Enhanced Shift and Scale Tolerance for Rotation Invariant Polar Matching With Dual-Tree Wavelets

James D. B. Nelson and Nick G. Kingsbury, *Member, IEEE*

***Abstract*—Polar matching is a recently developed shift and rotation invariant object detection method that is based upon dual-tree complex wavelet transforms or equivalent multiscale directional filterbanks. It can be used to facilitate both keypoint matching, neighborhood search detection, or detection and tracking with particle filters. The theory is extended here to incorporate an allowance for local spatial and dilation perturbations. With experiments, we demonstrate that the robustness of the polar matching method is strengthened at modest computational cost.**

***Index Terms*—Feature extraction, object detection, wavelet transforms.**

I. Introduction

SOME well known important wavelet properties include (bi)orthogonality, Besov regularity, compact support, and symmetry. Commonly, however, object detection problems require the consideration of extra properties because two objects are often defined to be in the same class if one object is similar to some transformation of the other. If wavelets are to be used for object detection tasks, then either the objects must somehow be normalised first, or the wavelet coefficients must be invariant to certain transformations.

In practice, normalization can be difficult. For translation invariance, some previous works have implemented a variant of the "spin-cycle" method of Coifman and Donoho [2] whereby extra training samples are created by shifting the original ones. A more elegant and computationally efficient method is to construct transforms which are themselves invariant.

The shiftable wavelet, introduced by Simoncelli *et al.* [17], satisfies a slightly weaker condition than shift invariance but is less redundant than the spin cycle. Monogenic wavelets [1], [13] are a 2-D extension of 1-D analytic wavelet transforms. They are rotationally steerable but not very directionally selective and can be expensive to compute. The dual-tree complex wavelet transform (DTCWT), introduced by Kingsbury [9], [15], has good shift invariance and offers low redundancy with good computational efficiency and good directional selectivity. Moreover, a recent extension of the DTCWT, known as polar matching [10], also possesses approximate rotation invariance. Unlike earlier DTCWT rotation invariant work of Hill *et al.* [8], polar matching retains the phase information of the complex coefficients and, therefore, represents a richer descriptor.

Polar matching applications include keypoint matching, neighborhood search detection, and detection and tracking with particle filters [12]. Keypoint matching proceeds by first using a keypoint detector to find salient features such as edges, corners, and blobs in two different images. Features are then extracted from keypoints in one image and compared to those in the second image. The pairwise correlation scores can then be binned over different shifts, dilations, and rotations to allow for limited affine transformations between the two different images. Since keypoints will not necessarily be centered on exactly the same object components from one image to the next, robustness to small displacement errors can be the key to the success of the method. The scale invariant feature transform (SIFT) offers improved robustness to changes in image scale compared with earlier keypoint detectors such as the Harris corner detector [7]. In a similar way to SIFT [11], DTCWT keypoints can be established in scale and space [4]. Contrary to SIFT, the polar keypoint matching method does not choose a dominant orientation for each keypoint but rather makes use of correlation scores at all rotations. In [4], the DTCWT method was shown to be more robust to rotation, and gave less redundant keypoints, than SIFT.

A second application is to use polar matching in a template matching approach for object detection in video [12]. Assuming that we have access to one or more examples of the target or object of interest, we can use polar matching to search an unknown test image by extracting features from some neighborhood or window in the test image and correlating each one with a template stored in a database. Hence, a correlation surface can be obtained. The steepness of the correlation surface about the maximum can be controlled to some extent by the template size and the choice of wavelet decomposition levels used. However, the size of the template relative to image size will be determined by the application, data, and object of interest. In practice, the correlation surface is computed over a discretised set of points. A full, exhaustive computation would involve the extraction of test features at every pixel, or perhaps subpixel, in the search region. If such an approach proves intractable then it becomes necessary to calculate the correlation surface over a sparser set of locations. On the other hand, a sparser search carries with it the risk of missing the correlation peak altogether. In this setting, robustness to shifts allows sparser sampling of the test features.

In the same context, particle filtering has also been used with polar matching [12]. Here, polar matching scores are computed at each particle location to form the observational model. With added shift and scale tolerance the location and scale of the particles becomes less critical and can either add robustness to the observational model or allow fewer particles to be used.

