2007 ICSPC International Conference on Signal Processing and Communications

TA-P1: Image and Video Processing 6

STEERABLE FILTERS GENERATED WITH THE HYPERCOMPLEX DUAL-TREE WAVELET TRANSFORM

J. Wedekind, B. Amavasai, K. Dutton[a]

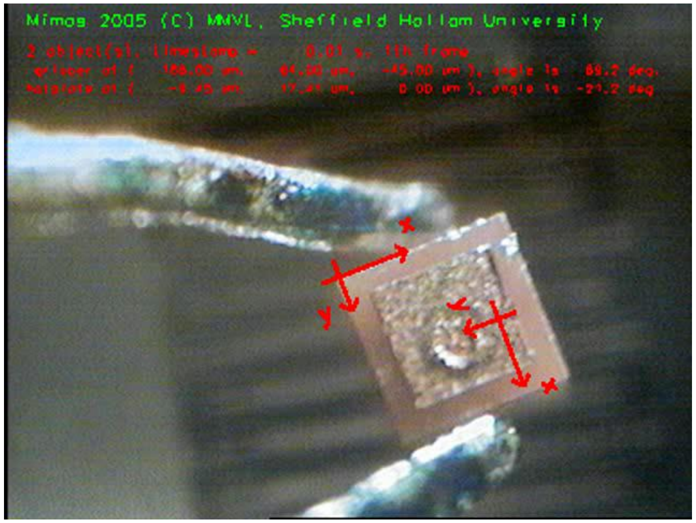
Tuesday, November 27th 2007

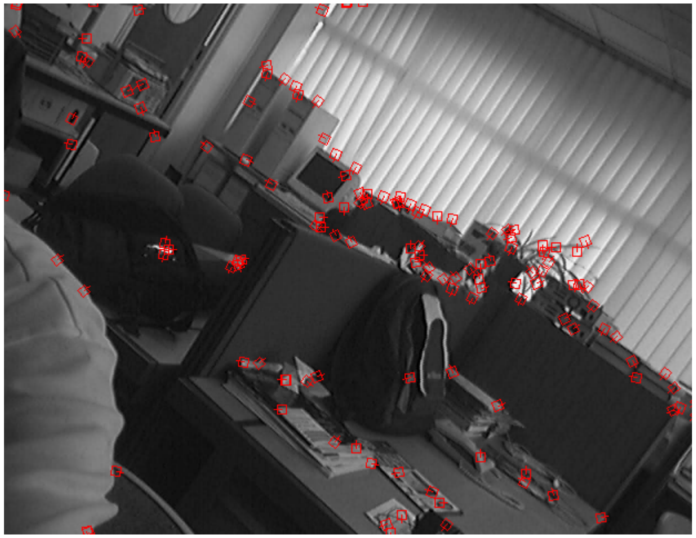
Microsystems and Machine Vision Laboratory
Materials Engineering Research Institute
Sheffield Hallam University
Sheffield
United Kingdom

[a] This project was supported by the Nanorobotics EPSRC Basic Technology grant GR/S85696/01

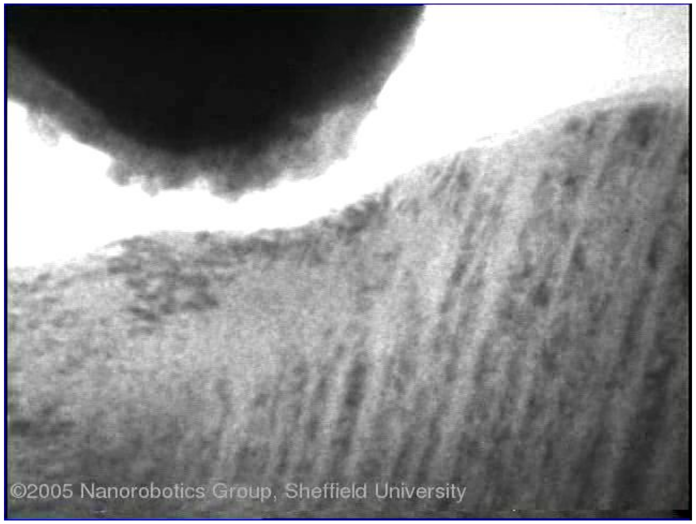
motivation
feature extraction

Micron


Mimos 2005 (C) MMVL, Sheffield Hallam University
2 object(s), timestamp = 0.01 s, 1th frame
gripper at (168.00 um, 64.00 um, -45.00 um), angle is 69.2 deg.
hotplate at (-9.45 um, 17.41 um, 0.00 um), angle is -21.2 deg
x
y
x
y



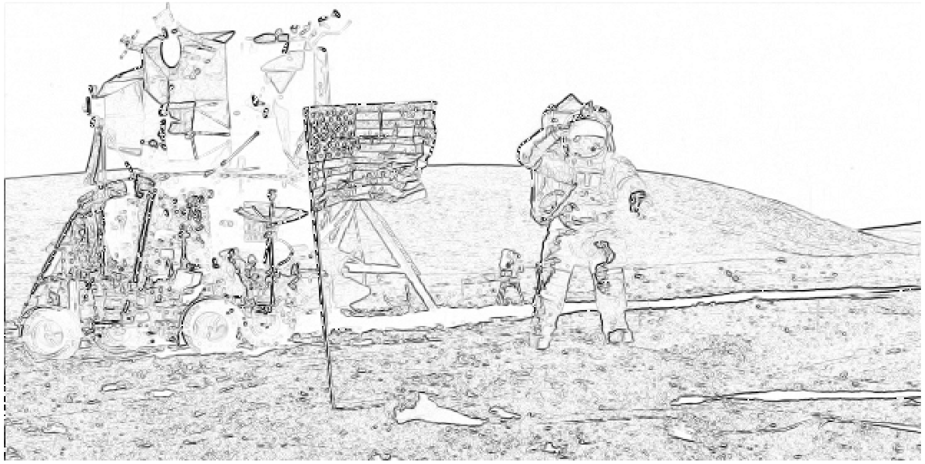
Nanorobotics



1. keypoint selection
   - rotation
   - translation
   - blur/scale
2. feature descriptor
3. RANSAC,BHT

state of the art
keypoint selection

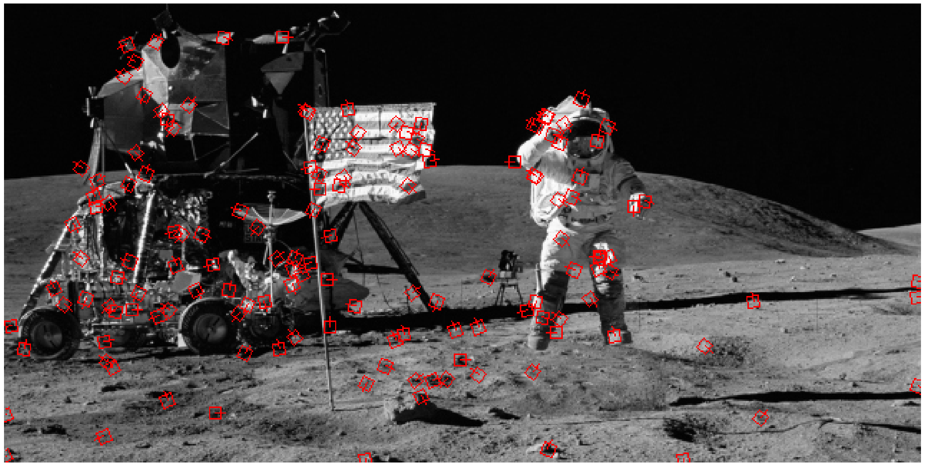
Roberts cross



Kanade-Lucas-Tomasi

Harris-Stephens

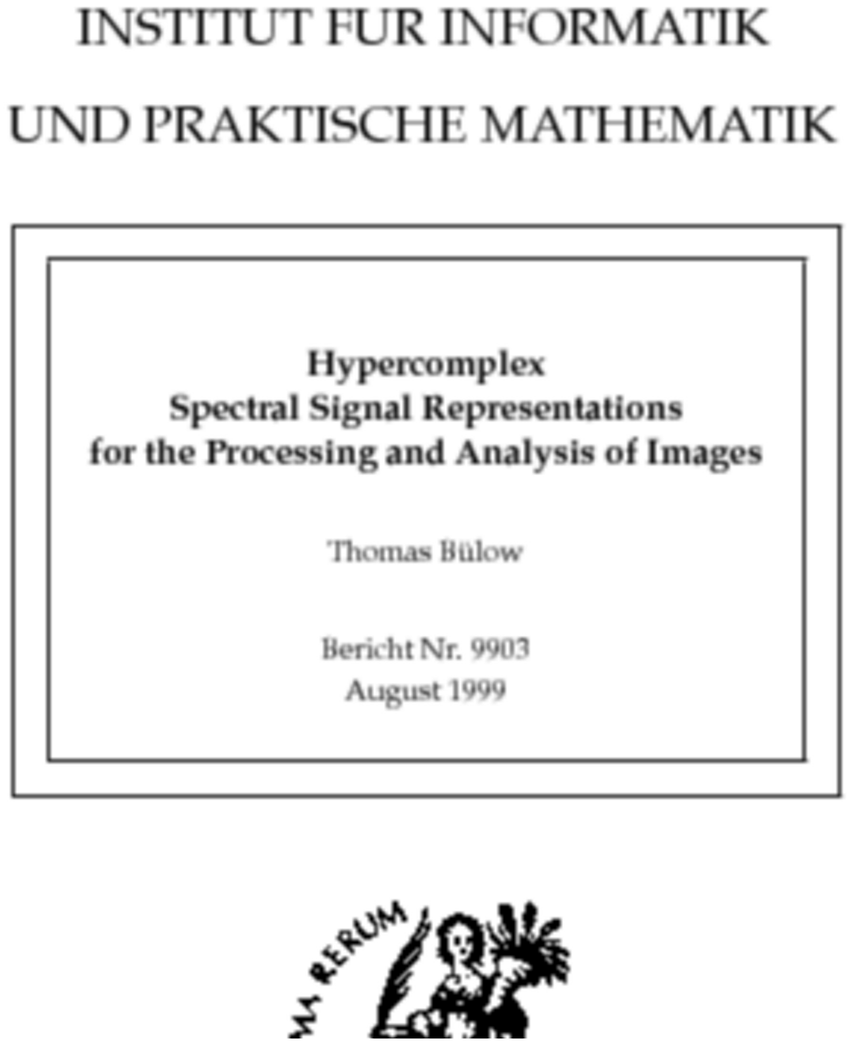
SIFT (Lowe), MOPS (Brown et al.)



next generation feature extractor?

