

An Introduction To The Monogenic Signal

Christopher P. Bridge

September 7, 2016

1 Introduction and Overview

The monogenic signal is an image analysis methodology that was introduced by Felsberg and Sommer in 2001 and has been employed for a variety of purposes in image processing and computer vision research. In particular, it has been found to be useful in the analysis of ultrasound imagery in several research scenarios mostly in work done within the BioMedIA lab at Oxford.

However, the literature on the monogenic signal can be difficult to penetrate due to the lack of a single resource to explain the various principles from basics. The purpose of this document is therefore to introduce the principles, purpose, applications, and limitations of the methodology. It assumes some background knowledge from the fields of image and signal processing, in particular a good knowledge of Fourier transforms as applied to signals and images. We will not attempt to provide a thorough mathematical description or derivation of the monogenic signal, but rather focus on developing an intuition for understanding and using the methodology and refer the reader elsewhere for a more mathematical treatment.

My MATLAB/GNU Octave implementation of the monogenic signal (used for figures in this document) is available on Github under the GNU Public License at https://github.com/CPBridge/monogenic_signal_matlab.

2 1D Signals and the Analytic Signal

The *analytic signal* [1] is a widely-used concept in signal processing concerning the analysis of 1D signals. It is essentially an alternative representation of any real-valued signal using a complex-valued signal that makes various processes (such as modulation and demodulation) conceptually simpler. It is inherently a 1D concept and therefore has no place in image analysis¹ but we will discuss the analytic signal because the concepts introduced are vital to understanding the monogenic signal, which is a generalisation of the analytic signal to multidimensional signals such as images.

Recall that the Fourier transform, $F(\omega)$, of any real-valued signal, $f(t)$, displays *Hermitian symmetry*; that is it has components at negative frequencies that are the complex conjugate of the positive frequency components:

$$F(-\omega) = \overline{F(\omega)}, \quad (1)$$

where \bar{x} denotes the complex conjugation operation on x . The analytic signal is a representation of the our original signal, $f(t)$, using *only the components at positive frequencies*: we can simply discard the negative frequency components without loss of information because there is redundancy in the Fourier transform of a real-valued signal due to the Hermitian symmetry. Clearly, however, the Hermitian symmetry of the Fourier transform is broken by this operation, meaning that the resulting time-domain analytic signal is no longer real-valued, but complex-valued. We will denote the spectrum of this analytic signal as $F_a(\omega)$, given by:

¹The analytic signal is in fact vital to the creation of ultrasound images from the reflected radio-frequency signals that are picked up by the transducer as it is used to recover the amplitude envelope. Several such demodulated signals from different scans lines are then combined to form a B-mode image. This stage of processing is, however, not the focus of the present document as we assume that all images have already gone through the standard processing chain to give human-viewable images.

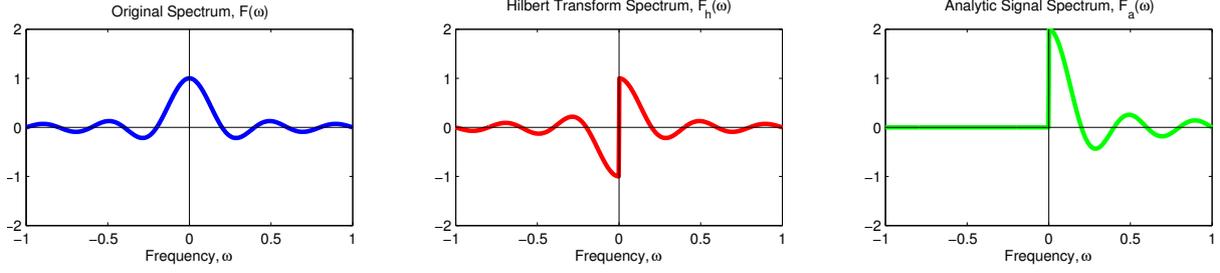


Figure 1: Formation of the analytic signal in the frequency domain (only real parts shown): the spectrum of the original, real signal (*left*) displays conjugate symmetry. The Hilbert transform of the original signal is odd symmetric (*centre*). Adding the two gives the spectrum of the analytic signal (*right*), which has no negative-frequency components.

$$F_a(\omega) = \begin{cases} 2F(\omega), & \omega > 0 \\ 0, & \omega < 0 \\ F(0), & \omega = 0. \end{cases} \quad (2)$$

Notice that we double all the positive frequency components to keep the same power of the resulting signal, and leave the zero-frequency (DC) component unchanged.

Conceptually, we can think of the operation of removing the negative frequency components from the original spectrum, $F(\omega)$, as adding a carefully constructed odd symmetric (anti-symmetric) spectrum to the spectrum of the original signal (see Figure 1).

$$F_a(\omega) = F(\omega) + F_h(\omega), \quad (3)$$

where $F_h(\omega)$ is formed by creating an ‘odd symmetric version’ of $F(\omega)$ as follows:

$$F_h(\omega) = \begin{cases} F(\omega) & \omega > 0 \\ -F(\omega) & \omega < 0 \\ 0 & \omega = 0, \end{cases} \quad (4)$$

or, equivalently, we can write using the *signum* function ($\text{sgn}(\cdot)$):

$$F_h(\omega) = F(\omega) \cdot \text{sgn}(\omega) \quad (5)$$

where

$$\text{sgn}(x) = \begin{cases} 1, & x > 0 \\ -1, & x < 0 \\ 0, & x = 0. \end{cases} \quad (6)$$

In this way we can also rewrite the relationship between a signal and its analytic signal (Equation (2)) as simple multiplication in the frequency domain:

$$F_a(\omega) = (1 + \text{sgn}(\omega)) F(\omega), \quad (7)$$

Returning to Equation (3), we have created the spectrum of the analytic signal, $F_a(\omega)$, by adding two spectra in the frequency domain. As we have discussed, the original spectrum, $F(\omega)$, is symmetric and therefore represents a *purely real* time-domain signal. Furthermore, the spectrum $F_h(\omega)$ is odd symmetric by construction and therefore represents a *purely imaginary* time-domain signal, which we will call $f_h(t)$. Therefore, in the time domain, the real part of the analytic signal is simply the original signal, and we have added a new imaginary part that ‘suppresses’ the negative frequency components (see Figure 2):

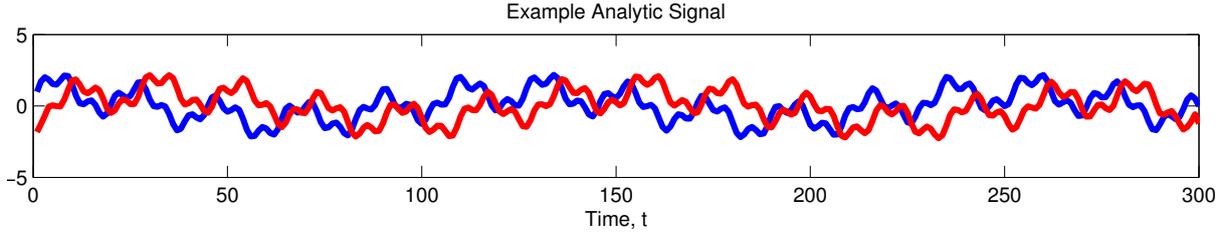


Figure 2: Example of an analytic signal. The analytic signal has a real part (*blue*) that is the same as the original signal, and an imaginary part (*red*) formed from the Hilbert transform of the original signal

$$f_a(t) = \underbrace{f(t)}_{\text{purely real}} + \underbrace{f_h(t)}_{\text{purely imaginary}} . \quad (8)$$

The process of forming the signal $f_h(t)$ from the original signal $f(t)$ is called the *Hilbert transform* and $f_h(t)$ is called the *Hilbert transform* of $f(t)$. The Hilbert transform may also be represented in the time domain by convolution with the distribution $\frac{1}{\pi t}$:

$$f_h(t) = i \cdot \text{p.v.} \int_{-\infty}^{\infty} \frac{f(\tau)}{\pi(t - \tau)} d\tau, \quad (9)$$

where p.v. is the Cauchy principal value of the integral and is necessary because of the singularity at $t = \tau$. However, it is generally simpler both conceptually and computationally to use the frequency-domain definition of the Hilbert transform as described above.

Note that an alternative definition for the Hilbert transform is $F_h(\omega) = -iF(\omega) \cdot \text{sgn}(\omega)$, where the multiplication by i means that the resulting time-domain signal is purely real and correspondingly $F_a(\omega) = F(\omega) + iF_h(\omega)$ and $f_a(t) = f(t) + if_h(t)$. This form is more useful when the Hilbert transform is considered in isolation, as the result of applying the transform to a real-value signal is a second real-valued signal.

In MATLAB, one can use the Signal Processing Toolbox function ‘hilbert’ to work with the analytic signal. Despite its name, this function returns the complex-valued analytic signal – the Hilbert transform can be found by then extracting the imaginary part of this using the ‘imag’ function.

2.1 Local Amplitude and Local Phase

Let us consider the analytic signal of a simple sinusoid of some fixed amplitude, A , and frequency, ω_0 :

$$f(t) = A \sin(\omega_0 t). \quad (10)$$

Applying the above relationships, we find that the Hilbert transform of $f(t)$ is a cosine wave, and the analytic signal is therefore a complex exponential.

$$f_h(t) = iA \cos(\omega_0 t) \quad (11)$$

$$f_a(t) = f(t) + f_h(t) \quad (12)$$

$$= A \sin(\omega_0 t) + iA \cos(\omega_0 t) \quad (13)$$

$$= Ae^{i\omega_0 t}. \quad (14)$$

In other words, the Hilbert transform has manifested itself as a *phase shift* of $-\frac{\pi}{2}$ (one quarter cycle) of the original signal (this is because a cosine wave is equivalent to a sine wave with a phase shift of $-\frac{\pi}{2}$). As Fourier theory tells us that a general signal (such as that in Figure 2) may be written as a (possibly infinite) sum of scaled and shifted sinusoids, it is straightforward to see that the Hilbert transform phase shifts *each* frequency component by $-\frac{\pi}{2}$. The signal $f(t)$ and its Hilbert transform $f_h(t)$ are said to be in *phase quadrature*.