Manuscript received September 17, 2009; revised March 10, 2010; accepted March 10, 2010. Date of publication August 26, 2010; date of current version February 18, 2011. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Sharathchandra Pankanti.

The authors are with the Signal Processing and Communications Group, University of Cambridge, Cambridge CB2 1TN, U.K. (e-mail: jdbn2@cam.ac.uk; ngk@eng.cam.ac.uk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2010.2069711

In all of these applications, scale tolerance would add robustness to unknown changes in distance between sensor and target.

In this paper we shall consider the question of how to efficiently incorporate an allowance for an unknown local spatial and dilation perturbation into the polar matching method. The next section summarizes the basic polar matching method and Section III introduces a shift tolerant extension to polar matching. This is further extended to include scale tolerance in Section IV. Experiments on a 73 image dataset from Caltech are described in Section V. Finally, conclusions are drawn in Section VI.

II. Polar Matching

At each scale level, the 2-D DTCWT decomposes an image into six subbands [15]. Each subband coefficient can be thought of as a response to a bandpass directional filter at a particular location. Together with their complex conjugates, the coefficients constitute 12 different directions, regularly spaced at $(30k - 15)^\circ$, for $k = 1, \ldots, 12$. For the purposes of polar matching the 2-D real and imaginary impulse responses in the 45° and 135° directions are modified, as described in [10] and depicted in Fig. 1, to have center frequencies that match those of the other directions or subbands. In addition, the phases of the six band outputs are all centered to zero by a simple multiplication of $\{j, -j, j, -1, 1, -1\}$, respectively. In doing so, six opposing directions can be obtained by conjugating the six complex subband coefficients.

As illustrated in Figs. 2 and 3, the elementary form of the polar matching method samples these six subband coefficients at 12 points around a circle and at one point at the circle center. The coefficients are then assembled into what is known as a polar matching matrix (P-matrix), thus

$$\mathbf{P} = \begin{bmatrix} m_1 & j_1 & k_1 & l_1 & a_1 & b_1 & c_1 \\ m_2 & i_2 & j_2 & k_2 & l_2 & a_2 & b_2 \\ m_3 & h_3 & i_3 & j_3 & k_3 & l_3 & a_3 \\ m_4 & g_4 & h_4 & i_4 & j_4 & k_4 & l_4 \\ m_5 & f_5 & g_5 & h_5 & i_5 & j_5 & k_5 \\ m_6 & e_6 & f_6 & g_6 & h_6 & i_6 & j_6 \\ m_1^* & d_1^* & e_1^* & f_1^* & g_1^* & h_1^* & i_1^* \\ m_2^* & c_2^* & d_2^* & e_2^* & f_2^* & g_2^* & h_2^* \\ m_3^* & b_3^* & c_3^* & d_3^* & e_3^* & f_3^* & g_3^* \\ m_4^* & a_4^* & b_4^* & c_4^* & d_4^* & e_4^* & f_4^* \\ m_5^* & l_5^* & a_5^* & b_5^* & c_5^* & d_5^* & e_5^* \\ m_6^* & k_6^* & l_6^* & a_6^* & b_6^* & c_6^* & d_6^* \end{bmatrix}$$

where the subscripts k determine the subband orientations $(30k - 15)^\circ$, the coefficients labelled m are taken from the midpoint, and the coefficients $a, b, \ldots l$ determine the locations of the sample points as in Fig. 2. The arrangement of the DTCWT coefficients ensures that each 30° rotation of the image about the center point of the sampling circle produces a cyclical shift by one element of each of the columns of the polar matching matrix (P-matrix).

Given two images, one a $30k^\circ$ rotation of the other, a summation of the column-wise correlations between the two P-matrices will give a response curve with respect to rotation angle, and with a maximum at a shift of k. Hence, the location of the correlation peak can be used to estimate the difference in orientation angle between two similar objects. However, when computed directly, this correlation response curve will only give responses at increments of 30°. The resolution can be improved by performing the correlations as a sum of zero padded dot products in the Fourier domain before using a single inverse FFT to obtain an upsampled correlation result. Typically, the original 12 samples are upsampled by a factor of 4 to obtain 7.5° rotational spacing. One then arrives at a response curve as a function of orientation, sampled at $7.5k^\circ$, for $k = 0, \ldots, 47$.