Thomas Bülow

Hypercomplex Spectral Signal Representations for Image Processing and Analysis



INSTITUT FUR INFORMATIK

UND PRAKTISCHE MATHEMATIK

Hypercomplex
Spectral Signal Representations
for the Processing and Analysis of Images

Thomas Bülow

Bericht Nr. 9903
August 1999

MATERIALS AND ENGINEERING
RESEARCH INSTITUTE

Ivan Selesnick

The design of approximate Hilbert transform pairs of wavelet bases

The Design of Approximate Hilbert Transform Pairs of Wavelet Bases

Ivan W. Selesnick, *Member, IEEE*

Abstract—Several authors have demonstrated that significant improvements can be obtained in wavelet-based signal processing by utilizing a pair of wavelet transforms where the wavelets form a Hilbert transform pair. This paper describes design procedures, based on spectral factorization, for the design of pairs of dyadic wavelet bases where the two wavelets form an approximate Hilbert transform pair. Both orthogonal and biorthogonal FIR solutions are presented, as well as IIR solutions. In each case, the solution depends on an allpass filter having a flat delay response. The design procedure allows for an arbitrary number of vanishing wavelet moments to be specified. A Matlab program for the procedure is given, and examples are also given to illustrate the results.

Index Terms—Dual-tree complex wavelet transform, Hilbert transform, wavelet transforms.

I. Introduction

THIS paper describes design procedures, based on spectral factorization, for the design of pairs of dyadic wavelet bases where the two wavelets form an approximate Hilbert transform pair. Several authors have advocated the simultaneous use of two wavelet transforms where the wavelets are so related. For example, Abry and Flandrin suggested using a Hilbert pair of wavelets for transient detection [2] and turbulence analysis [1]. Ozturk *et al.* suggested it for waveform encoding [14]. They are also useful for implementing complex and directional wavelet transforms. Freeman and Adelson employ the Hilbert transform in the development of steerable filters [5], [20]. Kingsbury's complex dual-tree DWT [8], [9] is based on (approximate) Hilbert pairs of wavelets. The steerable pyramid and the dual-tree DWT have numerous benefits, including improved denoising capability and the fact that they are both directional and nearly shift-invariant. The paper by Beylkin and Torrésani [3] is also of related interest.

One could start with a known wavelet and then take its Hilbert transform to obtain the second wavelet; however, in that case

the scaling filters should be offset by a half sample. In [18], a design problem was formulated for the minimal length scaling filters such that 1) the wavelets each have a specified number of vanishing moments [illegible], and 2) the half-sample delay approximation is flat at $\omega = 0$ with specified degree [illegible]. However, this formulation leads to nonlinear design equations, and the examples in [18] had to be obtained using Gröbner bases. In this paper, we describe a design procedure based on spectral factorization. It results in filters similar to those of [18]; however, the design algorithm is much simpler and more flexible.

A. *Preliminaries*

Let the filters [illegible] represent a CQF pair [22]. That is

$$\sum_{n} [illegible]$$

and [illegible], where [illegible] is an odd integer. Equivalently, in terms of the Z-transform, we have

$$[illegible] = 2$$

and

$$[illegible]$$

Let the filters [illegible] represent a second CQF pair. In this paper, we assume [illegible] are real-valued filters. It is convenient to write the CQF condition in terms of the autocorrelation functions defined as

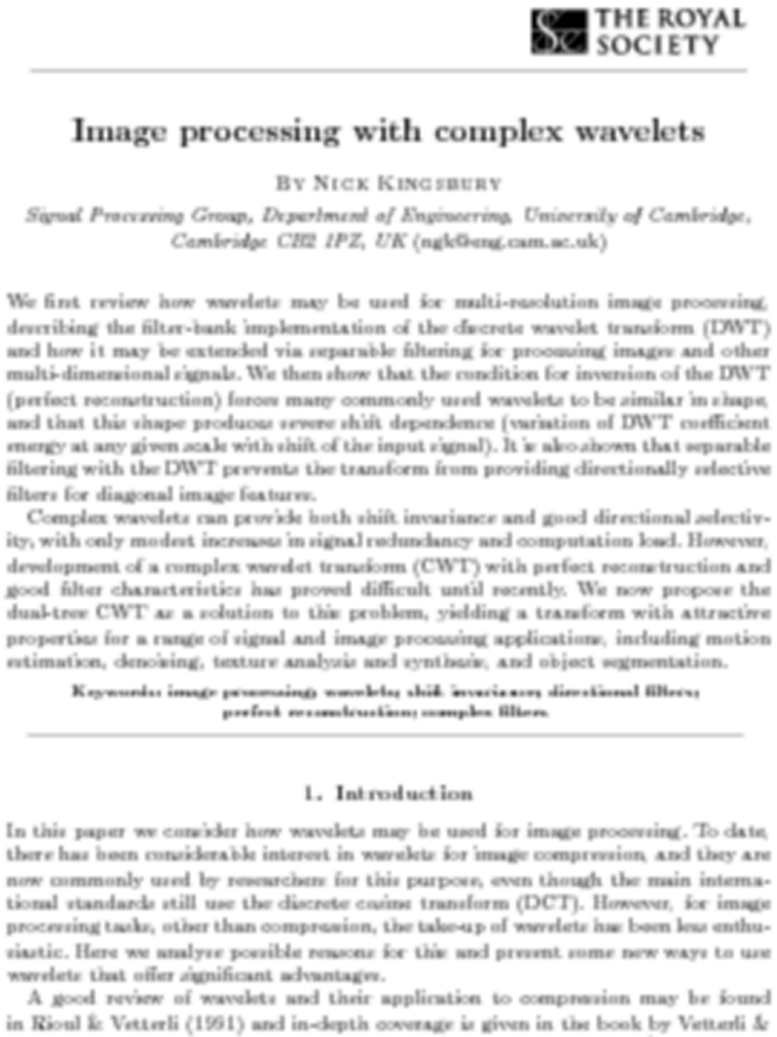
$$[illegible] := \sum_{n} [illegible]$$

$$[illegible] := \sum_{n} [illegible]$$

Sheffield
Hallam University

Nick Kingsbury

Image processing with complex wavelets



THE ROYAL SOCIETY

Image processing with complex wavelets

By Nick Kingsbury

Signal Processing Group, Department of Engineering, University of Cambridge, Cambridge CB2 1PZ, UK (ngk@eng.cam.ac.uk)

We first review how wavelets may be used for multi-resolution image processing, describing the filter-bank implementation of the discrete wavelet transform (DWT) and how it may be extended via separable filtering for processing images and other multi-dimensional signals. We then show that the condition for inversion of the DWT (perfect reconstruction) forces many commonly used wavelets to be similar in shape, and that this shape produces severe shift dependence (variation of DWT coefficient energy at any given scale with shift of the input signal). It is also shown that separable filtering with the DWT prevents the transform from providing directionally selective filters for diagonal image features.

Complex wavelets can provide both shift invariance and good directional selectivity, with only modest increases in signal redundancy and computation load. However, development of a complex wavelet transform (CWT) with perfect reconstruction and good filter characteristics has proved difficult until recently. We now propose the dual-tree CWT as a solution to this problem, yielding a transform with attractive properties for a range of signal and image processing applications, including motion estimation, denoising, texture analysis and synthesis, and object segmentation.

Keywords: image processing; wavelets; shift invariance; directional filters; perfect reconstruction; complex filters

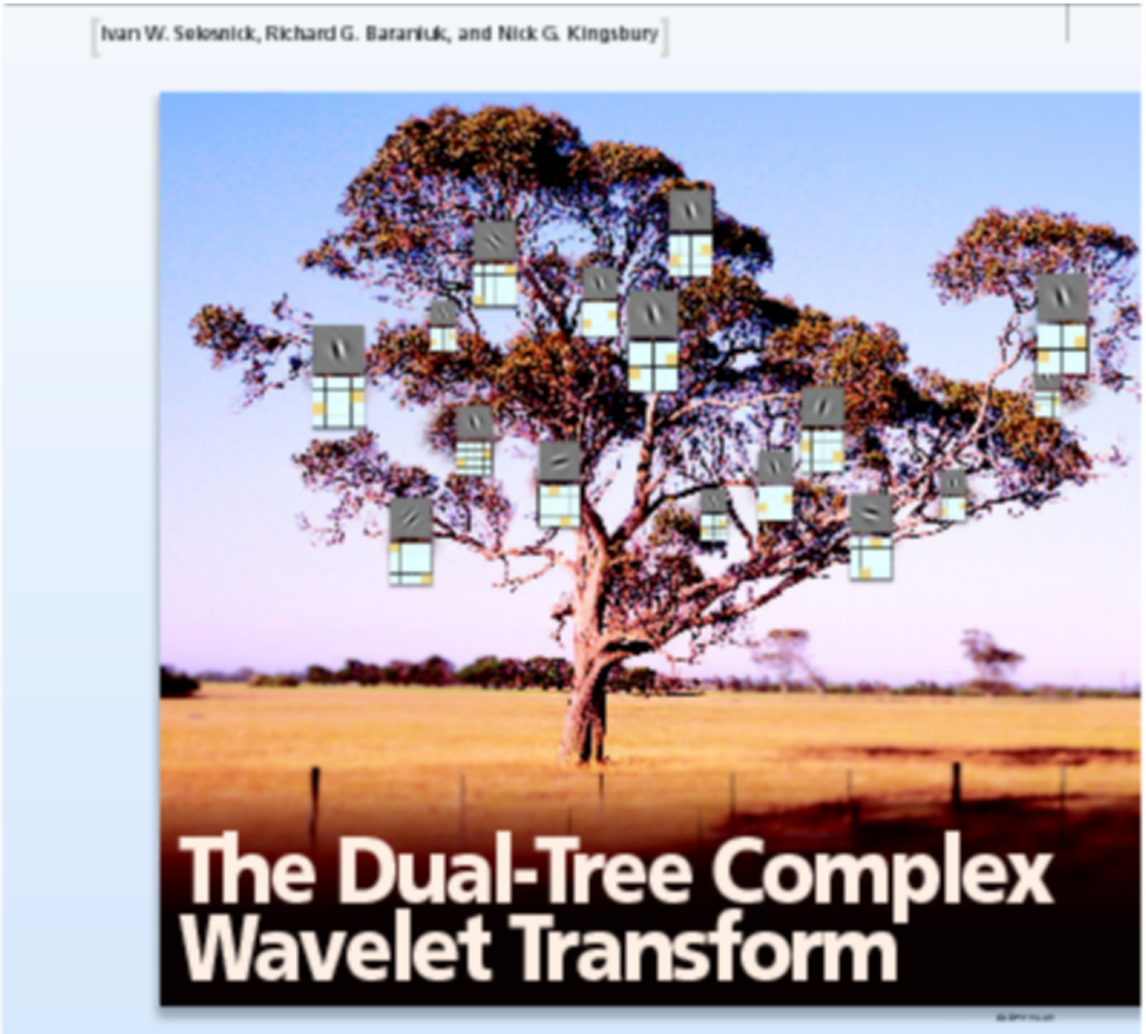
1. Introduction

In this paper we consider how wavelets may be used for image processing. To date, there has been considerable interest in wavelets for image compression, and they are now commonly used by researchers for this purpose, even though the main international standards still use the discrete cosine transform (DCT). However, for image processing tasks, other than compression, the take-up of wavelets has been less enthusiastic. Here we analyse possible reasons for this and present some new ways to use wavelets that offer significant advantages.