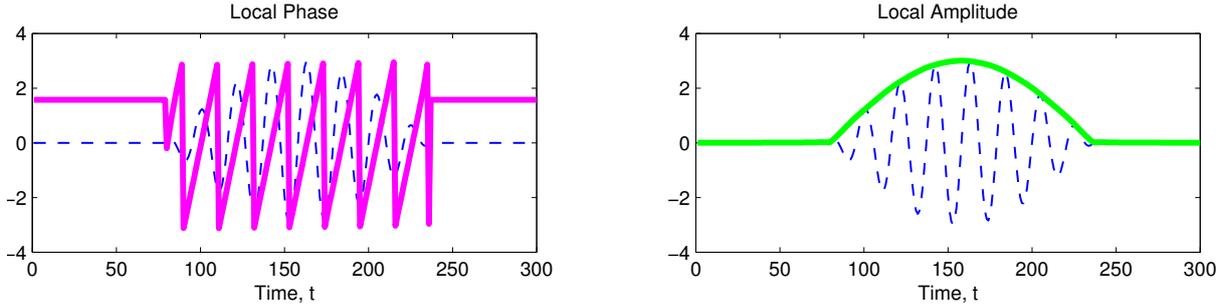


Figure 3: The local phase (*left, magenta*) and local amplitude (*right, green*) measures of a sine wave modulated by a cosine pulse (*blue, dashed*). It is apparent that the local amplitude gives the envelope of the oscillations, while the local phase gives the point in the cycle of the local oscillations.

The analytic version of our sinusoidal signal is useful to us as it allows us to extract values for the amplitude, A , and phase, $\phi = \omega_0 t$, of the signal by expressing the complex analytic signal in polar form:

$$A(t) = \sqrt{f(t)^2 + f_h(t)^2} \quad (15)$$

$$\phi(t) = \arctan\left(\frac{f_h(t)}{f(t)}\right). \quad (16)$$

However this result is not limited to simple sinusoids of fixed amplitude and constant frequency, but can in fact be applied to general signals. We can consider the analytic form of a signal to be produced by a complex exponential with time-varying amplitude and frequency:

$$f_a(t) = A(t)e^{i\phi(t)}. \quad (17)$$

This gives rise to the concepts of *local phase* and *local amplitude*, which are the phase and amplitude of this polar representation. The local amplitude is a measure of the envelope of the signal at that point, and the local phase is a measure of the shape of the signal at that point (e.g., a phase of 0 corresponds to a ‘peak’ part of a signal and a phase of π corresponds to a ‘trough’), thus we have achieved a separation of these two important characteristics of a signal (some authors refer to this as the ‘split of identity’). We can also define *local frequency* as the rate of change of local phase. Figure 3 shows the local phase and local amplitude measures of a modulated sinusoidal signal. It is easy to see from this example why the analytic signal is useful for signal processing operations such as demodulation and envelope detection.

2.2 Scale in Local Phase Analysis

Our example in Figure 3 demonstrated a very well-behaved analytic signal where the interpretation of the local phase and local amplitude was very clear. However, for more general signals this is not always the case. Consider instead the example in Figure 2, which is composed of the sum of three sinusoidal oscillations at different frequencies. It is not clear in this case what constitutes the amplitude envelope, and what constitutes the oscillations within this envelope. Indeed we find that the local amplitude and local phase measures do not help us much in this regard (see Figure 4).

The problem is that structure exists in this signal at different scales – to get a useful output we need to choose what scale of structure we are interested in. We can do this by filtering the signal with a bandpass filter centred on the frequencies of interest before calculating the analytic signal. There are a number of different filter types that may be used for this purpose, for example Gabor, log-Gabor, Poisson, Cauchy, Gaussian derivative and others. We will focus on log-Gabor filters here as they are a popular choice.

A log-Gabor filter [2, 3] (sometimes referred to as a log-normal filter) is defined in the frequency domain as having a Gaussian frequency response when viewed on a logarithmic frequency axis:

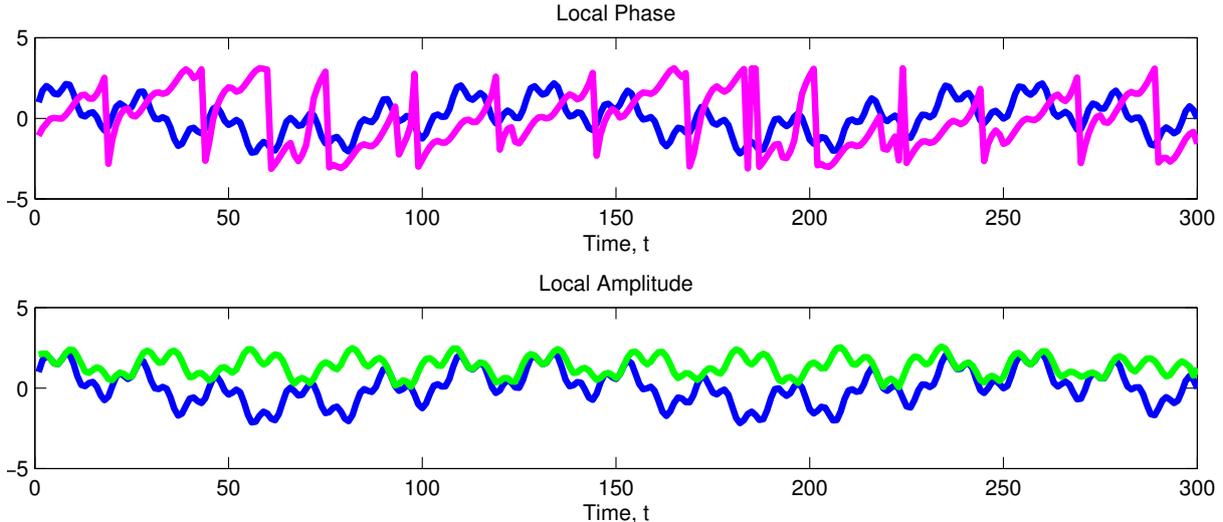


Figure 4: The local phase (*top, magenta*) and local amplitude (*bottom, green*) measures of the signal shown in Figure 2 (*blue*).

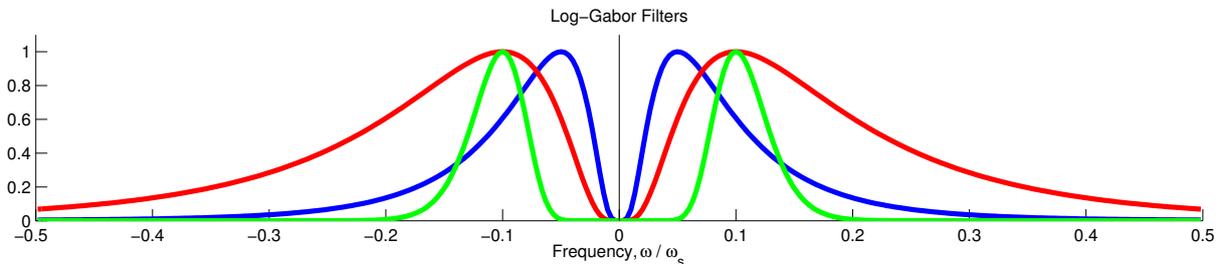


Figure 5: Frequency domain representations of log-Gabor filters. *Red*: $\omega_0 = 0.2\omega_s$, $\sigma_0 = 0.5$; *Blue*: $\omega_0 = 0.1\omega_s$, $\sigma_0 = 0.5$; *Green*: $\omega_0 = 0.2\omega_s$, $\sigma_0 = 0.8$ where ω_s is the sampling frequency

$$G(\omega) = \exp\left(-\frac{\left(\log\left(\frac{|\omega|}{\omega_0}\right)\right)^2}{2(\log(\sigma_0))^2}\right). \quad (18)$$

Log-Gabor filters are usually defined in the frequency domain since there is no closed-form expression for their time-domain form (i.e. impulse response). There are two degrees of freedom when designing a log-Gabor filter. The first, ω_0 , is the centre-frequency of the passband. It governs what scale structures are picked out by the filter. Structures with wavelengths approximately equal to that of the filter, $\lambda_0 = 2\pi/\omega_0$, will be selected. The second, σ_0 , is a shape parameter that governs how selective the filter is in the frequency domain (i.e. it governs the bandwidth of the passband). See Figure 5 for some examples of log-Gabor filters. One of the key advantages of log-Gabor filters is that they can be constructed with arbitrary bandwidth (by varying σ_0) whilst maintaining a gain of zero at a $\omega = 0$ (the DC frequency); this can clearly be seen in the figure. Further discussion of the different filter types is out of the scope of this document, but see Boukerroui et al. [4] for a detailed comparison.

