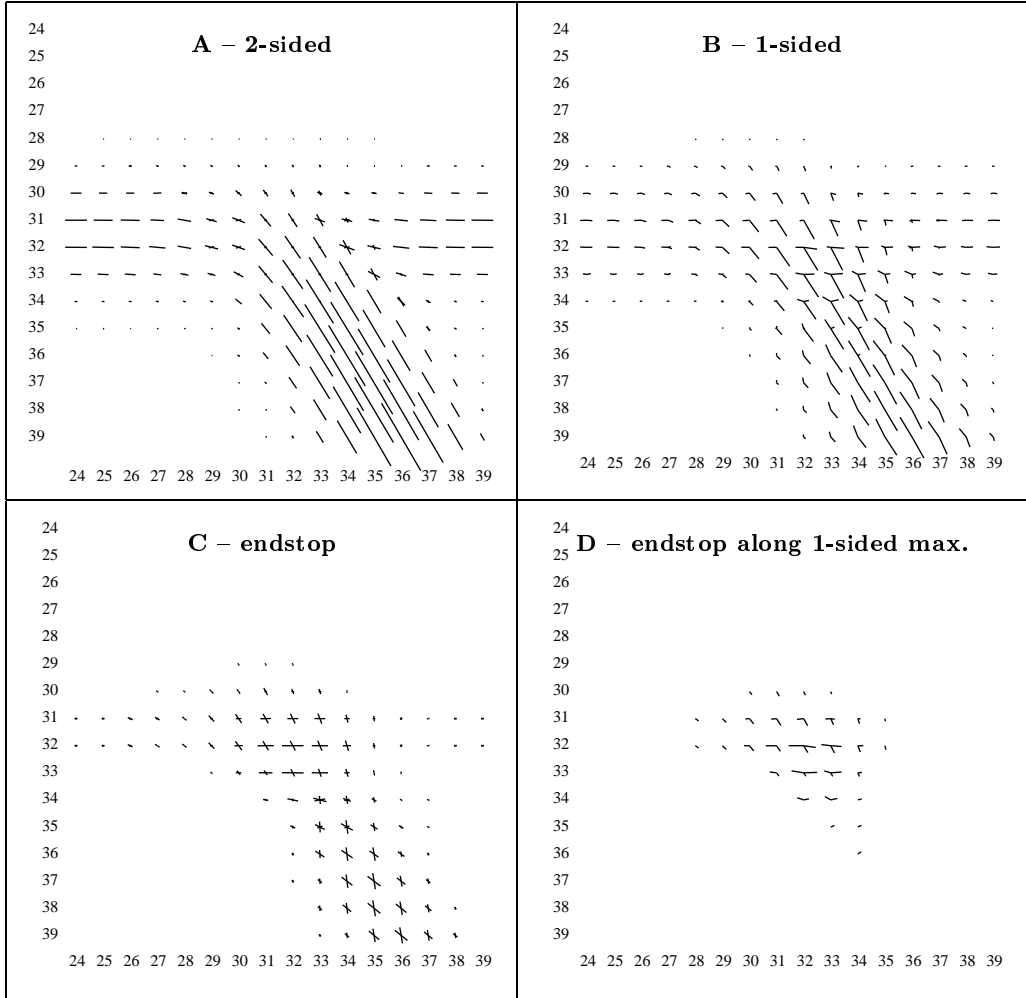
- (E) the function is even-symmetric,
- ( $\Pi^+$ ) a translation of the function by  $\pi$  returns the same function (i.e. it is  $\pi$ -periodic).

For the imaginary part:

- (O) the function is odd-symmetric,
- ( $\Pi^-$ ) a translation of the function by  $\pi$  returns the function multiplied by  $-1$ .

These symmetries imply corresponding properties in the discrete Fourier transform (DFT) of the functions: symmetry (E) implies a DFT with zero coefficients for the sinusoidal components; symmetry (O) a DFT with zero cosinusoidal components; symmetry ( $\Pi^+$ ) implies that the odd-frequency components are zero; symmetry ( $\Pi^-$ ) that the even-frequency components are zero.