A good review of wavelets and their application to compression may be found in Rioul & Vetterli (1991) and in-depth coverage is given in the book by Vetterli &

Ivan Selesnick, Richard G. Baraniuk, Nick Kingsbury

The dual-tree complex wavelet transform

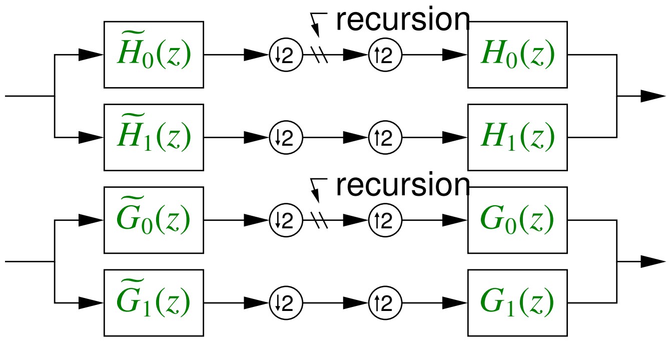


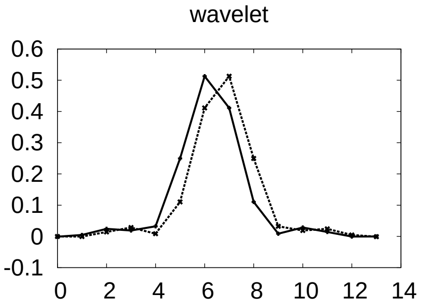
Ivan W. Selesnick, Richard G. Baraniuk, and Nick G. Kingsbury

The Dual-Tree Complex Wavelet Transform

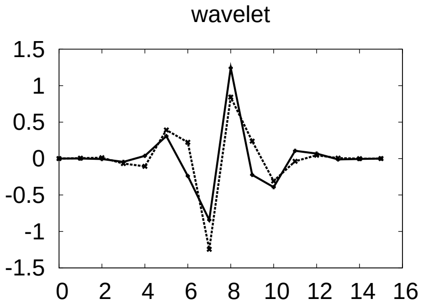
1-D basis wavelets (Selesnick)

wavelets for $K = \widetilde{K} = 5$ and $L = 4$

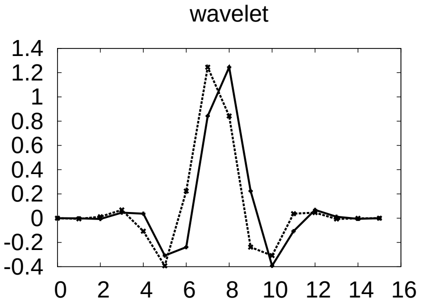




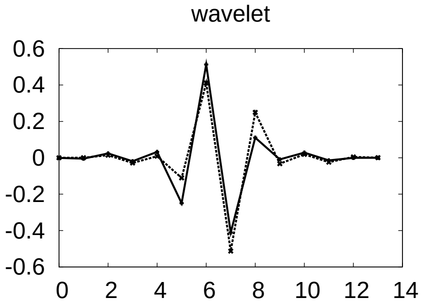
H_0, G_0



H_1, G_1



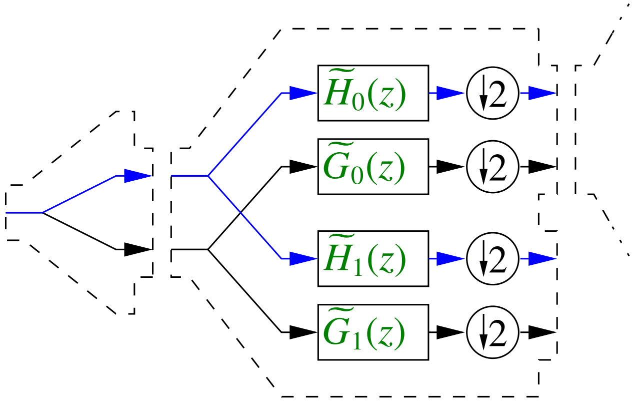
$\widetilde{H}_0, \widetilde{G}_0$



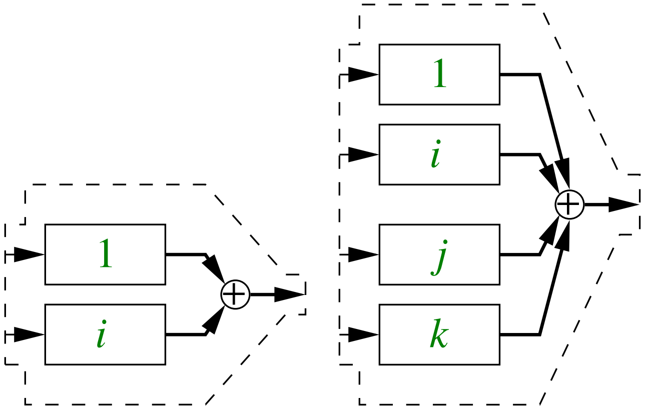
$\widetilde{H}_1, \widetilde{G}_1$

dual-tree complex wavelet transform (Kingsbury)

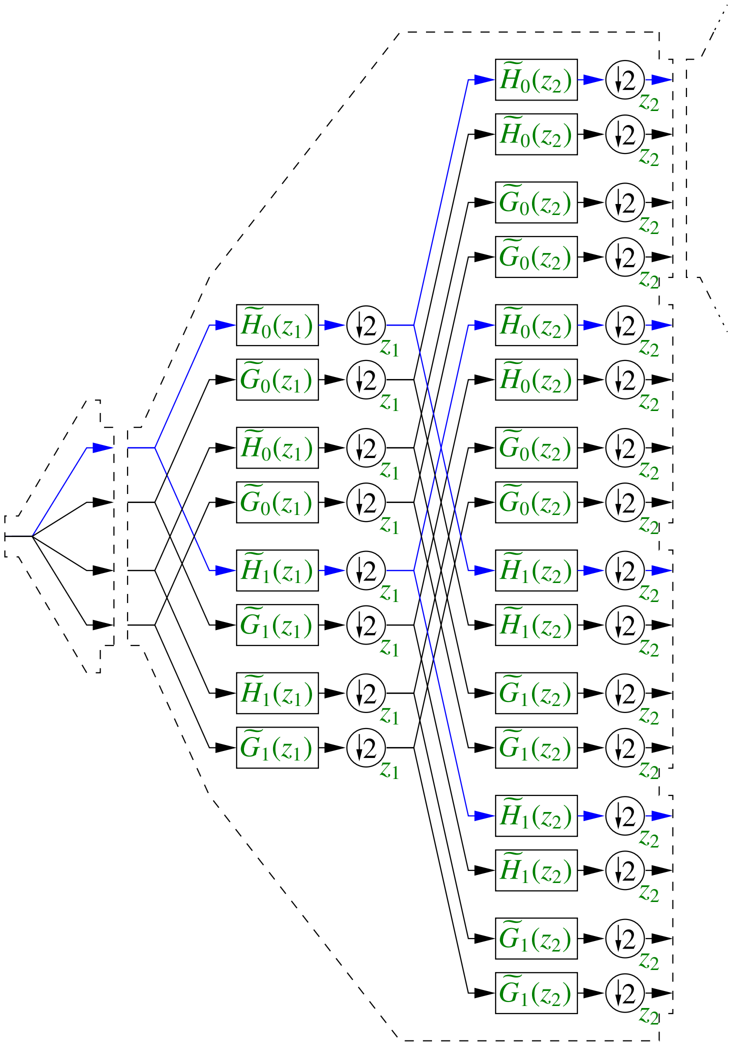
1-d



(hyper)complex representation



2-d

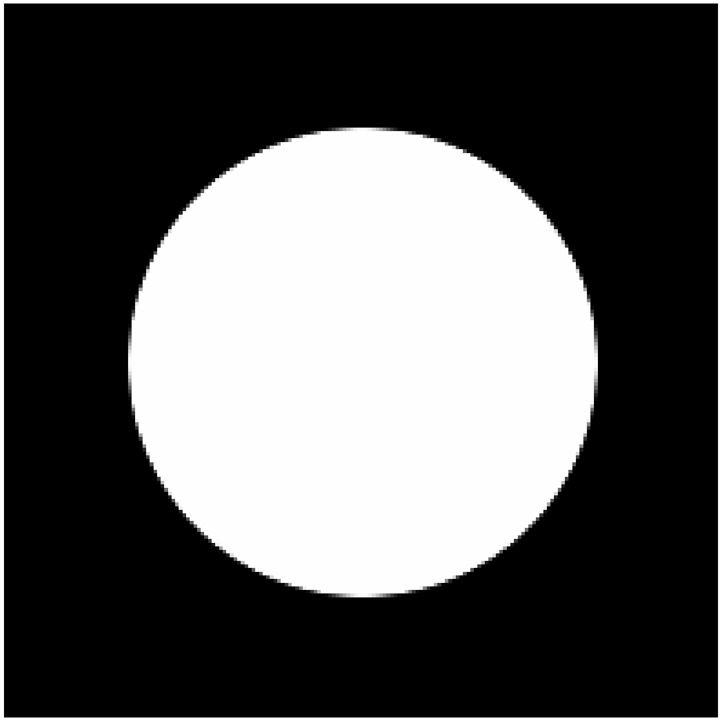


commutative hypercomplex algebra

multiplication table of $\mathbb{H}$

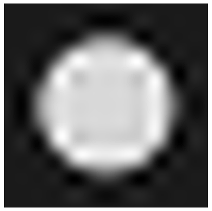
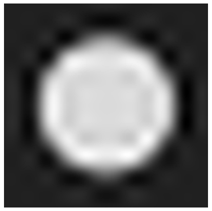
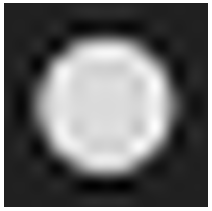
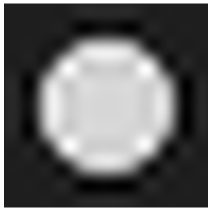
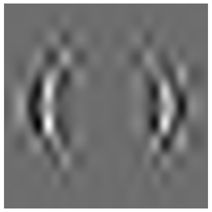
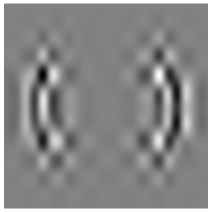
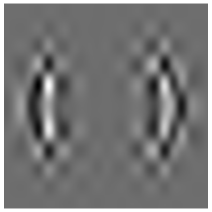
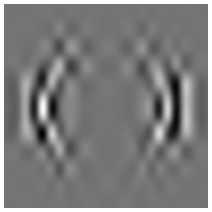
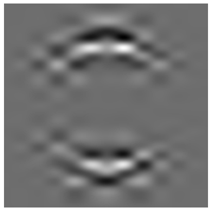
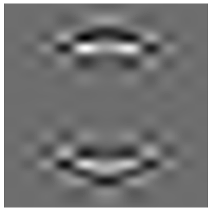
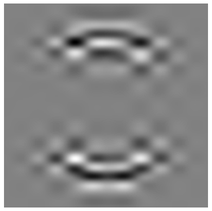
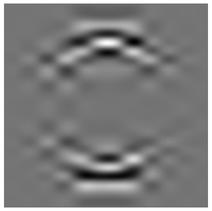
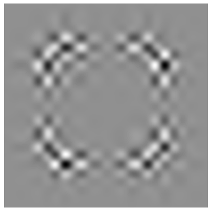
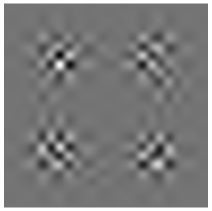
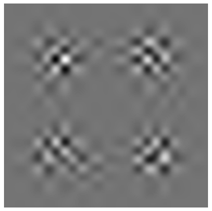
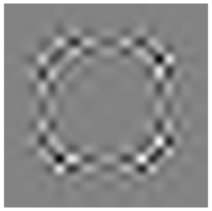
	$\mathbf{1}$	$\mathbf{i}$	$\mathbf{j}$	$\mathbf{k}$
$\mathbf{1}$	1	i	j	k
$\mathbf{i}$	i	-1	k	$-j$
$\mathbf{j}$	j	k	-1	$-i$
$\mathbf{k}$	k	$-j$	$-i$	1