The P-matrix represents an approximately rotation invariant feature under the operation of correlation. In a similar fashion to the Fourier–Mellin transform [3], [16], polar matching exploits radial sampling to transform rotations of the original object into shifts in the feature space. Unlike Fourier–Mellin, the polar matching descriptor is constructed from coefficients that are well localised in both space and frequency. Although the Fourier–Mellin descriptors are also scale invariant, they are not designed to detect the location of a template by searching a larger image. Accordingly, they are usually applied to image retrieval [3] or global registration [6] problems where the two images to be compared have either been captured or preprocessed

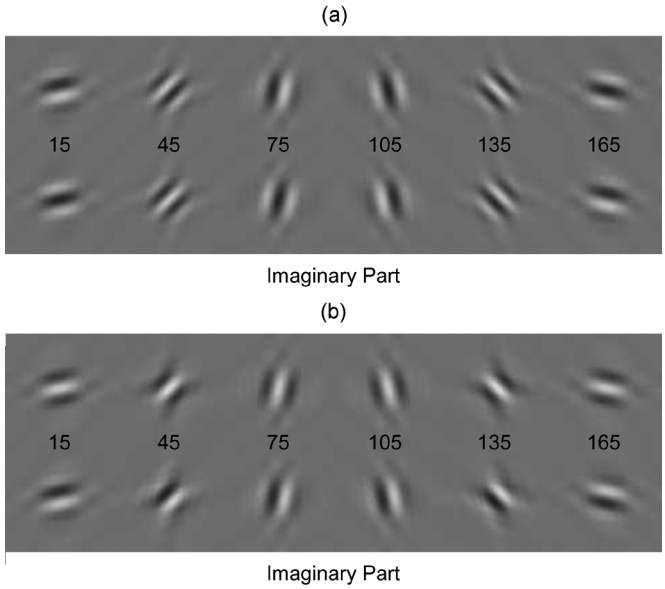

Fig. 1. In the interests of rotation invariance, the DTCWT 2-D real and imaginary impulse responses in the 45° and 135° directions are modified. Taken from [10]. (a) Dual-tree complex wavelets: real part. (b) Modified complex wavelets: real part.

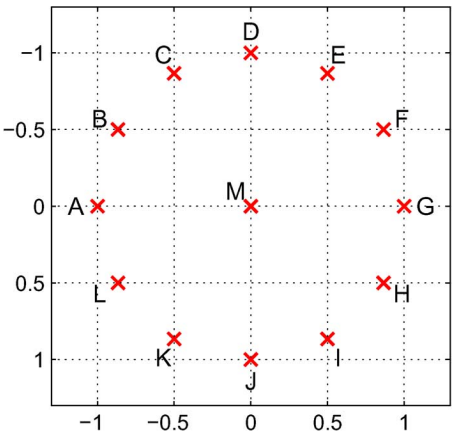

Fig. 2. Sampling locations of the DTCWT coefficients for the polar matching method. Taken from [10].

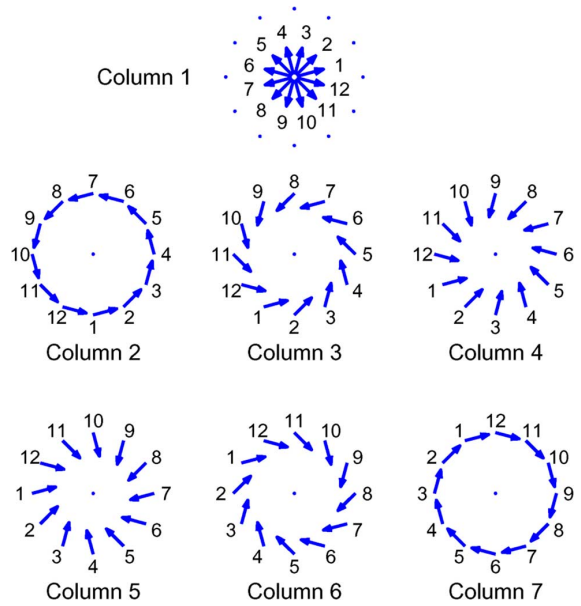

Fig. 3. Locations and orientations of the DTCWT coefficients and how they are arranged in the polar matching matrix (P-matrix). Each orientation describes a coefficient, or conjugate, of one of the six subbands. The numbers denote the orientation and the P-matrix row position. Taken from [10].