As a consequence, the DFT of the real part of the 2-sided kernel is only made up of even-frequency cosine components, while the imaginary part is only made up of odd-frequency sinus components. If the complex-exponential, rather than the sinus-cosine, notation is used, as in eq. (7) and (10), this implies that the odd-frequency coefficients only depend on the imaginary part of the kernel, while the even-frequency components depend on the real part. The negative-frequency components  $\sigma_i a_i$  are equal to the positive-frequency components for even frequencies and to the opposite for odd-frequencies. The negative-frequency components therefore do not need to be computed and stored thus saving a factor 2 of storage and computations. Equation (3) may therefore be re-written as follows (for convenience of notation suppose that the number of components  $n$  is odd:  $n=2b+1$ , and that the  $n$  frequencies  $\nu_i$  involved are the ones from  $-b$  to  $b$ ):



**Fig. 8.** Demonstration of the use of the kernels shown in Fig. 7 for the analysis of orientation and position of edges and junctions. For each pixel in a  $16 \times 16$  neighbourhood of the T-junction in Fig. 7 (right) the local maxima in orientation of the modulus of corresponding filter responses are shown. **A – 2-sided:** (equivalent to Fig. 2 top-right) Within a distance of approximately  $2 - 2.5\sigma_y$  from an isolated edge this kernel gives an accurate estimate of edge orientation. Near the junction there is a distortion in the estimate of orientation; notice that the needles indicating the orientation of the horizontal edge bend clockwise by approximately  $15^\circ$  within a distance of approx.  $1.5\sigma_x$  from the junction. The periodicity of the maxima is  $180^\circ$  making it difficult to take a local decision about the identity of the junction (L, T, X). **B – 1-sided:** Notice the good estimate of orientation near the junction; from the disposition of the local maxima it is possible to identify the junction as a T-junction. The estimate of edge orientation near an isolated edge is worse than with the 2-sided kernel since the 1-sided kernel has a  $360^\circ$  symmetry. **C – endstop:** The response along an ‘isolated’ edge (far from the junction) is null along the orientation of the edge, while the response in the region of the junction has maxima along the directions of the intervening edges. **D – endstop along 1-sided maxima:** Response of the endstop kernel along the orientations of maximal response of the 1-sided kernel. Notice that there is significant response only in the region of the junction. The junction may be localized at the position with maximal total endstop response.

$$F_{(2S)\theta}^{[n]} = \sum_{i=1}^n \sigma_i a_i(\mathbf{x}) e^{j2\pi\theta\nu_i} = \sum_{\nu=-b}^b \sigma_\nu a_\nu(\mathbf{x}) e^{2\pi\nu\theta} = \sum_{\nu=0}^b \sigma_\nu a_\nu(\mathbf{x}) (e^{2\pi\nu\theta} + j^{2\nu} e^{-2\pi\nu\theta}) \quad (14)$$

where the indexing is now by frequency:  $a_\nu$  and  $\sigma_\nu$  denote the  $a_i$  and  $\sigma_i$  associated to the frequency  $\nu = \nu_i$ , and  $\sigma_{\nu=0} = \frac{1}{2}\sigma_i$ ,  $i = \arg(\mathbf{u}_i = 0)$ .

Consider now the endstopped kernel (Fig. 7, middle): the same symmetries are found in a different combination: the real part has symmetries (E) and  $(\Pi^-)$  while the imaginary part has symmetries (O) and  $(\Pi^+)$ . A kernel of this form may be clearly obtained from the coefficients of the 2-sided kernel exchanging the basis functions: sinusoids for the even frequencies and cosinusoids for the odd frequencies (equivalent to taking the Hilbert transform of the 2-sided kernel along the circular concentric paths):

$$F_{(ES)\theta}^{[n]}(\mathbf{x}) = \sum_{\nu=0}^b \sigma_\nu a_\nu(\mathbf{x}) (e^{2\pi\nu\theta} + j^{2\nu+1} e^{-2\pi\nu\theta}) \quad (15)$$

The endstopped kernel shown in Fig. 7 has been obtained following this procedure from the decomposition  $(\sigma_i, a_i, b_i)$  of the 2-sided kernel in the same figure.

A kernel of the form 1-sided can now be obtained by summing the 2-sided and endstopped kernels previously constructed. It is the one shown in Fig. 7, right side. The corresponding reconstruction equation is:

$$F_{(1S)\theta}^{[n]}(\mathbf{x}) = \sum_{\nu=0}^b \sigma_\nu a_\nu(\mathbf{x}) e^{2\pi\nu\theta} \quad (16)$$

## 6 Conclusions

A technique has been presented for implementing families of deformable kernels for early vision applications. A given family of kernels obtained by deforming continuously a template kernel is approximated by interpolating a finite discrete set of kernels. The technique may be applied if and only if the family of kernels involved satisfy a compactness condition. This improves upon previous work by Freeman and Adelson on steerable filters and Perona and Simoncelli et al. on scalable filters in that (a) it is formulated with maximum generality to the case of any compact deformation, or, equivalently any compact family of kernels, and (b) it provides a design technique which is guaranteed to find the most parsimonious discrete approximation. It has also been shown how to build edge-terminator- and junction-tuned kernels out of a same family of ‘basis’ function.