test image

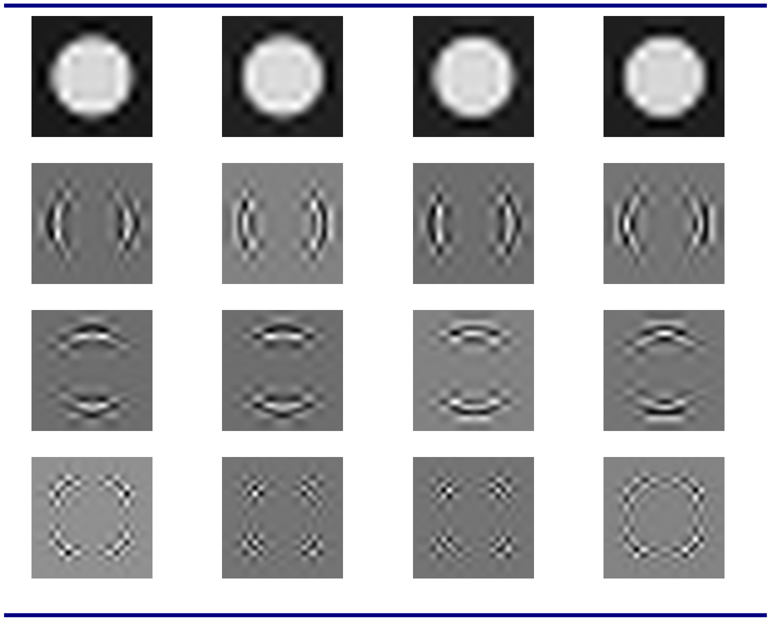


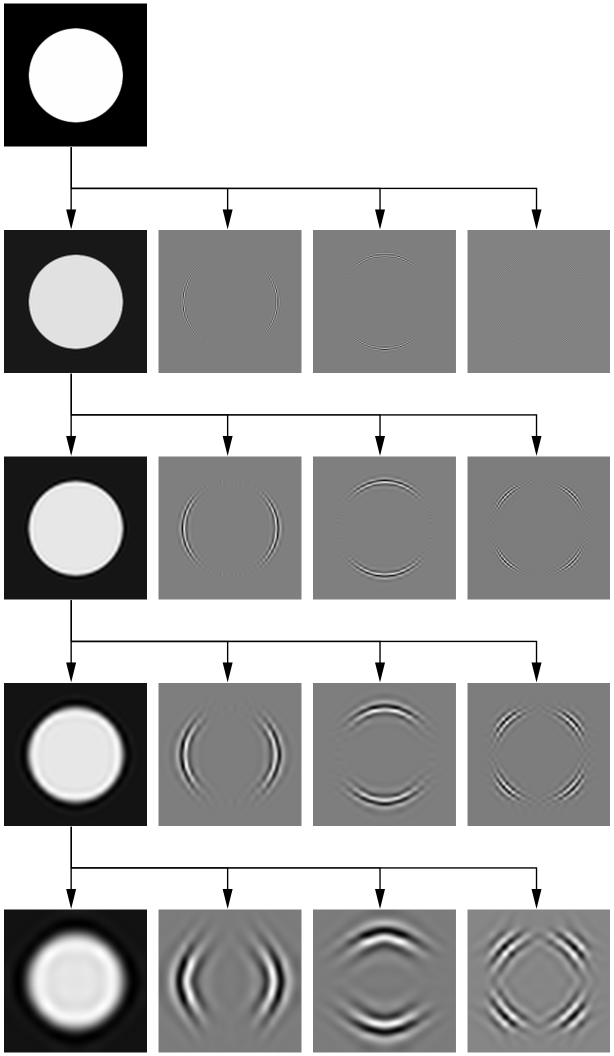
```
require 'selesnick'
require 'kingsbury'
img=MultiArray.load_grey8("circle.png")
hwt=HWT.new(MultiArray::SFLOAT)
himg=hwt.decompose(hwt.prepare(img),3)
img2=hwt.finalise(hwt.compose(himg,3))
diff=img2-img
diff.range
# -0.00522037548944354..0.0319054499268532
diff.normalise.display
Math::sqrt((diff**2).sum/diff.size)
# 0.00274400669319458
```

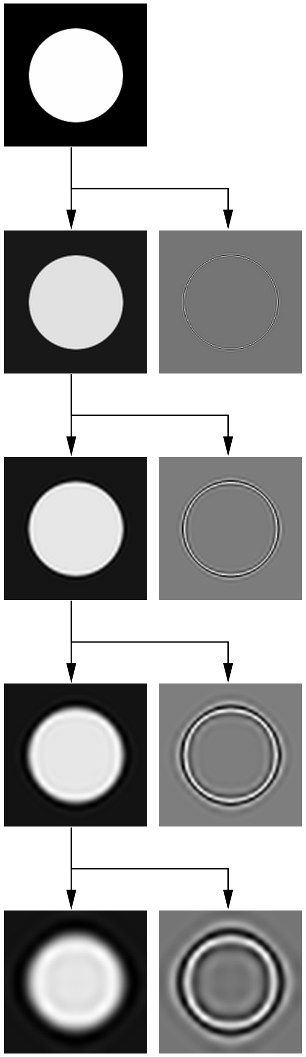
hypercomplex components

	1	i	j	k
	$A = H,\ B = H$	$A = G,\ B = H$	$A = H,\ B = G$	$A = G,\ B = G$
$A_0(z_1)\,B_0(z_2)$				
$A_1(z_1)\,B_0(z_2)$				
$A_0(z_1)\,B_1(z_2)$				
$A_1(z_1)\,B_1(z_2)$				

wavelet tree

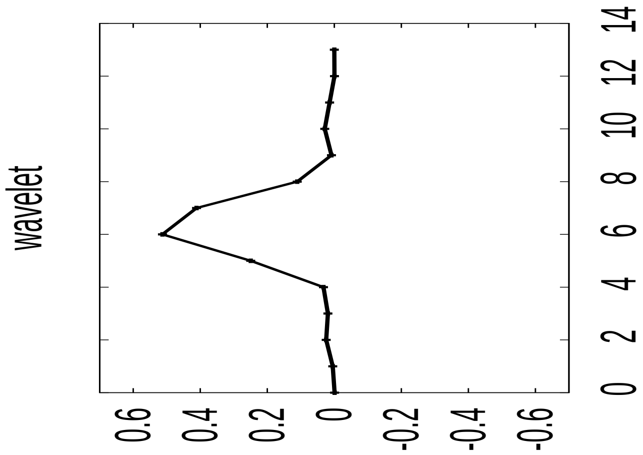


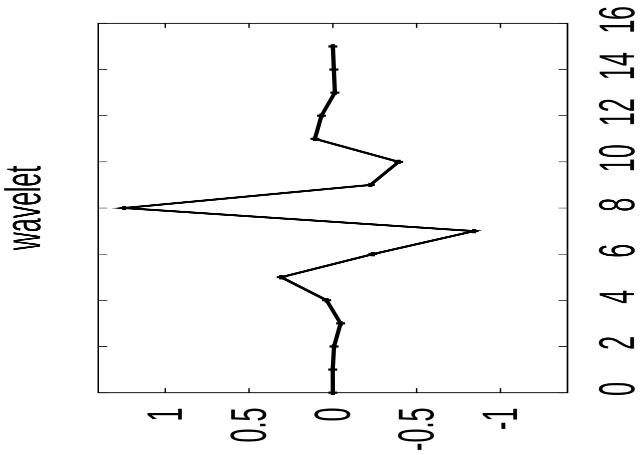




linear combinations of 1-D basis wavelets

animating Δx





$$\begin{pmatrix}1\\0\end{pmatrix} \quad \begin{pmatrix}i\\0\end{pmatrix}$$

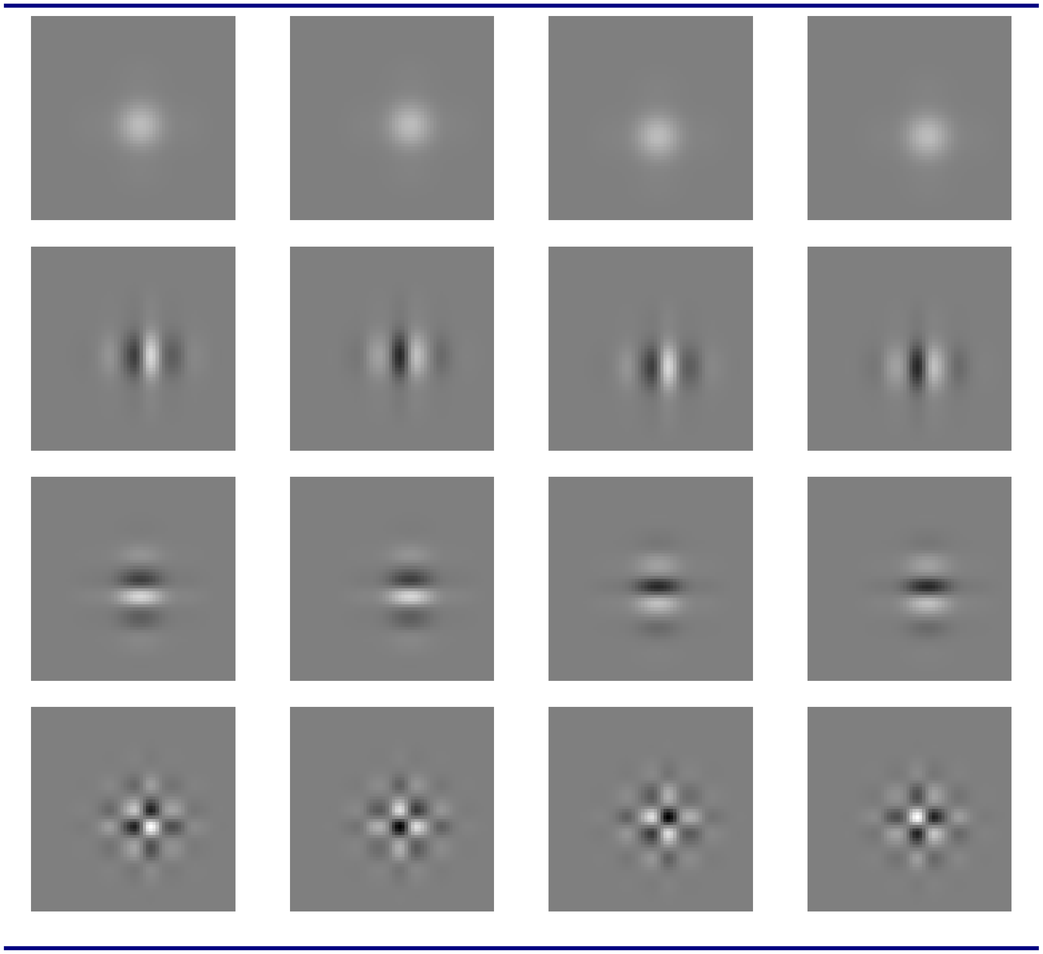
$$\begin{pmatrix}0\\1\end{pmatrix} \quad \begin{pmatrix}0\\i\end{pmatrix}$$