such that the background or clutter is removed. A local version of the Fourier–Mellin can be computed using a windowed Fourier approach [5]. Strictly speaking, if this was used directly to perform a template matching search on an image then a windowed Fourier transform would have to be computed at every single pixel in the search image. This is not necessary for polar matching.

To enrich the polar matching descriptor, further scale levels, sampling circles, and color channels may be considered by appending coefficients as extra columns of the P-matrix. For the experiments carried out in this paper, two sampling rings and their center point from the third finest level are used, together with one ring and center point at the fourth finest, and just the center point at the fifth finest level. This is illustrated in Fig. 4. The sampling pattern used in our experiments was chosen to be the same one used in a target detection and tracking application of polar matching [12]. Generally, the scale levels and sampling ring radii are chosen such that the outer rings approximately overlap at least part of the object boundary. For simplicity, color information was converted into monochrome values prior to any processing. This results in 13 columns of 12 coefficients from the third finest level, seven columns from the fourth, and one column from the fifth. Hence, the number of columns $L = 21$.

Consider the column-wise Fourier transform of a template P-matrix taken about some center point. As stated previously, the 12 Fourier coefficients in each column are periodically extended by a factor of four to generate $K = 48$ rows. In practice, there are computational short-cuts for this process, which will be discussed later. Denote the (k, ℓ)th element of the resulting matrix by $h_{k,\ell}$ and its conjugate by $\overline{h}_{k,\ell}$. Likewise, let $f_{k,\ell}(\mathbf{x})$ be the elements in the extended column-wise Fourier transform of a test image P-matrix taken about the point $\mathbf{x}$. The polar matching operation between the two can be expressed as

$$g(\mathbf{x};\theta) = \Re\left\{\frac{1}{K}\sum_{k=0}^{K-1}\exp\left(\frac{2\pi i\theta k}{K}\right)\sum_{\ell=0}^{L-1}\chi_\ell[k]\,\overline{h}_{k,\ell}f_{k,\ell}(\mathbf{x})\right\}.$$