In order to apply a log-Gabor filter (or other filter) to the problem of scale-selectivity with the analytic signal, we can alter Equation (7) to include a filtering stage before finding the analytic signal by multiplying by our frequency-domain filter, $G(\omega)$:

$$F_a(\omega) = (1 + \text{sgn}(\omega)) G(\omega) F(\omega). \quad (19)$$

The ‘scaleogram’ in Figure 6 shows the scale selection behaviour of the local phase measure. At the smaller scales (< 10 pixels), the local phase measure describes the very small scale oscillations, and then

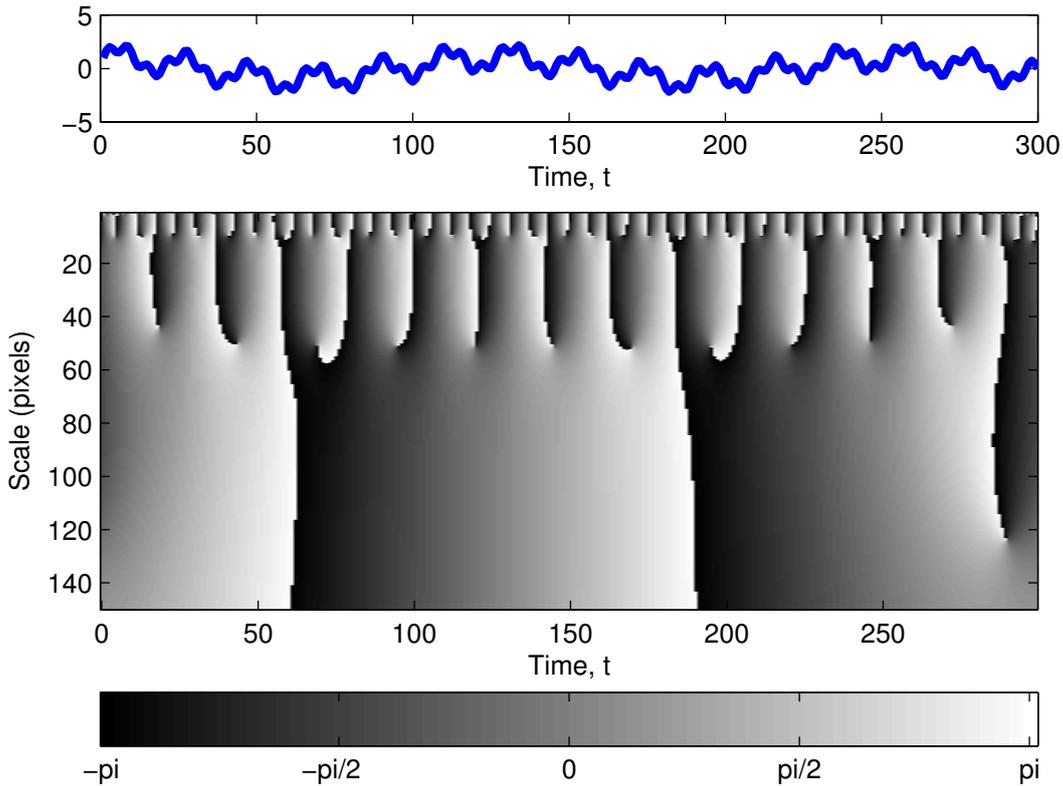


Figure 6: Scaleogram of local phase for the signal in Figure 2 (shown above). The image represents local phase of the signal through time along the horizontal axis and over different scales on the vertical axis. The grey value of each pixel represents the local phase value according to the scale below the image. At each different scale (horizontal line on the image) the local phase was calculated after applying a log-Gabor filter at the relevant scale (i.e. wavelength λ_0) and shape parameter $\sigma_0 = 0.6$. The different interpretations of the signal structure at different scales are evident.

becomes dominated by the medium scale changes as the centre-wavelength of the filter increases, before finally becoming dominated by the long term trends at scales of about 60 pixels and greater.

It is important to realise how this local phase and amplitude model differs from the Fourier transform. The Fourier transform expresses the *entire signal* as the sum of complex exponentials with different amplitudes and phases. It therefore provides localisation in frequency but has infinite spatial extent. By contrast, the local phase model that we have discussed is localised in space, as the phase and amplitude estimates are local, but only covers a range of frequencies due to the bandpass filtering. Therefore, the model can be seen as a compromise between localisation in space and in frequency. For further discussion of this idea, see the famous paper by Field [2].

2.3 Odd and Even Quadrature Filters

Take another look at Equation (19). We described this process as first filtering the signal (i.e. finding $G(\omega)F(\omega)$) and then finding the analytic signal of this filtered signal using multiplication by $(1 + \text{sgn}(\omega))$. However, there is another equivalent and equally valid interpretation of Equation (19): first find the analytic representation of *the filter* by multiplying its spectrum in the usual way:

$$G_a(\omega) = G(\omega) (1 + \text{sgn}(\omega)), \quad (20)$$

and now use this new ‘analytic’ filter to filter the original signal. If we choose to think along these lines,

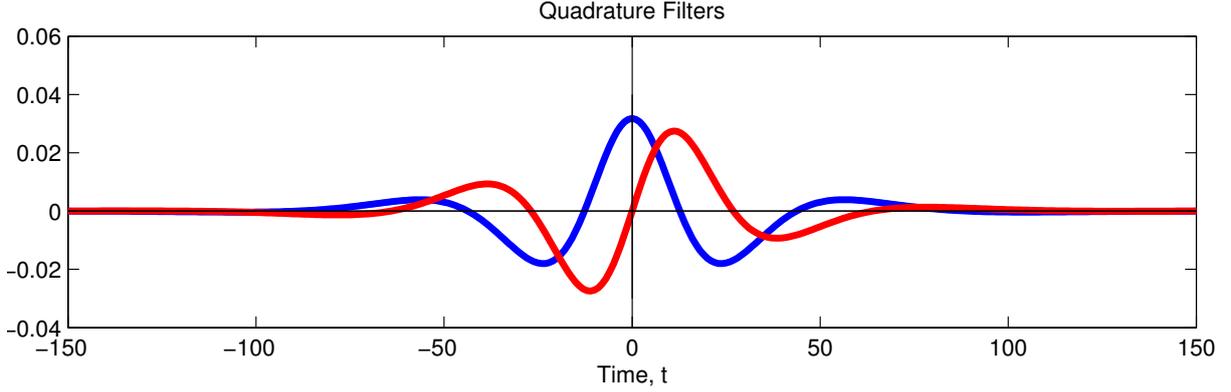


Figure 7: Time-domain impulse responses of a log-Gabor quadrature pair: *blue* real, even-symmetric part, *red* imaginary, odd-symmetric part. These have been calculated from the frequency-domain representation (Equation (18)) using an inverse FFT, as there is no closed-form expression.

what can we say about our ‘analytic’ filter? Just like any analytic signal, it is complex-valued with the real part being the original filter impulse response, and the imaginary part being in quadrature with it. This means that if we use the filter on a real signal, the result is a complex-valued signal where the real part is the signal filtered with the real part of the filter, and the imaginary part is the signal filtered with the imaginary part of the filter.

Additionally, bandpass filters, such as the log-Gabor filter, have symmetric impulse responses and therefore real-valued symmetric spectra. Consequently, the Hilbert transform (imaginary part) of the filter has a real-valued, odd symmetric spectrum and an odd symmetric filter response. This is also evident from the phase-shift argument – an even-symmetric function may be written as a sum of cosines and the Hilbert transform induces a phase shift of $-\pi/2$, meaning that the resulting signal may be written as a sum of pure sines and must therefore be odd-symmetric. See Figure 7 for an illustration of a log-Gabor quadrature filter.

It is therefore common to think of the process of finding the analytic signal as convolving the original signal with two filters: one even-symmetric (or ‘even’ filter) and one odd-symmetric (or ‘odd’ filter). However it is important to note that, as we have just discussed, this is mathematically equivalent to finding the analytic representation of the signal after filtering by the even filter alone.

2.4 Relation to Gradient

We have said that the Hilbert transform of a signal can be thought of in terms of phase-shifting the frequency components in the signal. Are there any other interpretations? Look again at our definition of the Hilbert transform in the frequency domain, Equations (4) and (5). We can rewrite these in yet another equivalent form (this time we multiply by $-i$ to give a real time-domain signal):

$$F_h(\omega) = \begin{cases} -i \frac{\omega}{|\omega|} \cdot F(\omega), & \omega \neq 0 \\ 0, & \omega = 0. \end{cases} \quad (21)$$

Compare this to the Fourier domain form of the derivative operation on a signal:

$$F'(\omega) = i\omega F(\omega), \quad (22)$$

where $F(\omega)$ is the Fourier transform of $f(t)$ and $F'(\omega)$ is the Fourier transform of $f'(t) = \frac{d}{dt}f(t)$. We can see that the Hilbert transform of a signal can be thought of as a (negative) smoothed gradient, where the smoothing kernel is $|\omega|^{-1}$ in the frequency domain. In fact, the relationship between the gradient and the Hilbert transform is deeper than this and has its roots in fractional calculus. However for the purposes of this document we will simply note that it can be helpful to think of the Hilbert transform as a smoothed gradient and refer the reader to Unser et al. [5] for more details.

3 2D Images and the Monogenic Signal

Given the usefulness of the analytic signal in signal processing and interpretation, it is natural to try to extend the approach to two dimensions. Indeed the importance of phase in images has long been established [6]. Ideally we would like a way to extract a phase quadrature image from an original image, such that we can extract local phase and amplitude information. Unfortunately, this process is not straightforward in two dimensions because we now need to consider different directions. Some authors have tried to generate representations that consider the Hilbert transform along a preferred or pre-selected direction in the image. However, we have only generated a truly useful representation if it is *isotropic*, i.e. treats all directions in the image in the same way. The *monogenic signal* of Felsberg and Sommer [7, 8] is the only isotropic generalisation of the analytic signal, and has found applications in a number of different image processing tasks. It is this approach that we will consider from now on (see [8] for a discussion of other approaches).

Recall that to create the analytic signal we generated a single imaginary, phase-quadrature part via the Hilbert transform and used it in combination with the (real) original signal to give the analytic signal. Just like we need two components to express the gradient in an image, it turns out that in the 2D case we need *two* imaginary parts – one for each axis direction – and therefore we *cannot* represent the monogenic signal using a complex number. It is instead necessary to turn a branch of mathematics called *geometric algebra*, and in particular to the algebra of *quaternions*, which in simple terms can be thought of as generalised complex numbers with one real part and three imaginary parts. This makes the derivation and some of the principles of the monogenic signal rather inaccessible to those without the relevant mathematical background, and unfortunately geometric algebra is not well-known within the image processing community. This does not mean, however, that we cannot understand and use the monogenic signal. Imagine if we had presented §2 with no understanding of complex numbers: we could simply have treated the signal and its Hilbert transform as two different signals and neglected the fact that they in fact form the real and imaginary parts of a single complex signal. This is the approach we will take here for the monogenic signal: we shall (at least initially) present the monogenic signal as having three real-valued parts. However we should bear in mind that in fact we are really dealing with different parts of the same signal within the quaternion framework.