$$V'(z) = \sum_{a\in\{0,1\}} H_a(z)\,\mathcal{R}(v'_a) + G_a(z)\,\mathcal{I}(v'_a)$$

$$\vec{v}' = \begin{pmatrix} v'_0 \\ v'_1 \end{pmatrix} = e^{\begin{pmatrix} 2\pi\,\Delta x\, i/2 & 0 \\ 0 & 2\pi\,\Delta x\, i \end{pmatrix}} \cdot \begin{pmatrix} v_0 \\ v_1 \end{pmatrix}, \quad \vec{v} \in \mathbb{C}^2$$

animations require Javascript

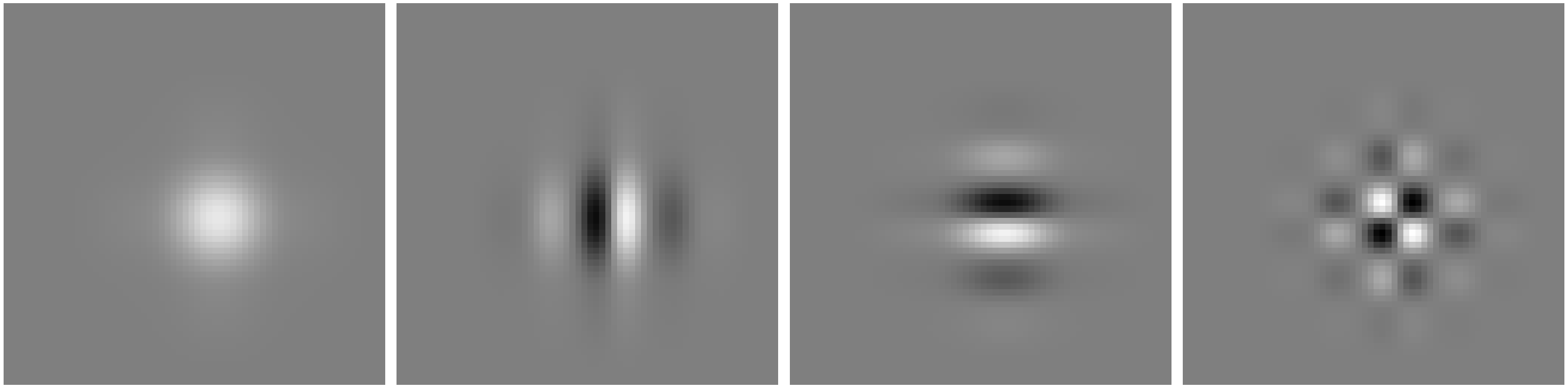
2-D separable basis wavelets



$$
\begin{array}{cccc}
\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} & \begin{pmatrix} i & 0 \\ 0 & 0 \end{pmatrix} & \begin{pmatrix} j & 0 \\ 0 & 0 \end{pmatrix} & \begin{pmatrix} k & 0 \\ 0 & 0 \end{pmatrix} \\
\begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} & \begin{pmatrix} 0 & i \\ 0 & 0 \end{pmatrix} & \begin{pmatrix} 0 & j \\ 0 & 0 \end{pmatrix} & \begin{pmatrix} 0 & k \\ 0 & 0 \end{pmatrix} \\
\begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} & \begin{pmatrix} 0 & 0 \\ i & 0 \end{pmatrix} & \begin{pmatrix} 0 & 0 \\ j & 0 \end{pmatrix} & \begin{pmatrix} 0 & 0 \\ k & 0 \end{pmatrix} \\
\begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} & \begin{pmatrix} 0 & 0 \\ 0 & i \end{pmatrix} & \begin{pmatrix} 0 & 0 \\ 0 & j \end{pmatrix} & \begin{pmatrix} 0 & 0 \\ 0 & k \end{pmatrix}
\end{array}
$$

linear combinations of 2-D basis wavelets

animation



$$
\begin{aligned}
V'(z_1, z_2) = \sum_{a,b\in\{0,1\}} & H_a(z_1)\, H_b(z_2)\, \mathcal{R}(v'_{a,b}) + \\
& G_a(z_1)\, H_b(z_2)\, \mathcal{I}(v'_{a,b}) + \\
& H_a(z_1)\, G_b(z_2)\, \mathcal{J}(v'_{a,b}) + \\
& G_a(z_1)\, G_b(z_2)\, \mathcal{K}(v'_{a,b})
\end{aligned}
$$

$$
\mathcal{V}' = e^{\begin{pmatrix} 2\,\pi\,\Delta x_2\, j/2 & 0 \\ 0 & 2\,\pi\,\Delta x_2\, j \end{pmatrix}} \cdot \mathcal{V} \cdot e^{\begin{pmatrix} 2\,\pi\,\Delta x_1\, i/2 & 0 \\ 0 & 2\,\pi\,\Delta x_1\, i \end{pmatrix}}, \quad \mathcal{V} \in \mathbb{C}^{2\times 2}
$$

animations require Javascript

coping with rotations (Kingsbury) (Bharath,Ng)

Rotation-invariant local feature matching with complex wavelets

ROTATION-INVARIANT LOCAL FEATURE MATCHING WITH COMPLEX WAVELETS

Nick Kingsbury

Signal Processing Group, Dept. of Engineering, University of Cambridge, Cambridge, CB2 1PZ, U.K.
phone: + (44) 1223 338514 email: ngk@eng.cam.ac.uk web: www.eng.cam.ac.uk/~ngk

ABSTRACT

This paper describes a technique for using dual-tree complex wavelets to obtain rich feature descriptors of keypoints in images. The main aim has been to develop a method for retaining the full phase and amplitude information from the complex wavelet coefficients at each scale, while presenting the feature descriptors in a form that allows for arbitrary rotations between the candidate and reference image patches.

1. INTRODUCTION

An important problem in image analysis is that of finding similar objects in sets of images, where the objects are often at different locations, scales and orientations in the various images. Partial occlusion of objects is also quite common. An effective general approach to this problem is first to find a relatively large number (typically several thousand) of key feature points in each image, using for example the Harris corner and edge detector [1], and then to develop a more detailed descriptor for each keypoint, which allows points from different images to be compared and matched to create candidate pairings. Often a reference object is taken from one image and then other instances of the object are searched for in the remaining images, so the number of reference keypoints is quite small (10 - 100), but the number of candidate keypoints can be very large (10^3 - 10^5). Hence it is important to develop keypoint descriptors which allow efficient comparison of pairs of keypoints (reference-to-candidate), and this is the main topic of this paper.

One of the most popular recent algorithms for this application has been Lowe's Scale Invariant Feature Transform (SIFT) [2]. In SIFT, keypoints are located by detecting extrema in a 3-D scale-space, formed by differences on a Gaus-

Like the DWT, the DTCWT is a multi-scale transform with decimated subbands, but instead of three subbands per scale in 2-D, the DTCWT has six, and each coefficient is complex (i.e. it has a real and imaginary part). Figure 1(a) shows the real and imaginary parts of the 2-D impulse responses that define the six subbands at a given scale (level 4 in this case). We see that these responses are similar to those of a 6-directional Gabor transform with orientations of $\{15°, 45°, 75°, 105°, 135°, 165°\}$, as labelled. An alternative similar transform that could be used here is Simoncelli's Steerable Pyramid [3] which has the attractive property of approximate rotational symmetry, but here we concentrate on the DTCWT because of its lower redundancy and greater computational efficiency.

A key feature of the DTCWT is that it is approximately *shift invariant*, which means that the *z*-transfer function, through any given subband of a forward and inverse DTCWT in tandem, is invariant to spatial shifts, and that aliasing effects due to decimation within the transform are small enough to be neglected for most image processing purposes [5]. A corollary of this is that the complex wavelet coefficients within any given subband are sufficiently bandlimited that we can interpolate between them in order to calculate coefficients that correctly correspond to any desired sampling location or pattern of locations. Hence for a given keypoint location we may calculate the coefficients for an arbitrary sampling pattern centred on that location. To obtain circular symmetry consistent with our 6 subband orientations, we have chosen the 13-point sampling pattern of fig. 2.

The main innovative feature of this paper is the technique for assembling complex coefficients from the 13 sampling locations, 6 subband orientations, and one or more scales, such that they form a 'polar' matching matrix *P*, in which a ro-

A steerable complex wavelet construction and its application to image denoising

A Steerable Complex Wavelet Construction and Its Application to Image Denoising

Anil Anthony Bharath and Jeffrey Ng

Abstract—This work addresses the design of a novel complex steerable wavelet construction, the generation of transform-space feature measurements associated with corner and edge presence and orientation properties, and the application of these measurements directly to image denoising. The decomposition uses pairs of bandpass filters that display symmetry and antisymmetry about a steerable axis of orientation. While the angular characterization of the bandpass filters is similar to those previously described, the radial characteristic is new, as is the manner of constructing the interpolation functions for steering. The complex filters have been engineered into a multirate system, providing a synthesis and analysis subband filtering system with good reconstruction properties. Although the performance of our proposed denoising strategy is currently below that of recently reported state-of-the-art techniques in denoising, it does compare favorably with wavelet coring approaches employing global thresholds and with an "Oracle" shrinkage technique, and presents a very promising avenue for exploring structure-based denoising in the wavelet domain.

Index Terms—Complex wavelet, corner and edge features, steerable pyramid, structure-based denoising.