Unlike common techniques used in early vision where the set of orientations is discretized, here the kernel and the response of the corresponding filter may be computed in a continuum for *any* value of the deformation parameters, with no anisotropies. The approximation error is computable a priori and it is constant with respect to the deformation parameter. This allows one, for example, to recover edges with great spatial and angular accuracy.

## 7 Acknowledgements

I have had useful conversations concerning this work with Ted Adelson, Stefano Casadei, Charles Desoer, David Donoho, Peter Falb, Bill Freeman, Federico Girosi, Takis Konstantopoulos, Paul

Kube, Olaf Kübler, Jitendra Malik, Stephane Mallat, Sanjoy Mitter, Richard Murray, Massimo Porrati. Federico Girosi and Peter Falb helped with references to the functional analysis textbooks. The simulations have been carried out using Paul Kube's "viz" image-manipulation package. The images have been printed with software provided by Eero Simoncelli. Some of the simulations have been run on a workstation generously made available by prof. Canali of the Università di Padova. Part of this work was conducted while at the M.I.T.. I am very grateful to Sanjoy Mitter and the staff of LIDS for their warm year-long hospitality.

## References

1. ADELSON, E., AND BERGEN, J. Spatiotemporal energy models for the perception of motion. *J. Opt. Soc. Am.* 2, 2 (1985), 284–299.
2. ADELSON, E., AND BERGEN, J. *Computational models of visual processing*. M. Landy and J. Movshon eds. MIT press, 1991, ch. "The plenoptic function and the elements of early vision". Also appeared as MIT-MediaLab-TR148. September 1990.
3. BINFORD, T. Inferring surfaces from images. *Artificial Intelligence* 17 (1981), 205–244.
4. BOVIK, A., CLARK, M., AND GEISLER, W. Multichannel texture analysis using localized spatial filters. *IEEE trans. Pattern Anal. Mach. Intell.* 12, 1 (1990), 55–73.
5. BURT, P., AND ADELSON, E. The laplacian algorithm as a compact image code. *IEEE Transactions on Communications* 31 (1983), 532–540.
6. CANNY, J. A computational approach to edge detection. *IEEE trans. Pattern Anal. Mach. Intell.* 8 (1986), 679–698.
7. CHOQUET, G. *Lectures on analysis*, vol. I. W. A. Benjamin Inc., New York, 1969.
8. DIEUDONNE, J. *Foundations of modern analysis*. Academic Press, New York, 1969.
9. E. SIMONCELLI, W. FREEMAN, E. A., AND HEEGER, D. Shiftable multi-scale transforms. Tech. Rep. 161, MIT-Media Lab, 1991.
10. FOGEL, I., AND SAGI, D. Gabor filters as texture discriminators. *Biol. Cybern.* 61 (1989), 103–113.
11. FREEMAN, W., AND ADELSON, E. Steerable filters for early vision, image analysis and wavelet decomposition. In *Third International Conference on Computer Vision* (1990), IEEE Computer Society, pp. 406–415.
12. FREEMAN, W., AND ADELSON, E. The design and use of steerable filters for image analysis, enhancement and multi-scale representation. *IEEE trans. Pattern Anal. Mach. Intell.* (1991).
13. FREEMAN, W., AND ADELSON, E. Junction detection and classification. *Invest. Ophthalmol. Vis. Sci. (Supplement)* 32, 4 (1991), 1279.
14. GRANLUND, G. H. In search of a general picture processing operator. *Computer Graphics and Image Processing* 8 (1978), 155–173.
15. HEEGER, D. Optical flow from spatiotemporal filters. In *Proceedings of the First International Conference on Computer Vision* (1987), pp. 181–190.
16. HEITGER, F., ROSENTHALER, L., VON DER HEYDT, R., PETERHANS, E., AND KÜBLER, O. Simulation of neural contour mechanism: From single to end-stopped cells. Tech. Rep. 126, IKT/Image science lab ETH-Zuerich, 1991.
17. HORN, B. The binford-horn linefinder. Tech. rep., MIT AI Lab. Memo 285, 1971.
18. HUBEL, D., AND WIESEL, T. Receptive fields of single neurones in the cat's striate cortex. *J. Physiol. (Lond.)* 148 (1959), 574–591.
19. HUBEL, D., AND WIESEL, T. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *J. Physiol. (Lond.)* 160 (1962), 106–154.
20. JONES, D., AND MALIK, J. Computational stereopsis—beyond zero-crossings. *Invest. Ophthalmol. Vis. Sci. (Supplement)* 31, 4 (1990), 529.
21. JONES, D., AND MALIK, J. Using orientation and spatial frequency disparities to recover 3d surface shape – a computational model. *Invest. Ophthalmol. Vis. Sci. (Supplement)* 32, 4 (1991), 710.