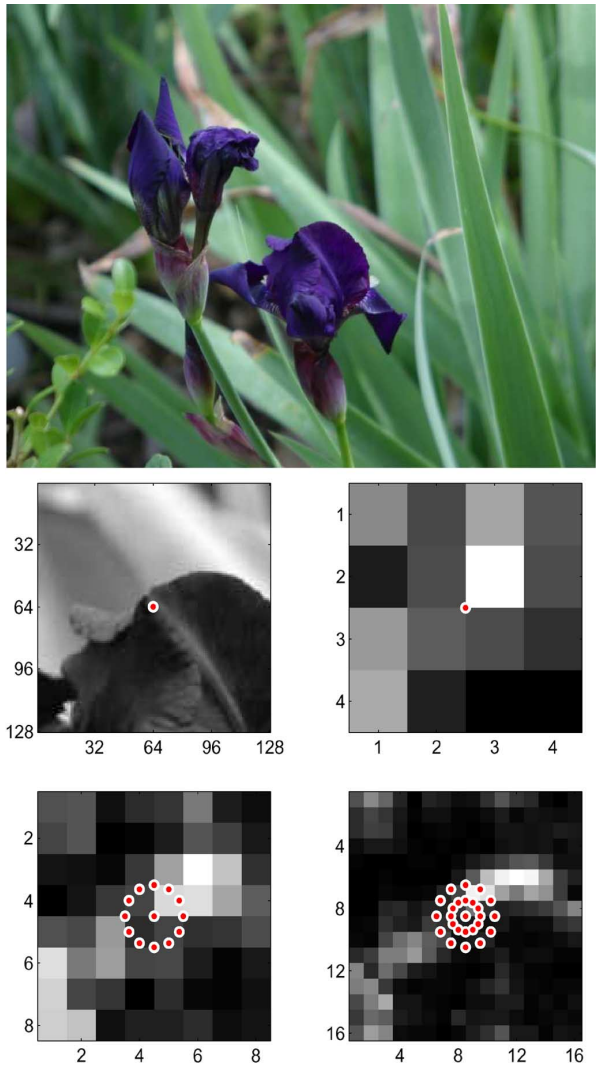

Fig. 4. Multiple scale level sampling locations of the DTCWT coefficients for the polar matching method. A 128×128 subimage is extracted from the center of the image (top most) and decomposed into three levels of detail. Upper left: original subimage. Upper right: fifth finest level. Bottom left: fourth finest level. Bottom right: third finest level. For illustrative purposes, only the absolute values of the DTCWT coefficients in subband 1 are shown. In practice, the real and imaginary parts are used from all subbands. The original image is "201.jpg" from Caltech's "PP_Toys_03" full resolution dataset [14].

The element-by-element products between the columns of the two matrices are carried out in the second summation. Before the products are taken, the elements are multiplied by the indicator function $\chi : \mathbb{Z} \mapsto \{0, 1\}$ which, for each column ℓ, takes zeros over an appropriate part of the Fourier domain and ones elsewhere; this can be seen as an ideal bandpass filter. The inverse Fourier transform is then carried out in the outer summation over k.

The part of the spectrum to be zero padded should be tailored differently to suit each column of the P-matrix. In particular, consider the P-matrix formed at the center of rotation of a single step edge. As the edge is rotated, the response of column 1 will vary as a lowpass function. Columns 2 and 7 will vary slightly quicker as bandpass functions, columns 3 and 6 quicker still, and columns 4 and 5 the quickest as highpass functions. The rate of change depends upon the subband orientation with respect to the radial direction. Denoting this angle by α, and referring to

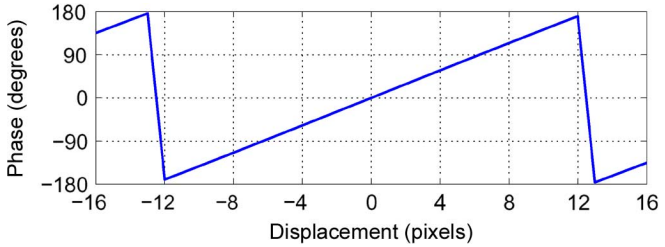

Fig. 5. Step edge orientated at 15° to the horizontal, and placed in the center of a 128 × 128 subimage is decomposed by the DTWCT. The phase angle of the center DTCWT coefficient from subband 1 at the fourth finest scale level is plotted with respect to vertical displacement. Note how the phase completes approximately 3/4 of a cycle over the sampling period of $2^4 = 16$ pixels.

Fig. 3, we have $\alpha = 75°$ for columns 2 and 7, 45° for columns 3 and 6, and 15° for columns 4 and 5. Generally, the center frequency of the columns is proportional to $\cos\alpha$.

Now consider the DTCWT decomposition of a step edge orientated at 15° to the horizontal, placed in the center of a 128 × 128 subimage. Fig. 5 shows that the phase response of the center coefficient at the fourth finest scale level taken from subband 1 (oriented such that the stripe direction is parallel to the edge direction) shifts by almost $180° = \pi$ radians over a displacement of 12 pixels (note that 12 pixels is equivalent to $12/2^4 = 3/4$ samples at the fourth finest scale level). Hence, over one sample interval at the fourth finest scale level, the phase will shift by $4\pi/3$ radians. Therefore, the rate of phase change with respect to a rotating step edge at a radius of one sample will be approximately equal to $(4\pi/3)\cos\alpha$.