For obvious reasons, we will focus our discussion on the monogenic signal of 2D images, however everything here generalises trivially to higher dimensions by adding further odd parts (one for each dimension). Indeed previous work has shown the 3D monogenic signal to be useful in analysing 3D volumetric ultrasound data [9].

To generate the two odd parts from the original signal, we need to use a generalisation of the Hilbert transform called the *Riesz transform*. Let $f(\mathbf{x})$ be a 2D image of a spatial variable $\mathbf{x} = (x, y)^T$, and let $F(\boldsymbol{\omega})$ be its frequency-domain representation found using the 2D Fourier transform, where $\boldsymbol{\omega} = (\omega_x, \omega_y)^T$ is a two-dimensional frequency. The two odd parts of the monogenic signal, $F_{o1}(\boldsymbol{\omega})$ and $F_{o2}(\boldsymbol{\omega})$, are found in the frequency domain as follows:

$$F_{o1}(\boldsymbol{\omega}) = \begin{cases} i \frac{\omega_x}{\|\boldsymbol{\omega}\|} F(\boldsymbol{\omega}), & \boldsymbol{\omega} \neq 0 \\ 0, & \boldsymbol{\omega} = 0, \end{cases} \quad (23)$$

$$F_{o2}(\boldsymbol{\omega}) = \begin{cases} i \frac{\omega_y}{\|\boldsymbol{\omega}\|} F(\boldsymbol{\omega}), & \boldsymbol{\omega} \neq 0 \\ 0, & \boldsymbol{\omega} = 0. \end{cases} \quad (24)$$

We can show using symmetry arguments² that, if $F_{o1}(\boldsymbol{\omega})$ and $F_{o2}(\boldsymbol{\omega})$ are defined in this way, the corresponding image domain representations, $f_{o1}(\mathbf{x})$ and $f_{o2}(\mathbf{x})$, are purely real.

Comparing this definition to Equation (21), it should be clear how this is a generalisation of the Hilbert transform (note that one can also include a minus sign in this definition, but it makes no difference as long as there is consistency). Each of the frequency components in each of the resulting odd parts is

²In 2D a real-valued image has a spectrum with an even-symmetric real part ($\text{Re}\{F(\omega_x, \omega_y)\} = \text{Re}\{F(-\omega_x, -\omega_y)\}$ and $\text{Re}\{F(-\omega_x, \omega_y)\} = \text{Re}\{F(\omega_x, -\omega_y)\}$) and an odd-symmetric imaginary part ($\text{Im}\{F(\omega_x, \omega_y)\} = -\text{Im}\{F(-\omega_x, -\omega_y)\}$ and $\text{Im}\{F(-\omega_x, \omega_y)\} = -\text{Im}\{F(\omega_x, -\omega_y)\}$). Since in Equations (23) and (24), $F(\boldsymbol{\omega})$ is the transform of a real image, it obeys the symmetry conditions. Following this through, $F_{o1}(\boldsymbol{\omega})$ and $F_{o2}(\boldsymbol{\omega})$ also obey the symmetry conditions. Therefore the image-domain representations of the odd parts are purely real.

phase-shifted by $-\pi/2$, and also scaled such that the original amplitude is split between the two odd components.

It is also possible to express the Riesz transform as a convolution in the image domain, though we will not discuss it because the frequency domain representation is conceptually and computationally simpler (see [10, 5] for the relevant equations).

3.1 Spherical Quadrature Filters

Just like in the 1D case, it is usually advantageous to filter the image with a 2D bandpass filter to introduce scale-selectivity before finding the monogenic signal. Just as in the 1D case, we can also think of this process as first finding the monogenic representation of the filters and then filtering the image with these filters.

In the 2D case, we start with a bandpass filter that is radially symmetric about the origin ('even') in both the frequency domain and image domain. We will again use the log-Gabor filter as an example, and note that a number of other filters are in common usage too (see [4]). The (even) 2D log-Gabor filter is given by:

$$G_e(\boldsymbol{\omega}) = \exp\left(-\frac{\left(\log\left(\frac{\|\boldsymbol{\omega}\|}{\omega_0}\right)\right)^2}{2(\log(\sigma_0))^2}\right). \quad (25)$$

We then form two odd parts of the filter, $G_{o1}(\boldsymbol{\omega})$ and $G_{o2}(\boldsymbol{\omega})$, using the Riesz transform, as in Equations (23) and (24). Each of the two resulting filters are odd-symmetric, with the axis of symmetry along the two image axes, see Figures 8 and 9. The resulting odd filter is isotropic because the combined amplitude, $\sqrt{G_{o1}(\boldsymbol{\omega})^2 + G_{o2}(\boldsymbol{\omega})^2} = G(\boldsymbol{\omega})$, is isotropic.

After filtering, we can write the monogenic signal as a combination of the three parts (one even, two odd) as a vector:

$$\mathbf{f}_m(\mathbf{x}) = \begin{bmatrix} f_e(\mathbf{x}) \\ f_{o1}(\mathbf{x}) \\ f_{o2}(\mathbf{x}) \end{bmatrix}. \quad (26)$$

See Figure 10 for an example of the different components of the monogenic signal calculated at different wavelengths $\lambda_0 = 2\pi/\omega_0$. Notice how the two parts of the odd filter pick out horizontal and vertical changes in the image (recall the analogy with the smoothed gradient, which is very useful in 2D), and how structures of different sizes are picked out when filters of different wavelengths are used. Realising that the two parts of the odd filter are in fact part of one filter leads to more intuitive visualisations such as those in Figure 11, which clearly demonstrate the odd filter picking out changes in the image and their directions.

It is also common to see the monogenic signal presented as the combination of one even part and *one* odd part. Where this slightly misleading terminology occurs, the single odd part has been formed from the combined magnitude of the two odd parts. We will write the single odd part as $f_o(\mathbf{x})$ where:

$$f_o(\mathbf{x}) = \sqrt{f_{o1}(\mathbf{x})^2 + f_{o2}(\mathbf{x})^2}. \quad (27)$$

Note that the two odd parts $f_{o1}(\mathbf{x})$ and $f_{o2}(\mathbf{x})$ must be calculated first and then combined to form the single part in the image domain (the magnitude cannot be found directly using a single filter). In some cases this notation is convenient, but beware that the orientation information is lost by performing this calculation.

3.2 Local Phase, Amplitude, and Orientation in Images

Equipped with the monogenic signal for 2D images, we are now able to extend the concepts of local phase and local amplitude to two dimensions. Recall how for 1D signals, we wrote the two parts of the analytic signal in a polar coordinate system and this gave us the amplitude and phase. In two dimensions, we have three components so we write them in a standard *spherical polar* coordinate system, using the radius, elevation angle, and azimuthal angle.

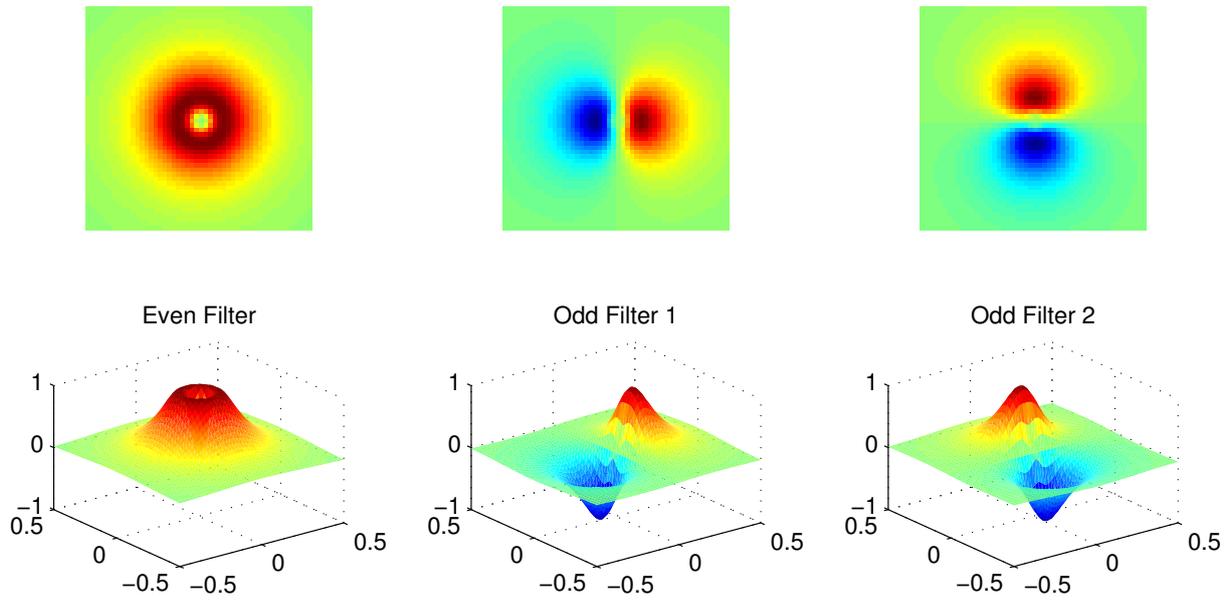


Figure 8: Frequency-domain representations of a set of log-Gabor spherical quadrature filters ($\sigma_0 = 0.5$).