22. KASS, M. Computing visual correspondence. In *Proceedings: Image Understanding Workshop* (McLean, Virginia, June 1983), Science Applications, Inc, pp. 54–60.
23. KNUTTSON, H., AND GRANLUND, G. H. Texture analysis using two-dimensional quadrature filters. In *Workshop on Computer Architecture for Pattern Analysis and Image Database Management* (1983), IEEE Computer Society, pp. 206–213.
24. KOENDERINK, J., AND VAN DOORN, A. Representation of local geometry in the visual system. *Biol. Cybern.* 55 (1987), 367–375.
25. MALIK, J., AND GIGUS, Z. A model for curvilinear segregation. *Invest. Ophthalmol. Vis. Sci. (Supplement)* 32, 4 (1991), 715.
26. MALIK, J., AND PERONA, P. Preattentive texture discrimination with early vision mechanisms. *Journal of the Optical Society of America – A* 7, 5 (1990), 923–932.
27. MORRONE, M., AND BURR, D. Feature detection in human vision: a phase dependent energy model. *Proc. R. Soc. Lond. B* 235 (1988), 221–245.
28. MORRONE, M., AND BURR, D. *Robots and biological systems*. Academic Press, eds. P. Dario and G. Sandini, 1990, ch. : “A model of human feature detection based on matched filters”.
29. MORRONE, M., BURR, D., ROSS, J., AND OWENS, R. Mach bands depend on spatial phase. *Nature*, 324 (1986), 250–253.
30. MORRONE, M., AND OWENS, R. Feature detection from local energy. *Pattern Recognition Letters* 6 (1987), 303–313.
31. PARENT, P., AND ZUCKER, S. Trace inference, curvature consistency, and curve detection. *IEEE trans. Pattern Anal. Mach. Intell.* 11, 8 (1989), 823–839.
32. PERONA, P. Finite representation of deformable functions. Tech. Rep. 90-034, International Computer Science Institute, 1947 Center st., Berkeley CA 94704, 1990.
33. PERONA, P. Deformable kernels for early vision. *IEEE Conference on Computer Vision and Pattern Recognition* (June 1991), 222–227.
34. PERONA, P. Deformable kernels for early vision. Tech. Rep. 2039, LIDS-MIT, October 1991. Submitted to IEEE PAMI.
35. PERONA, P., AND MALIK, J. Detecting and localizing edges composed of steps, peaks and roofs. Tech. Rep. UCB/CSD 90/590, Computer Science Division (EECS), U.C.Berkeley, 1990.
36. PERONA, P., AND MALIK, J. Detecting and localizing edges composed of steps, peaks and roofs. In *Proceedings of the Third International Conference of Computer Vision* (Osaka, 1990), IEEE Computer Society, pp. 52–57.
37. PINKUS, A. *n-Widths in Approximation Theory*. Springer Verlag, 1985.
38. PRESS, W., FLANNERY, B., TEUKOLSKY, S., AND VETTERLING, W. *Numerical Recipes in C*. Cambridge University Press, 1988.
39. ROSENTHALER, L., HEITGER, F., VON DER HEYDT, R., AND KÜBLER, O. Detecting general edges and keypoints. Tech. Rep. 130, IKT/Image science lab ETH-Zuerich, 1991.
40. TURNER, M. Texture discrimination by gabor functions. *Biol. Cybern.* 55 (1986), 71–82.
41. ZHONG, S., AND MALLAT, S. Compact image representation from multiscale edges. In *Proceedings of the Third International Conference of Computer Vision* (Osaka, 1990), IEEE Computer Society.