NOMENCLATURE

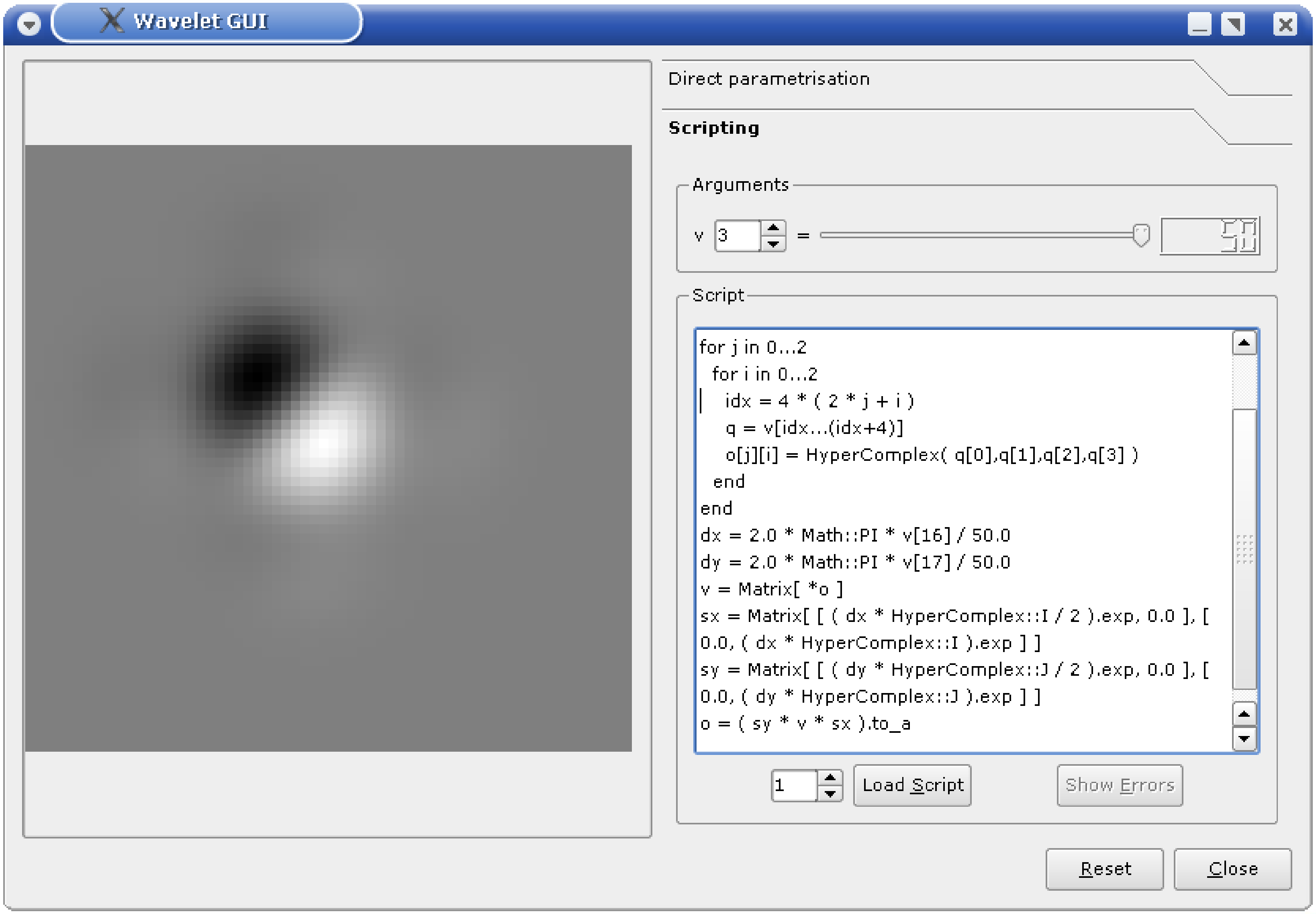
[illegible]	Bandpass analysis filter in the frequency domain in the [illegible]th direction.
[illegible]	Bandpass analysis filter in the spatial domain in the [illegible]th direction.
[illegible]	Radial specification of bandpass analysis filter.
[illegible]	Radial specification of bandpass synthesis filter.
[illegible]	Angular specification of bandpass analysis filter.
[illegible]	Low-pass analysis filter in the frequency domain.
[illegible]	Low-pass synthesis filter in the frequency domain.
[illegible]	Steered output response in direction [illegible] at level [illegible].
[illegible]	Steering function.
[illegible]	Steering function.
[illegible]	Orientation dominance complex field.
[illegible]	Orientation dominance scalar field.
[illegible]	Corner likelihood map.
[illegible]	Steering angle field.
[illegible]	Phase estimate field
[illegible]	Spatially varying gain functions for denoising.
[illegible]	Noise-dependent gain function used in denoising.

I. INTRODUCTION

THE application of wavelets to signal and image compression and to denoising is well researched. Orthogonal wavelet decompositions, based on separable, multirate filtering systems have been widely used in image and signal processing, largely for data compression. Through the wavelet shrinkage techniques of Donoho and others, orthogonal decompositions have been successfully applied to denoise images, with quite promising results [27].

As pointed out by Kingsbury [16] and Freeman [12], one of the problems of Mallat-type algorithms is the lack of shift invariance in such decompositions. A manifestation of this is that coefficient power may dramatically re-distribute itself throughout subbands when the input signal is shifted in time or in space. To address this, Simoncelli introduced the shiftable (shift-invariant subband energy) transforms, employing undecimated bandpass branches [29]. A filter is considered steerable for a certain range of translations [illegible] if there exists a function [illegible] with the property that

wavelet editor

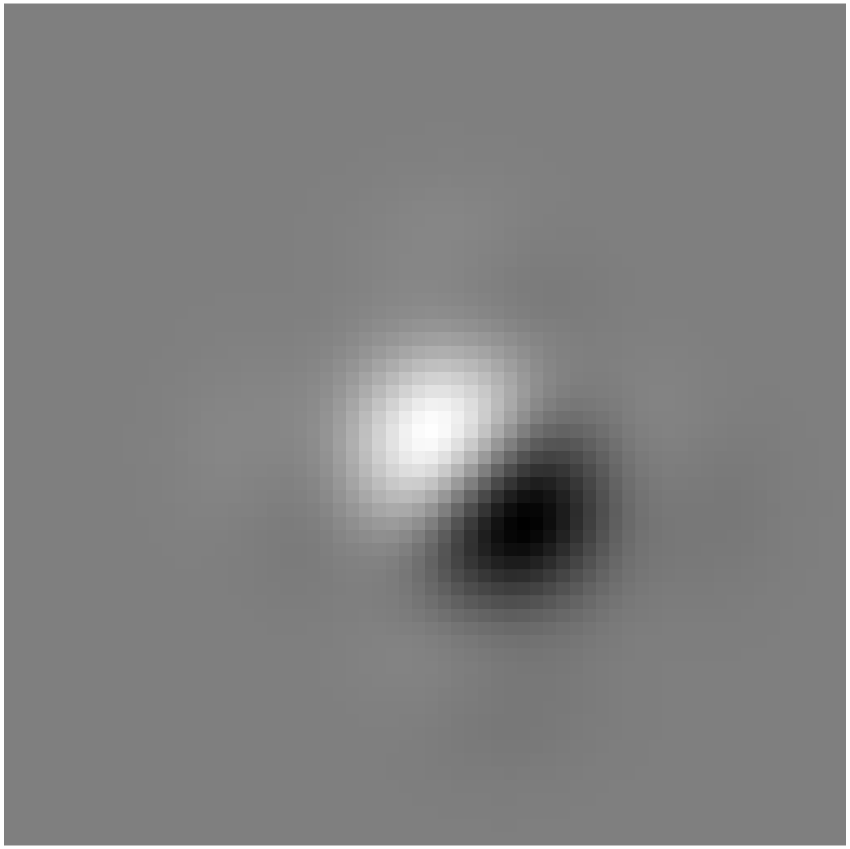


v 3 = 50

```
for j in 0...2
  for i in 0...2
    idx = 4 * ( 2 * j + i )
    q = v[idx...(idx+4)]
    o[j][i] = HyperComplex( q[0],q[1],q[2],q[3] )
  end
end
dx = 2.0 * Math::PI * v[16] / 50.0
dy = 2.0 * Math::PI * v[17] / 50.0
v = Matrix[ *o ]
sx = Matrix[ [ ( dx * HyperComplex::I / 2 ).exp, 0.0 ], [
0.0, ( dx * HyperComplex::I ).exp ] ]
sy = Matrix[ [ ( dy * HyperComplex::J / 2 ).exp, 0.0 ], [
0.0, ( dy * HyperComplex::J ).exp ] ]
o = ( sy * v * sx ).to_a
```

1

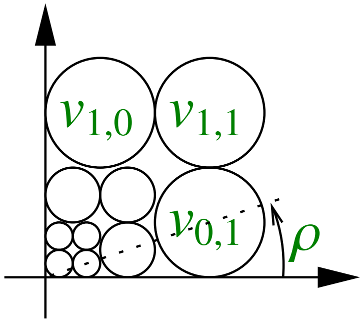
blob-like pattern

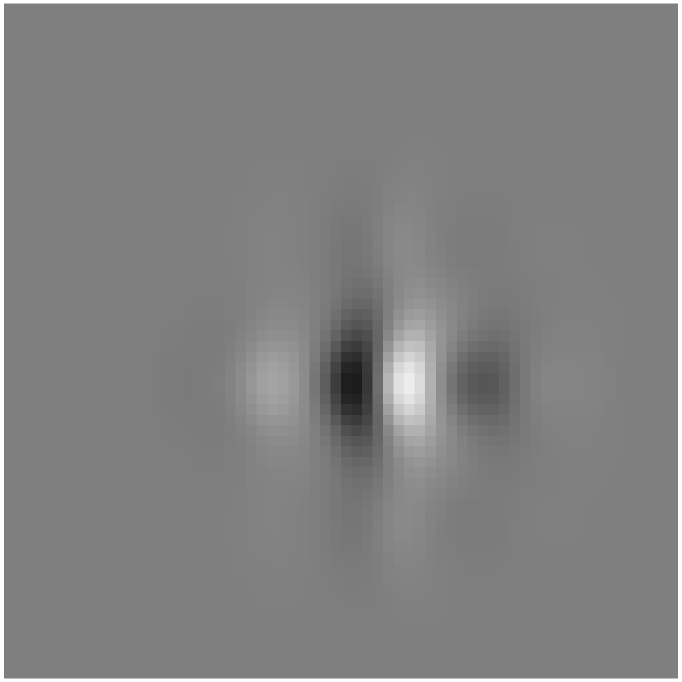


$$(1-k)\ \cos(\alpha)\begin{pmatrix}1 & 0\\ 0 & 0\end{pmatrix}+(i-j)\ \sin(\alpha)\begin{pmatrix}1 & 0\\ 0 & 0\end{pmatrix}$$