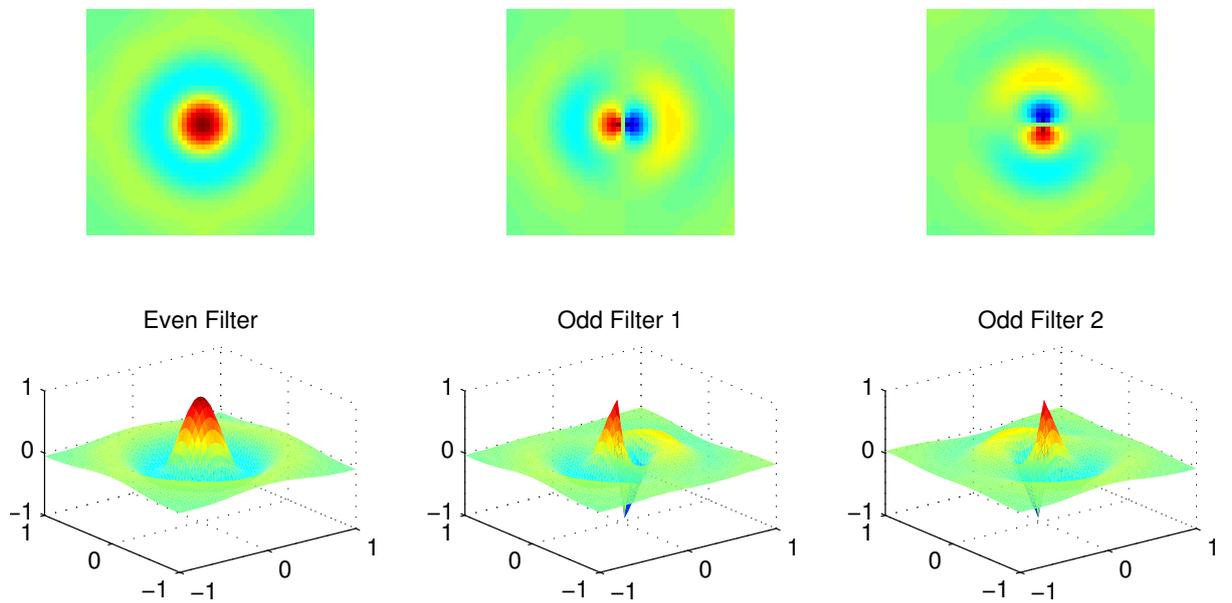


Figure 9: Image-domain representations (impulse responses) of a set of log-Gabor spherical quadrature filters ($\sigma_0 = 0.5$). Like the 1D case, there is no closed-form representation in the image domain, so these have been calculated numerically using the inverse FFT.

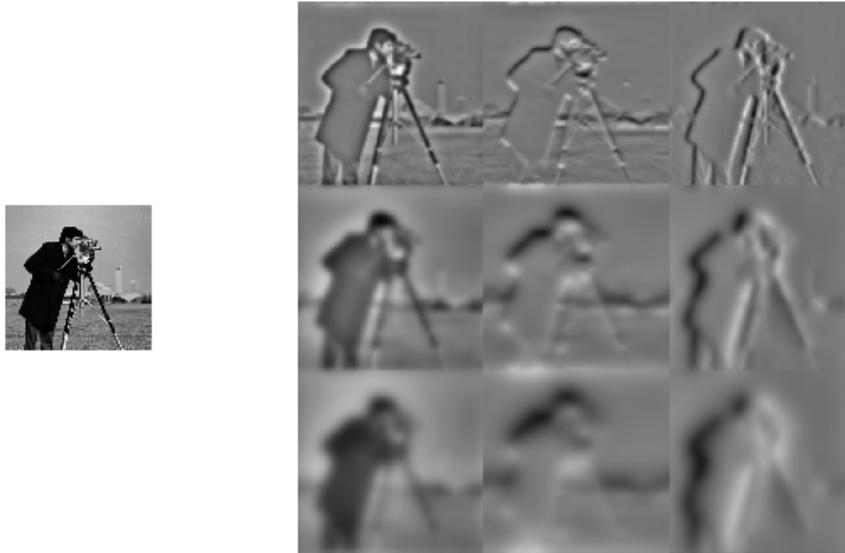


Figure 10: Cameraman test image (*left*) and its monogenic signal (*right*). In the right hand image: *left column* $f_e(\mathbf{x})$, *centre column* $f_{o1}(\mathbf{x})$, *right column* $f_{o2}(\mathbf{x})$, *top row* $\lambda_0 = 20$ pixels, *centre row* $\lambda_0 = 60$ pixels, *bottom row* $\lambda_0 = 100$ pixels. Original image size 256×256 pixels.

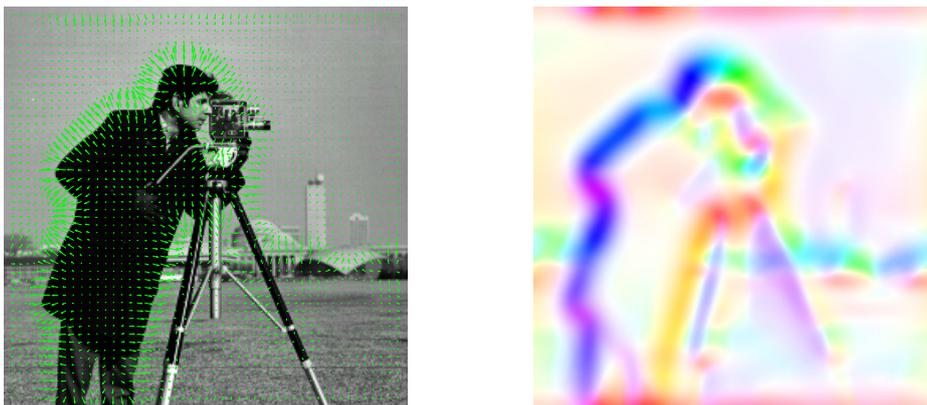


Figure 11: Alternative ways of visualising the odd filter response in the middle row of Figure 10: *left* as a vector field superimposed over the original image, *right* as a colour image with hue representing orientation and saturation representing magnitude.

As before, the local amplitude is the radial part of the representation (see Figure 12):

$$A(\mathbf{x}) = \sqrt{f_e(\mathbf{x})^2 + f_{o1}(\mathbf{x})^2 + f_{o2}(\mathbf{x})^2} \quad (28)$$

$$= \sqrt{f_e(\mathbf{x})^2 + f_o(\mathbf{x})^2}. \quad (29)$$

Additionally, some works refer to *local energy*, which is simply the square of the local amplitude.

The local phase is found from the angle between the even part and the *combined* odd part, and hence forms the elevation angle of the spherical polar representation:

$$\phi(\mathbf{x}) = \arctan\left(\frac{f_o(\mathbf{x})}{f_e(\mathbf{x})}\right). \quad (30)$$

To complete the spherical polar representation we also need the *local orientation*, θ , which is simply the orientation of the odd filter response and represents the dominant direction in the image at that point (the orientation of the arrows in Figure 11):

$$\theta(\mathbf{x}) = \arctan\left(\frac{f_{o2}(\mathbf{x})}{f_{o1}(\mathbf{x})}\right). \quad (31)$$

The local orientation thus forms the azimuthal part of the spherical polar representation. The estimation of the local orientation can be unstable if the phase, ϕ is close to 0, and so there some exist methods to estimate it more robustly (e.g. [10, 5]). The isotropy of the monogenic filter means that, ignoring imperfection due to sampling, the local amplitude and local phase are unchanged by rotations of the input image while the local orientation rotates along with the input image.

We can interpret the image in the local area around a point as being approximated by a cosine wave with amplitude A and phase ϕ , and orientated on the image in direction θ [11]. If we define a unit vector in the local orientation $\mathbf{n}(\mathbf{x}) = [\cos(\theta(\mathbf{x})), \sin(\theta(\mathbf{x}))]^T$ then around a point $\bar{\mathbf{x}}$ we have:

$$f(\mathbf{x}') \approx A(\bar{\mathbf{x}}) \cos(\mathbf{x}' \cdot \mathbf{n}(\bar{\mathbf{x}}) + \phi(\bar{\mathbf{x}})), \quad (32)$$

where $\mathbf{x}' = \mathbf{x} - \bar{\mathbf{x}}$. Notice that we are only modelling intensity variation in the direction of the local orientation. This means that the monogenic signal is useful for modelling image features such as edges and lines that have variation in one direction only (so called *intrinsically 1D* or *1iD* signals [11]), but cannot model image features such as corners that have variation in two directions (*intrinsically 2D* or *2iD* signals). This is one important limitation of the monogenic signal.

Note also that there is a potential ambiguity here when defining the orientation and phase: for any given orientation and phase pair, we could always change the orientation to point in the opposite direction and change the sign of the phase to get another pair of values that are equally valid under the model in Equation (32). To constrain this, in the 2D case the local phase can therefore only take values between 0 and π (this is a natural consequence of the fact that the combined odd filter $f_o(\mathbf{x})$ is always positive due to the definition in Equation (27)). Another way of looking at this is that the local orientation is defined to always point ‘uphill’ towards areas of greater intensity, and the phase value is constrained to half the full range.

The amplitude, phase and orientation images for the *cameraman* image are shown in Figure 13.

3.3 Complex Odd Filter Representation

Whilst it is most accurate to represent the monogenic signal using quaternions (which is often undesirable in practical implementations), there are some advantages of representing just the odd part of the filter using a complex number. If we define a *complex odd filter* in the frequency domain [5] as:

$$G_{oc}(\boldsymbol{\omega}) = G_{o1}(\boldsymbol{\omega}) + iG_{o2}(\boldsymbol{\omega}) \quad (33)$$

$$= i \frac{\omega_x}{\|\boldsymbol{\omega}\|} G_e(\boldsymbol{\omega}) + i \left(i \frac{\omega_y}{\|\boldsymbol{\omega}\|} G_e(\boldsymbol{\omega}) \right) \quad (34)$$

$$= \frac{i\omega_x - \omega_y}{\|\boldsymbol{\omega}\|} G_e(\boldsymbol{\omega}). \quad (35)$$

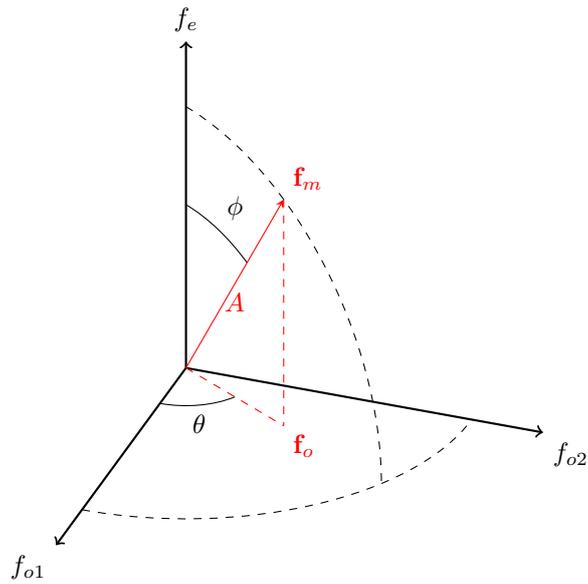


Figure 12: Spherical polar representation representation of the components of the monogenic signal. The red arrow shows the monogenic signal at one point in an image. The angles ϕ , representing the *local phase*, and θ , representing the *local orientation*, are shown.

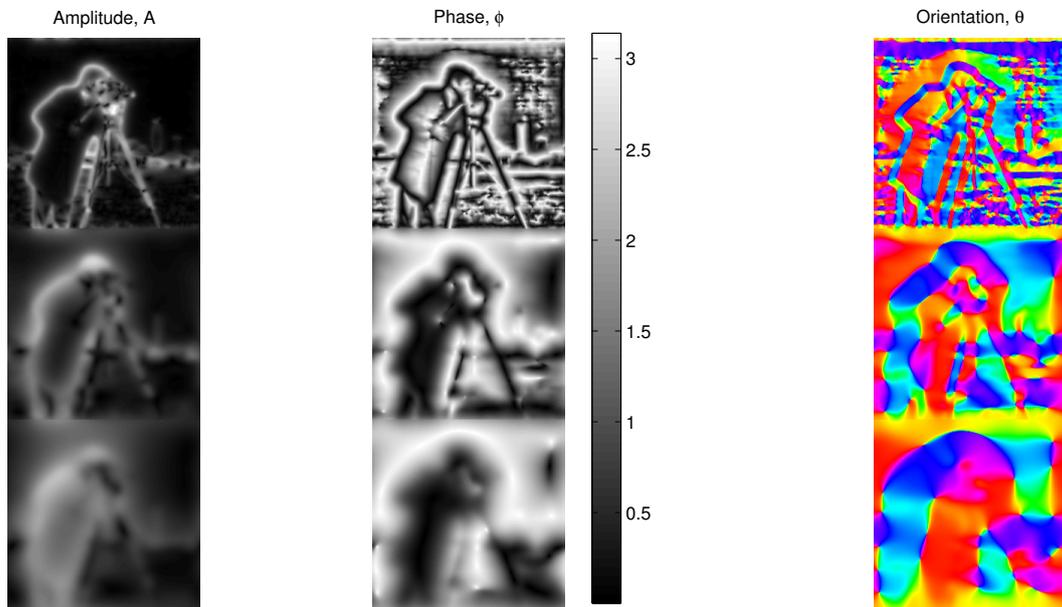


Figure 13: Local amplitude, phase and orientation images derived from the monogenic signals in Figure 10. The orientation image encodes angle as a hue on a colour wheel.

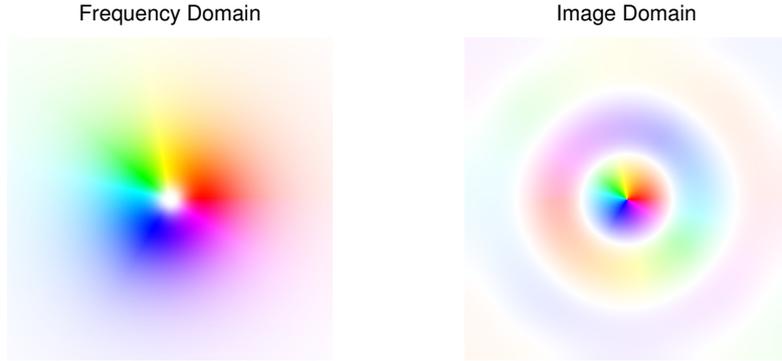


Figure 14: Representations of complex valued filters: (*left*) the odd filters from Figure 8 shown in the frequency domain, (*right*) the odd filters from Figure 9 shown in the image domain. In each case, the hue represents the argument of the complex value, and the saturation represents the magnitude.

By linearity of the Fourier transform, in the image domain this gives us an impulse response of

$$g_{oc}(\mathbf{x}) = g_{o1}(\mathbf{x}) + ig_{o2}(\mathbf{x}), \quad (36)$$

i.e. a complex-valued filter whose real and imaginary parts are the two parts of the odd filter. Convolution with this complex filter with an image then gives us an image $f_{oc}(\mathbf{x})$ whose real and imaginary parts are the two odd parts of the image's monogenic signal. This is convenient because the local orientation is intuitively recovered as the argument of the complex number³, and the amplitude of the complex number is the magnitude of the combined odd filter:

$$\theta(\mathbf{x}) = \arg(f_{oc}(\mathbf{x})) \quad (37)$$

$$f_o(\mathbf{x}) = |f_{oc}(\mathbf{x})| \quad (38)$$

$$A(\mathbf{x}) = \sqrt{f_e(\mathbf{x})^2 + |f_{oc}(\mathbf{x})|^2} \quad (39)$$

$$\phi(\mathbf{x}) = \arctan\left(\frac{|f_{oc}(\mathbf{x})|}{f_e(\mathbf{x})}\right). \quad (40)$$

Using a complex odd filter also has computational advantages. Since performing an inverse fast Fourier transform is inherently an operation involving complex numbers, it is generally no more computationally expensive to perform the inverse transform $f_{oc}(\mathbf{x}) = \mathcal{F}^{-1}\{F_{oc}(\boldsymbol{\omega})\}$ than it is to perform either of the filters individually $f_{o1}(\mathbf{x}) = \mathcal{F}^{-1}\{F_{o1}(\boldsymbol{\omega})\}$ or $f_{o2}(\mathbf{x}) = \mathcal{F}^{-1}\{F_{o2}(\boldsymbol{\omega})\}$. Therefore, by implementing a single complex filter rather than two separate filters, we save ourselves one FFT operation. This is particularly important where processing time is important, for example in video applications.

Figure 14 shows the complex-valued alternatives to the odd filter pairs shown in previous examples. From this representation it is easier to see that the combined effect of the odd filters is isotropic. Because the argument of the complex number is the local orientation, if we rotate the input image by an angle, ψ , the resulting complex odd filter response, $f_{oc}(\mathbf{x})$, is simply rotated by multiplication by the unit complex number $e^{i\psi}$.

4 Quantities Derived from the Monogenic Signal

We have now introduced the monogenic signal for image analysis. This section will introduce some important quantities that may be derived from the monogenic signal. Beware that the quantities discussed here pre-date the development of the monogenic signal, so many earlier texts used sets of quadrature filters at a range of orientations to calculate the local amplitude, phase, and orientation measures.

³This puts the complex odd filter neatly within the framework of rotationally *equivariant* filters that are used for rotation-invariant description in several works, e.g. [12]

4.1 Phase Symmetry and Asymmetry

It is often useful to measure the extent to which the structure around a point in an image is locally *symmetric* or *asymmetric*. For a signal to be symmetric, all the constituent frequency components must be symmetric, i.e. all frequencies must have a phase of 0 or π or equivalently be pure cosine waves. Conversely all components of an asymmetric signal must have phases of $\pi/2$ or $3\pi/2$ or equivalently be pure sine waves.

One way to quantify symmetry therefore is to find the difference between the sine and cosine of the local phase. Kovessi’s feature symmetry S and asymmetry R measures [13] are based on this principle:

$$S(\mathbf{x}) = \frac{\lfloor |\cos(\phi(\mathbf{x}))| - |\sin(\phi(\mathbf{x}))| - T \rfloor}{A(\mathbf{x}) + \epsilon} \quad (41)$$

$$R(\mathbf{x}) = \frac{\lfloor |\sin(\phi(\mathbf{x}))| - |\cos(\phi(\mathbf{x}))| - T \rfloor}{A(\mathbf{x}) + \epsilon}, \quad (42)$$

where T is a threshold that sets the sensitivity of the measure, $\lfloor \cdot \rfloor$ is an unconventional shorthand meaning that values less than zero are replaced with zero, and ϵ is a small number that avoids division by zero. Normalising by the local amplitude gives a measure of symmetry that is independent of amplitude (i.e. a purely structural measure) and lies in the range $0 \leq S(\mathbf{x}) < 1$ and $0 \leq R(\mathbf{x}) < 1$. A large number of pixels in the image have symmetry/asymmetry values of 0 due to the thresholding.

The definitions of symmetry and asymmetry in equations (41) and (42) above are equivalent to the following, written in terms of parts of the monogenic signal, using the definition of the combined (directionless) odd filter in Equation (27):

$$S(\mathbf{x}) = \frac{\lfloor |f_e(\mathbf{x})| - |f_o(\mathbf{x})| - T \rfloor}{A(\mathbf{x}) + \epsilon} \quad (43)$$

$$R(\mathbf{x}) = \frac{\lfloor |f_o(\mathbf{x})| - |f_e(\mathbf{x})| - T \rfloor}{A(\mathbf{x}) + \epsilon}. \quad (44)$$

This is intuitive because the magnitude of the response of a near-symmetric function to the even-symmetric filter $g_e(\mathbf{x})$ will be large and the magnitude of the response to the odd-symmetric filter $g_o(\mathbf{x})$ will be small. The opposite holds for functions that are nearly asymmetric.

Equations (43) and (44) measure scale at just one scale in the image (the scale of the bandpass filter used to calculate the monogenic signal). Depending on the application, it is often more useful to accumulate results at a number of scales. If we denote properties calculated at a wavelength λ using a subscript i then a multiscale measure of symmetry is given by

$$S(\mathbf{x}) = \sum_i \frac{\lfloor |f_{e,\lambda_i}(\mathbf{x})| - |f_{o,\lambda_i}(\mathbf{x})| - T \rfloor}{A_{\lambda_i}(\mathbf{x}) + \epsilon} \quad (45)$$

$$R(\mathbf{x}) = \sum_i \frac{\lfloor |f_{o,\lambda_i}(\mathbf{x})| - |f_{e,\lambda_i}(\mathbf{x})| - T \rfloor}{A_{\lambda_i}(\mathbf{x}) + \epsilon}, \quad (46)$$

where $\{\lambda_i\}$ are a set of filter centre-wavelengths chosen to suit the application.