animations require Javascript

edge pattern

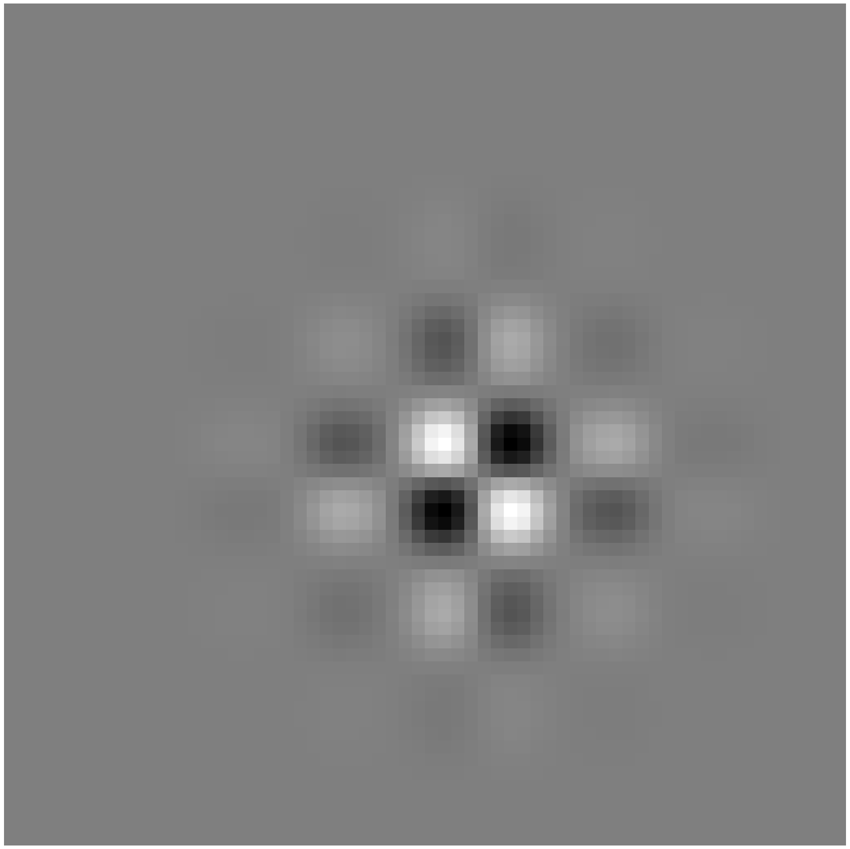




$$
\begin{aligned}
&(1+k)\ \cos(\alpha+\rho)\begin{pmatrix}0 & 1\\ 0 & 0\end{pmatrix} + (i+j)\ \cos(\alpha-\rho)\begin{pmatrix}0 & 1\\ 0 & 0\end{pmatrix} +\\
&(1+k)\ \sin(\alpha+\rho)\begin{pmatrix}0 & 0\\ 1 & 0\end{pmatrix} + (i+j)\ \sin(\alpha-\rho)\begin{pmatrix}0 & 0\\ 1 & 0\end{pmatrix} +\\
&\frac{1}{2}\,(1-i)\ \sin(\alpha)\begin{pmatrix}0 & 0\\ 0 & 1\end{pmatrix} + \frac{1}{2}\,(1-j)\ \cos(\alpha)\begin{pmatrix}0 & 0\\ 0 & 1\end{pmatrix}
\end{aligned}
$$

animations require Javascript

cross pattern



$$(1+i+j+k)\ \cos(\alpha)\begin{pmatrix}0 & 0\\ 0 & 1\end{pmatrix}+$$

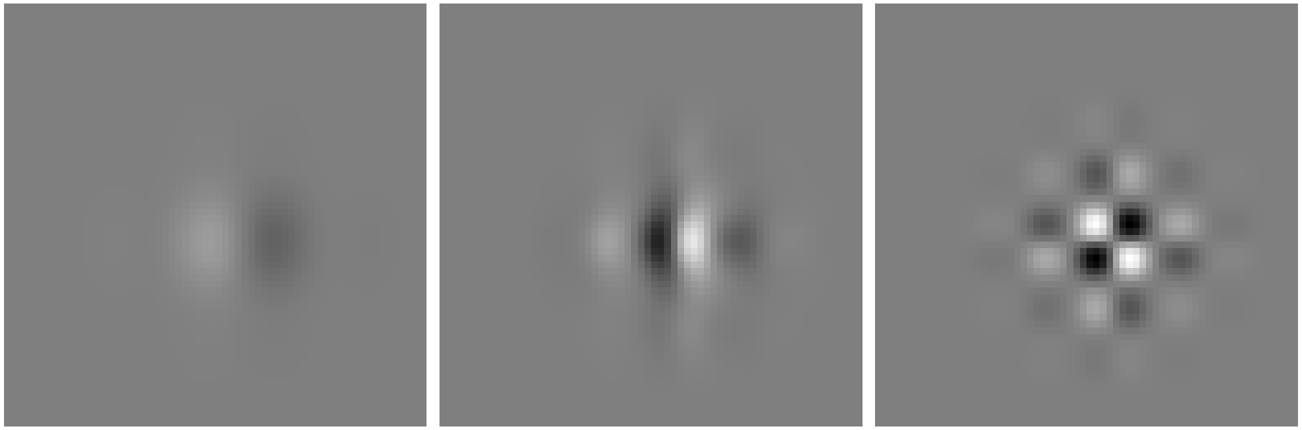
$$(1+i-j-k)\ \sin(\alpha)\begin{pmatrix}0 & 0\\ 1 & 0\end{pmatrix}+$$

$$(-1+i-j+k)\ \sin(\alpha)\begin{pmatrix}0 & 1\\ 0 & 0\end{pmatrix}$$

$$\cos(\alpha)\,\mathcal{H}_1+\sin(\alpha)\,\mathcal{H}_2,\ \text{where}\ \mathcal{H}_1,\mathcal{H}_2\in HCA^{2\times 2}$$

animations require Javascript

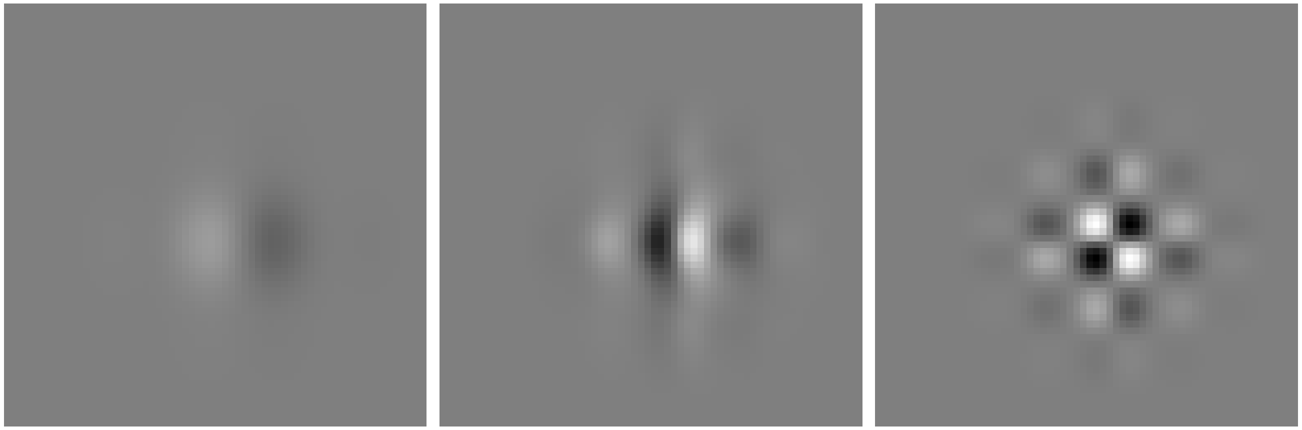
conclusion



- improved understanding of high frequency pattern
- enable use of hypercomplex wavelets beyond edge detection
- several fully steerable patterns (translation, rotation)
- hypercomplex matrices for representing local structure

animations require Javascript

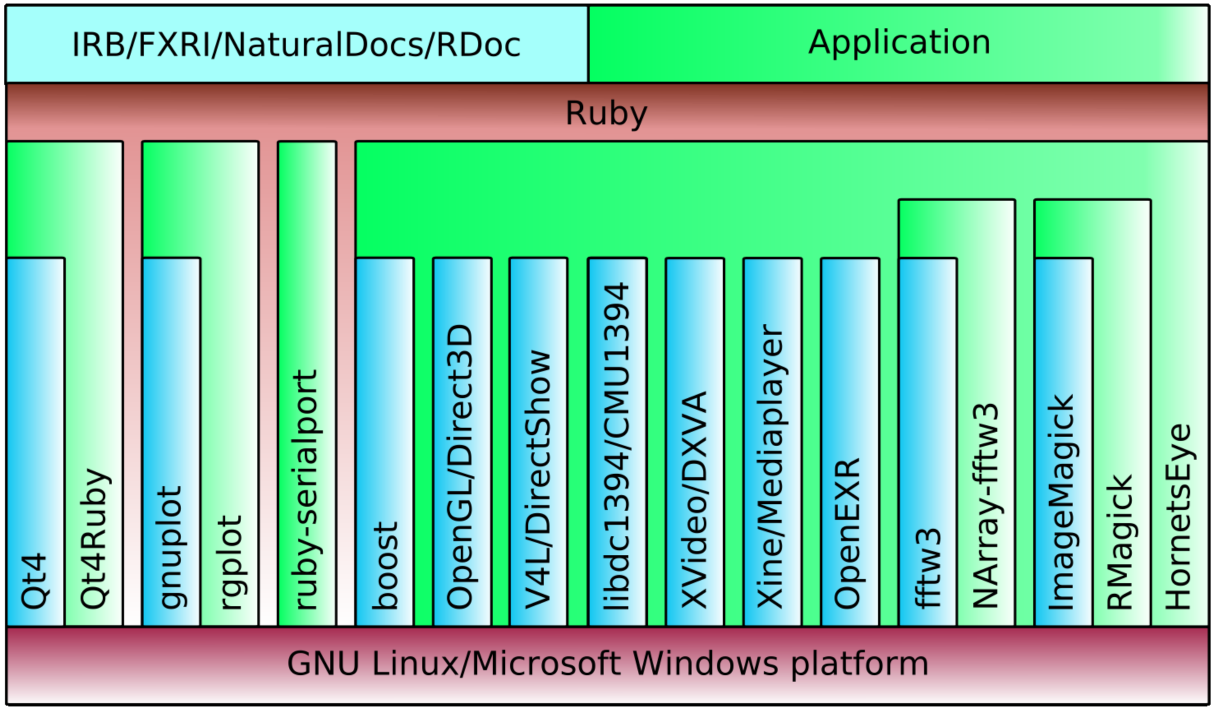
future work



- complete basis of steerable patterns? rotation around any point?
- understand interaction between neighbouring coefficients
- understand interaction of different layers of wavelet pyramid (Kingsbury)
- choose keypoints, descriptors

animations require Javascript

Hornetseye
GPL (free and open source software)



http://rubyforge.org/projects/hornetseye/

http://sourceforge.net/projects/hornetseye/

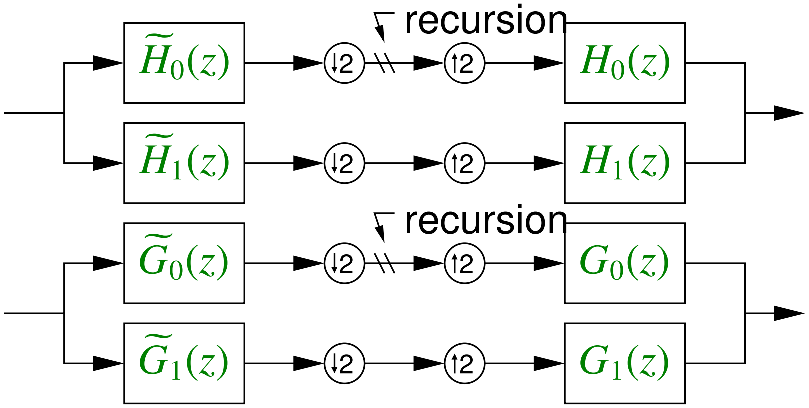
http://vision.eng.shu.ac.uk/mediawiki/index.php/Hornetseye