Note that there are other ways of obtaining a phase-based description of images, such as using a set of filters with different orientations or a steerable filters, and these have frequently been used in the past instead of the monogenic signal [13].

We return to the cameraman test image for an illustration of feature symmetry and asymmetry (see Figure 15). We can see that feature symmetry picks out ‘blobs’ (regions of similar appearance) at the scale of interest whereas feature asymmetry picks edge-like features that form the boundaries between them. Figure 16 shows an example of the multi-scale measure on the same image. Notice how multi-scale feature asymmetry is particularly helpful for detecting useful boundaries.

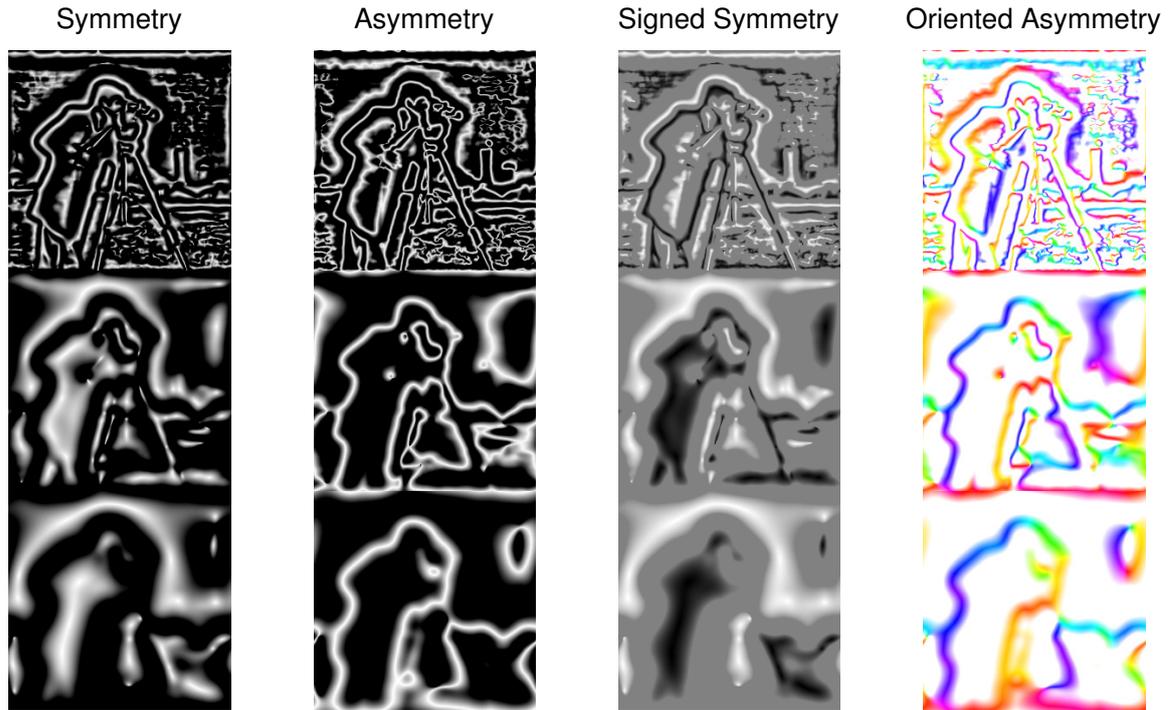


Figure 15: *First column* symmetry and *second column* asymmetry calculated for the cameraman test image using the same filter parameters as the corresponding rows in Figure 10 and a threshold $T = 0.18$. Values are displayed from 0 (black) to 1 (white). *Third column* signed symmetry and *fourth column* oriented asymmetry calculated for the same images and parameters. Signed symmetry is shown on a scale of -1 (black) to 1 (white). Oriented asymmetry is shown as a colour image with hue representing orientation and saturation representing magnitude.

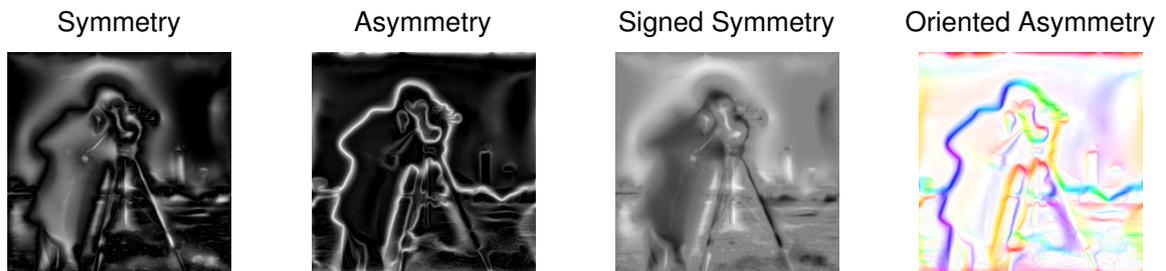


Figure 16: Multiscale symmetry and asymmetry for the cameraman test image using a set of log-Gabor filters with centre-wavelengths in the range 5 to 100 pixels and with a spacing of 5 pixels (i.e. $\{5, 10, 15, \dots, 100\}$) and a shape parameter of $\sigma_0 = 0.3$. The threshold used was a threshold $T = 0.18$. Image interpretation is as in Figure 15.

4.2 Retaining Orientation in Symmetry and Asymmetry

The potential usefulness of feature symmetry and asymmetry for detecting edges and structures of interest should be apparent from Figures 15 and 16. Notice, however, that when we found a value for symmetry and asymmetry we always took the magnitude of the monogenic signal parts (Equations (43) and (44)). The result is that we throw away some information that may be useful to us, depending on the application.

In the case of feature symmetry, using the magnitude of the monogenic even part means that we are discarding information about whether the pixel sits in a *peak* or a *trough*. This can be rectified by retaining the sign of the even part in Equation (43) by simply multiplying by the signum function:

$$\hat{S}(\mathbf{x}) = \frac{\lfloor |f_e(\mathbf{x})| - |f_o(\mathbf{x})| - T \rfloor}{A(\mathbf{x}) + \epsilon} \cdot \text{sgn}(f_e(\mathbf{x})) \quad (47)$$

$$= S(\mathbf{x}) \text{sgn}(f_e(\mathbf{x})) \quad (48)$$

$$= S(\mathbf{x}) \frac{f_e(\mathbf{x})}{|f_e(\mathbf{x})|}, \quad f_e(\mathbf{x}) \neq 0. \quad (49)$$

This gives a *signed feature symmetry* value in the range $-1 \leq \hat{S}(\mathbf{x}) < 1$. To my knowledge, this has not been explicitly described before except by my recent work [14]. However, some authors may have used a similar technique in earlier work, for example inspection of figures in [15] suggests that this technique may have been used.

In the case of feature asymmetry, by taking the magnitude of the monogenic odd part in Equation (44), we are throwing away the orientation of the edge. Edge orientation is particularly useful in many contexts, so this seems like something of an oversight. We can rectify this in a similar way, by multiplying by a unit complex number in the direction of the odd part found from the complex formulation of the odd filter. This gives a complex number $\hat{R}(\mathbf{x})$ that lies within the unit disc (i.e. $0 \leq |\hat{R}(\mathbf{x})| < 1$) quantifying both the magnitude and direction of the asymmetry (one could equally use a vector formulation, but a complex formulation is convenient when working with a complex odd filter definition):

$$\hat{R}(\mathbf{x}) = \frac{\lfloor |f_o(\mathbf{x})| - |f_e(\mathbf{x})| - T \rfloor}{A(\mathbf{x}) + \epsilon} \cdot \frac{f_{oc}(\mathbf{x})}{|f_{oc}(\mathbf{x})|} \quad (50)$$

$$= R(\mathbf{x}) \frac{f_{oc}(\mathbf{x})}{|f_{oc}(\mathbf{x})|}. \quad (51)$$

Figures 15 and 16 compare signed feature symmetry with feature, and oriented feature asymmetry with feature asymmetry for the cameraman test image.

4.3 Phase Congruency

Phase congruency is a further, related measure for feature detection in images. Where feature symmetry and asymmetry measure how close the local phase value is to 0 or π (symmetry) and $\pi/2$ (asymmetry), phase congruency measures how similar phase values at different scales are to *each other*. The intuition here is that interesting features appear where the phase values at different scales are the same. We will not discuss phase congruency in much detail as it has not been used much in ultrasound analysis. However it is widely used to detect edges and features in natural images. We refer the interested reader to Kovser's work [16] for more details.

5 Application to Ultrasound Image Analysis

In this section we shall briefly summarise the utility of the monogenic signal and related measures in analysis of ultrasound images and refer the reader to the literature for more details on the specific techniques.

The monogenic signal, feature symmetry/asymmetry, and phase congruency have proven useful in ultrasound analysis tasks where more traditional techniques have failed. There are two major important reasons for this:

- By tuning the frequency of the bandpass filters, it is possible to pick out features at the desired scale and ignore small scale speckle and large scale variations such as shadowing artefacts if desired.
- Phase-based measures are robust to the contrast variation that can occur due to different imaging parameters and other effects such as shadowing and enhancement.