http://raa.ruby-lang.org/project/hornetseye/

http://www.wedesoft.demon.co.uk/hornetseye-api/

filter design (Selesnick)

perfect reconstruction



$$\tfrac{1}{2}(\widetilde{H}_0(z)X(z)+\widetilde{H}_0(-z)X(-z))H_0(z)+$$

$$\tfrac{1}{2}(\widetilde{H}_1(z)X(z)+\widetilde{H}_1(-z)X(-z))H_1(z) \overset{(*)}{=} X(z)$$

$$\Leftrightarrow$$

$$\widetilde{H}_0(z)\,H_0(-z)+\widetilde{H}_1(z)\,H_1(-z)=0 \qquad (1)$$

$$\widetilde{H}_0(z)\,H_0(z)+\widetilde{H}_1(z)\,H_1(z)=2 \qquad (2)$$

Thiran filter ($\tau = 0.5$)

$$d(n) = \binom{L-1}{n}(-1)^n \prod_{k=0}^{n-1} \frac{\tau-L+1+k}{\tau+1+k}$$

vanishing moments

$K=1$	$1 \ -1$
$K=2$	$1 \ -2 \ 1$
$K=3$	$1 \ -3 \ 3 \ -1$
$K=4$	$1 \ -4 \ 6 \ -4 \ 1$
$\dots$	$\dots$

$$\Leftarrow$$

$$H_1(z)=\widetilde{H}_0(-z)$$

$$\widetilde{H}_1(z)=-H_0(-z) \qquad (1)\checkmark$$

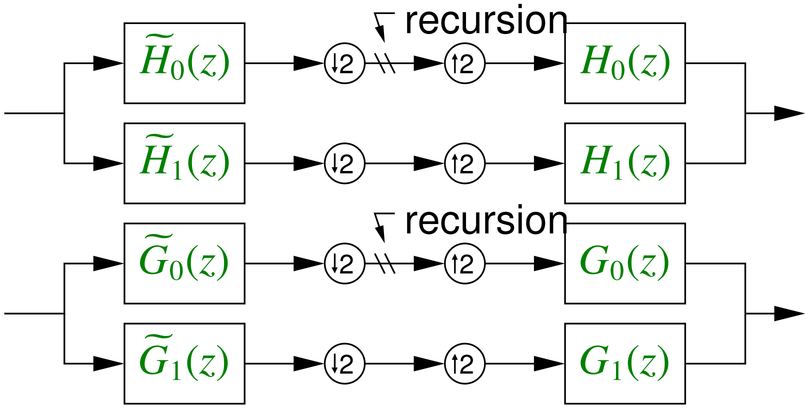
$$H_0(z) = Q(z)\,(1+z^{-1})^K\,D(z)$$

$$\widetilde{H}_0(z) = \widetilde{Q}(z)\,(1+z^{-1})^{\widetilde{K}}\,D(z^{-1})\,z^{1-L} \qquad (2)\cdots$$

$$(*)\ [\uparrow 2]\big([\downarrow 2](x(n))\big) \circ\!\!-\!\!\bullet\ \tfrac{1}{2}\,(X(z)+X(-z))$$

filter design (Selesnick)

perfect reconstruction



Thiran filter ($\tau = 0.5$)

$$d(n) = \binom{L-1}{n}(-1)^n \prod_{k=0}^{n-1} \frac{\tau-L+1+k}{\tau+1+k}$$

vanishing moments

$K = 1$	$1\ \ -1$
$K = 2$	$1\ \ -2\ \ 1$
$K = 3$	$1\ \ -3\ \ 3\ \ -1$
$K = 4$	$1\ \ -4\ \ 6\ \ -4\ \ 1$
$\dots$	$\dots$

$$\tfrac{1}{2}(\widetilde{G}_0(z)X(z) + \widetilde{G}_0(-z)X(-z))G_0(z) +$$
$$\tfrac{1}{2}(\widetilde{G}_1(z)X(z) + \widetilde{G}_1(-z)X(-z))G_1(z) \overset{(*)}{=} X(z)$$
$$\Leftrightarrow$$

$$\widetilde{G}_0(z)\,G_0(-z) + \widetilde{G}_1(z)\,G_1(-z) = 0 \qquad (3)$$
$$\widetilde{G}_0(z)\,G_0(z) + \widetilde{G}_1(z)\,G_1(z) = 2 \qquad (4)$$

$$\Leftarrow$$
$$G_1(z) = \widetilde{G}_0(-z)$$
$$\widetilde{G}_1(z) = -G_0(-z)$$
(3)✓

$$G_0(z) = Q(z)\,(1+z^{-1})^K\,D(z^{-1})\,z^{1-L}$$
$$\widetilde{G}_0(z) = \widetilde{Q}(z)\,(1+z^{-1})^{\widetilde{K}}\,D(z)$$
(4)$\cdots$

$$(*)\ [\uparrow 2]\Big([\downarrow 2](x(n))\Big) \circ\!\!-\!\!\bullet\ \tfrac{1}{2}\,(X(z) + X(-z))$$

filter design (Selesnick)

(2)···

$$\left.\begin{aligned}
&\widetilde{H}_0(z)\,H_0(z) + \widetilde{H}_1(z)\,H_1(z) = 2\\
&H_1(z) = \widetilde{H}_0(-z)\\
&\widetilde{H}_1(z) = -H_0(-z)\\
&H_0(z) = Q(z)\,(1+z^{-1})^{K}\,D(z)\\
&\widetilde{H}_0(z) = \widetilde{Q}(z)\,(1+z^{-1})^{\widetilde{K}}\,D(z^{-1})\,z^{1-L}
\end{aligned}\right\}
1 = \widetilde{H}_0(z)\,H_0(z) = \underbrace{(1+z^{-1})^{K+\widetilde{K}}\,D(z)\,D(z^{-1})\,z^{1-L}}_{=:S(z)}\;\underbrace{\widetilde{Q}(z)\,Q(z)}_{=:R(z)}$$

filter design (Selesnick)

(4)$\cdots$

$$\left.\begin{array}{l} \widetilde{G}_0(z)\,G_0(z)+\widetilde{G}_1(z)\,G_1(z)=2\\ G_1(z)=\widetilde{G}_0(-z)\\ \widetilde{G}_1(z)=-G_0(-z)\\ G_0(z)=Q(z)\,(1+z^{-1})^{K}\,D(z^{-1})\,z^{1-L}\\ \widetilde{G}_0(z)=\widetilde{Q}(z)\,(1+z^{-1})^{\widetilde{K}}\,D(z)\end{array}\right\}1=\widetilde{G}_0(z)\,G_0(z)=\underbrace{(1+z^{-1})^{K+\widetilde{K}}\,D(z)\,D(z^{-1})\,z^{1-L}}_{=:S(z)}\;\underbrace{\widetilde{Q}(z)\,Q(z)}_{=:R(z)}$$

filter design (Selesnick)

$$
\begin{array}{c}
\hphantom{S(z)\,\widetilde{Q}(z)\,Q(z) = 1 \Leftrightarrow} (2,4)\cdots \\
S\,(z)\;\underbrace{\widetilde{Q}(z)\;Q(z)}_{R(z)} = 1 \Leftrightarrow
\begin{pmatrix}
s_N & 0 & \cdots & & \\
s_{N-2} & s_{N-1} & s_N & 0 & \cdots \\
\vdots & \vdots & \vdots & \vdots & \vdots \\
\cdots & 0 & s_1 & s_2 & s_3 \\
 & & \cdots & 0 & s_1
\end{pmatrix}
\begin{pmatrix}
r_1 \\ r_2 \\ \vdots \\ r_{N-1} \\ r_N
\end{pmatrix}
=
\begin{pmatrix}
\vdots \\ 0 \\ 1 \\ 0 \\ \vdots
\end{pmatrix}
\rightsquigarrow R(z)
\end{array}
$$

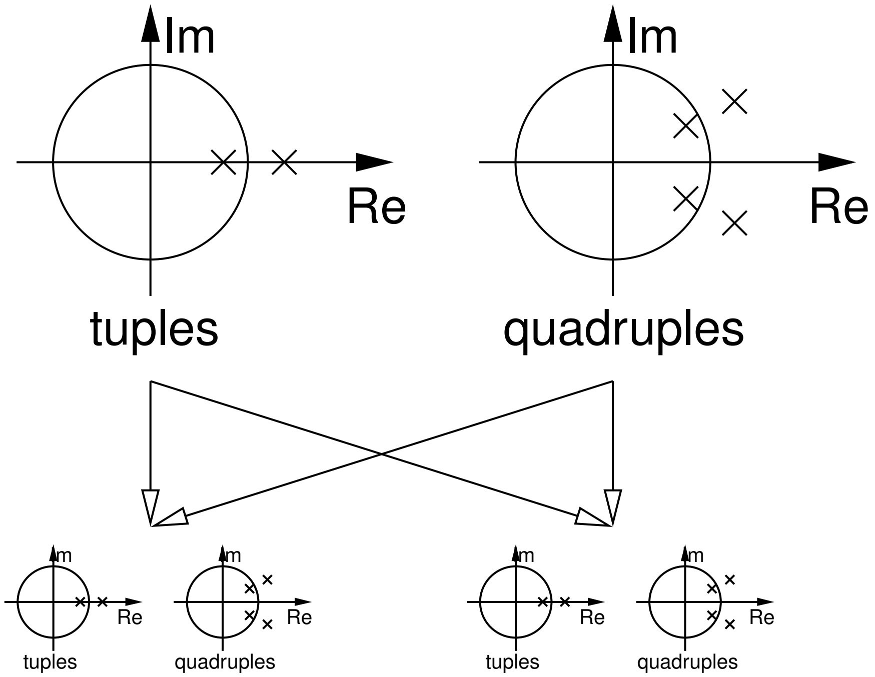
spectral factorisation (Laguerre)

Laguerre

```
Input: Polynomial $p(x)$
Output: zero crossing $o \in \mathbb{C}$ of $p$
$x \mapsto 0.0 + 0.0\,i$; $c \mapsto 100$;
while $c \geq 0$ do
    if $|p(x)|$ sufficiently small then
        return $o = x$;
    end
    $g = p'(x)/p(x)$;
    $h = g^2 - p''(x)/p(x)$;
    if $|g + d| > |g - d|$ then
        $a = n/(g + d)$;
    else
        $a = n/(g - d)$;
    end
    $x \mapsto x - a$; $c \mapsto c - 1$;
end
```

spectral factorisation

$R(z)$ symmetric, real-valued



$Q(z)$ $\widetilde{Q}(z)$

$$H_0(z) = Q(z)\,(1 + z^{-1})^K\,D(z)$$

$$\widetilde{H}_0(z) = \widetilde{Q}(z)\,(1 + z^{-1})^{\widetilde{K}}\,D(z^{-1})\,z^{1-L}$$

$$H_1(z) = \widetilde{H}_0(-z), \widetilde{H}_1(z) = -H_0(-z)$$

(2,4)✓