It has found use in a number of applications including:

- **Boundary detection:** Traditional edge detection methods tend to perform poorly on the indistinct edges in ultrasound images, and are confounded by speckle. Feature asymmetry has proven effective in detecting boundaries in both 2D. For example, Mulet-Parada and Noble [17] used feature asymmetry to detect boundaries in echocardiograms, though they used a set of oriented real-valued filters rather than the monogenic signal. Rajpoot et al. [18] used the monogenic signal for the same purpose, and this was also extended to 3D ultrasound images [9, 19].
- **Motion estimation:** Traditional ‘optical flow’ techniques tend to be based on the assumption that the brightness of corresponding image points does not change as the image moves. Felsberg [20] instead assumed the constancy of local phase between frames to derive an alternative method. Alessandrini et al. [21] built on this method to create a motion estimation algorithm for cardiac ultrasound and cardiac MR imaging.
- **Registration:** Similar to motion estimation, Grau et al. [22] used a similarity measure based on local phase rather than intensity for the alignment of 3D echocardiography imagery and find it to be more robust than intensity-based methods.
- **Compounding:** Compounding is the process of ‘fusing’ multiple aligned images to create a combined image that has the ‘best’ parts of all the original images. This usually involves using some sort of ‘feature importance’ measure to determine which parts of the final image should be taken from which input image. Grau and Noble [23] used a measure based on phase congruency for this purpose.
- **Recognition and detection:** The monogenic signal has been employed in diverse ways towards recognition in ultrasound images. Rahmatullah et al. [15] and Patwardhan [24] used feature symmetry to find candidate ‘blobs’ for detection of the stomach and umbilical vein in ultrasound. Maraci et al. [25] used histograms of keypoints based on dense SIFT descriptors calculated on a local phase image in order to recognise different views in an ultrasound video of the fetus. Bridge and Noble [14] incorporated various monogenic signal-based image representations into a rotation invariant sliding window object detection framework in order to localise the fetal heart in ultrasound videos.

However, beware the following limitations, as the monogenic signal is not appropriate for all applications in ultrasound:

- Information is lost by using the monogenic signal due to the bandpass filtering process. This may mean, for example, that speckle information and absolute intensity information is lost. Though this is often useful to provide invariance to contrast and robustness to speckle, it is important to consider whether this is appropriate for a particular application. Particularly in the case of applications that make use of learning techniques, it is important to realise that using monogenic features is forcing the algorithm to use a certain low-level representation that may be suboptimal.
- As described in §3.2, the monogenic signal assumes that the image is locally intrinsically one dimensional. This means that it is not capable of analysing 2D features such as corners or curvature.
- In the language of computer vision, the monogenic signal and derived features could be considered ‘hand-crafted’ features, based on mathematical principles rather than being optimised for the task at hand. Over the last few years, the general computer vision community has been moving away from such hand-crafted features in favour of learning task-specific image representations with techniques such as convolutional neural networks, which have achieved the recent state-of-the-art results on detection and recognition tasks. Such learned features have yet to be thoroughly investigated in the domain of ultrasound imaging, perhaps in part because of the large amount of training examples typically needed to train them.

6 Extensions and Generalisations

Here we briefly summarise some extensions to the theory presented so far that may be of interest to those working with the monogenic signal.

- Scale Space: Felsberg and Sommer [26] developed a mathematical framework for conducting scale-space analysis using phase-based processing, based on the link between the monogenic signal and the Poisson bandpass filter.
- Wavelets: Unser et al. [5] have developed a multi-scale wavelet analysis based on the Riesz transform that is essentially a monogenic wavelet decomposition acting in the frequency domain. A fast image domain approximation has been developed by Wadhwa et al. [27] and has been used to provide amplification of imperceptible motion in video.
- 2D Analytic Signal: It is possible to extend the monogenic signal to enable it to characterise image regions that are intrinsically 2D (i.e. have curvature in the second dimension such as a corner). Wietzke and colleagues [11, 28, 29] have achieved this using additional filters found from the even filter using a *second-order* Riesz transform and call the result (perhaps confusingly) the *2D analytic signal*. This has been used on radio frequency ultrasound data in order to improve the envelope detection method [30].

7 Acknowledgements

I would like to thank Ana Namburete and Thomas Nketia for the initial idea to write this document, and Davis Vigneault and Ruobing Huang for their comments on how to improve early drafts.

References

- [1] J.O. Smith. Mathematics of discrete Fourier transform (DFT) with audio applications. https://ccrma.stanford.edu/~jos/r320/Analytic_Signals_Hilbert_Transform.html.
- [2] D.J. Field. Relations between the statistics of natural images and the response properties of cortical cells. *J. Opt. Soc. Am. A*, 4:2379–2394, 1987.
- [3] P. Kovesi. What are log-Gabor filters and why are they good? <http://www.csse.uwa.edu.au/~pk/research/matlabfns/PhaseCongruency/Docs/convexpl.html>.
- [4] D. Boukerroui, J.A. Noble, and M. Brady. On the choice of band-pass quadrature filters. *Journal of Mathematical Imaging and Vision*, 21(1-2):53–80, 2004.
- [5] M. Unser, D. Sage, and D. Van De Ville. Multiresolution monogenic signal analysis using the Riesz–Laplace wavelet transform. *Image Processing, IEEE Transactions on*, 18(11):2402–2418, 2009.
- [6] A.V. Oppenheim and J.S. Lim. The importance of phase in signals. *Proceedings of the IEEE*, 69(5):529–541, May 1981.
- [7] M. Felsberg and G. Sommer. The multidimensional isotropic generalization of quadrature filters in geometric algebra. In *Proc. Int. Workshop on Algebraic Frames For The Perception-Action Cycle*, pages 175–185. Springer-Verlag, 2000.
- [8] M. Felsberg and G. Sommer. The monogenic signal. *Signal Processing, IEEE Transactions on*, 49(12):3136–3144, Dec 2001.
- [9] K. Rajpoot, V. Grau, and J.A. Noble. Local-phase based 3D boundary detection using monogenic signal and its application to real-time 3-D echocardiography images. In *Biomedical Imaging: From Nano to Macro, 2009. ISBI '09. IEEE International Symposium on*, pages 783–786, June 2009.
- [10] M. Felsberg and G. Sommer. A new extension of linear signal processing for estimating local properties and detecting features. In *22 DAGM Symposium Mustererkennung*, pages 195–202. Springer-Verlag, 2000.

- [11] L. Wietzke, G. Sommer, and O. Fleischmann. The geometry of 2D image signals. In *Computer Vision and Pattern Recognition, 2009 IEEE Conference on*, pages 1690–1697, June 2009.
- [12] K. Liu, H. Skibbe, et al. Rotation-invariant HOG descriptors using Fourier analysis in polar and spherical coordinates. *IJCV*, 106(3):342–364, 2014.
- [13] P. Kovési. Symmetry and asymmetry from local phase. In *Tenth Australian Joint Conference on Artificial Intelligence*, volume 190, 1997.
- [14] C.P. Bridge and J.A. Noble. Object localisation in fetal ultrasound images using invariant features. In *Biomedical Imaging: From Nano to Micro, IEEE International Symposium on*, April 2015.
- [15] B. Rahmatullah, A.T. Papageorghiou, and J.A. Noble. Integration of local and global features for anatomical object detection in ultrasound. In *Medical Image Computing and Computer-Assisted Intervention*, volume 7512 of *Lecture Notes in Computer Science*, pages 402–409. Springer Berlin Heidelberg, 2012.
- [16] P. Kovési. Phase congruency detects corners and edges. In *The Australian Pattern Recognition Society Conference: DICTA 2003*, pages 309–318, 2003.
- [17] M. Mulet-Parada and J.A. Noble. 2D+T acoustic boundary detection in echocardiography. *Medical Image Analysis*, 4(1):21–30, 2000.
- [18] K. Rajpoot, J.A. Noble, V. Grau, and N. Rajpoot. Feature detection from echocardiography images using local phase information. In *12th Medical Imaging Understanding and Analysis*, 2008.
- [19] R.V. Stebbing and J.A. Noble. Delineating anatomical boundaries using the boundary fragment model. *Medical Image Analysis*, 17(8):1123–1136, 2013.
- [20] M. Felsberg. Optical flow estimation from monogenic phase. In B. Jähne, R. Mester, E. Barth, and H. Scharf, editors, *1st International Workshop on Complex Motion*, volume 3417 of *LNCS*, pages 1–13, 2006.
- [21] M. Alessandrini, A. Basarab, H. Liebgott, and O. Bernard. Myocardial motion estimation from medical images using the monogenic signal. *Image Processing, IEEE Transactions on*, 22(3):1084–1095, March 2013.
- [22] V. Grau, H. Becher, and J.A. Noble. Phase-based registration of multi-view real-time three-dimensional echocardiographic sequences. In *Medical Image Computing and Computer-Assisted Intervention MICCAI 2006*, volume 4190, pages 612–619. 2006.
- [23] V. Grau and J.A. Noble. Adaptive multiscale ultrasound compounding using phase information. In *Medical Image Computing and Computer-Assisted Intervention MICCAI 2005*, volume 3749 of *Lecture Notes in Computer Science*, pages 589–596. Springer Berlin Heidelberg, 2005.
- [24] K.A. Patwardhan. Symmetry and appearance based automated detection of salient anatomical regions in ultrasound. In *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*, pages 4426–4428, Aug 2012.
- [25] M.A. Maraci, R. Napolitano, A. Papageorghiou, and J.A. Noble. Object classification in an ultrasound video using LP-SIFT features. In *Medical Computer Vision: Algorithms for Big Data, MICCAI Workshop On*, pages 71–81. Springer International Publishing, 2014.
- [26] M. Felsberg and G. Sommer. The monogenic scale-space: A unifying approach to phase-based image processing in scale-space. *Journal of Mathematical Imaging and Vision*, 21(1-2):5–26, 2004.
- [27] N. Wadhwa, M. Rubinstein, F. Durand, and W.T. Freeman. Riesz pyramids for fast phase-based video magnification. 2014.
- [28] L. Wietzke and G. Sommer. The signal multi-vector. *Journal of Mathematical Imaging and Vision*, 37(2):132–150, 2010.

- [29] L. Wietzke, O. Fleischmann, and G. Sommer. 2D image analysis by generalized Hilbert transforms. In *European Conference on Computer Vision*. Springer-Verlag, 2008.
- [30] C. Wachinger, T. Klein, and N. Navab. The 2D analytic signal for envelope detection and feature extraction on ultrasound images. *Medical Image Analysis*, 16(6):1073–1084, 2012.