Since the column 1 coefficients vary as a lowpass function

$$\chi_1[k] = \begin{cases} 1, & \text{if } 0 \le k \le 5, \text{ and } 43 \le k \le 47 \\ 0, & \text{if } 6 \le k \le 42. \end{cases}$$

Likewise, the center points from any other scale level will be multiplied by χ_1. For columns 2 to 7, since the rate of phase change is approximately $(4\pi/3)\cos\alpha$, it follows that, for $K = 48$:

$$\begin{aligned} \chi_2[k] &= \chi_7[k] = \chi_1\left[(k+1) \bmod 48\right] \\ \chi_3[k] &= \chi_6[k] = \chi_1\left[(k+3) \bmod 48\right] \\ \chi_4[k] &= \chi_5[k] = \chi_1\left[(k+4) \bmod 48\right]. \end{aligned}$$

The inner rings at other levels will similarly be multiplied by $\chi_2, \ldots, \chi_7$. Since the outer ring at the third finest level (see Fig. 4) has twice the radius of the inner ring, the rate of phase change will double. If the outer rings are assembled into columns 16 to 21, then

$$\begin{aligned} \chi_{16}[k] &= \chi_{21}[k] = \chi_1\left[(k+2) \bmod 48\right] \\ \chi_{17}[k] &= \chi_{20}[k] = \chi_1\left[(k+6) \bmod 48\right] \\ \chi_{18}[k] &= \chi_{19}[k] = \chi_1\left[(k+6) \bmod 48\right]. \end{aligned}$$

Generally $\chi_\ell[k] = \chi_1\left[\left(k + \text{round}\left(\max\left\{\frac{4\rho\pi}{3}\cos\alpha_\ell, 6\right\}\right)\right) \bmod 48\right]$ where α_ℓ is the subband orientation with respect to the radial direction of the filters. Note that the maximum allowable shift of the function χ_1 is $k+6$ (any further shifts would make the high pass functions χ_{18} and χ_{19} become band pass functions). The radius of the sampling circle ρ is measured in samples of the respective coefficient space. In our experiments, the fourth finest level has a ring of unit radius, and the third finest level has an inner ring of radius 1 and outer ring of radius 2.

The fast Fourier transform can be used to speed up the polar matching computation. The inverse DFT only needs to be done once on the accumulated sum of all the columns of the dot-product matrix, rather than on every column. Alternatively, if there exists prior information about the orientation of the target, the inverse DFT need only be computed for a subset of θ. For example, when tracking objects using video it is reasonable to assume that the target orientation rate of change is bounded. Finally, the real component $\Re$ is taken in order to return the purely real correlation intensity. For simplicity, and to aid the development of the shift and scale tolerant polar matcher in the next section, this is first rewritten as

$$g(\mathbf{x};\theta) = \Re\left\{\sum_{k=0}^{K-1}\sum_{\ell=0}^{L-1} w_{k,\ell}(\theta)\overline{h}_{k,\ell} f_{k,\ell}(\mathbf{x})\right\} \tag{1}$$

where

$$w_{k,\ell}(\theta) = \frac{1}{K}\exp\left(\frac{2\pi i\theta k}{K}\right)\chi_\ell[k].$$

Now put $j = k + K\ell$, and $n = KL$. Define the column vector $\mathbf{h} = (h_j)_0^{n-1} \in \mathbb{C}^n$ as $\mathbf{h} \triangleq [h_{0,0}, \ldots, h_{K-1,0}, h_{0,1}, \ldots, h_{K-1,1}, \ldots h_{K-1,L-1}]^T$. That is, $\mathbf{h}$ concatenates the columns of the matrix $(h_{k,\ell})$. Likewise, form $\mathbf{w}$ and $\mathbf{f}$. We then have

$$g(\mathbf{x};\theta) = \Re\left\{\sum_{j=0}^{n-1} w_j(\theta)\overline{h}_j f_j\right\}$$

which is now just a weighted inner product, namely $\Re\{\mathbf{h}^H\mathbf{W}_\theta\mathbf{f}(\mathbf{x})\}$, where the superscript ^{H} denotes complex conjugate (Hermitian) transpose. For each orientation θ, the matrix $\mathbf{W}_\theta = \text{Diag}(\mathbf{w}(\theta))$ is diagonal. This summation only takes place over $n/4 = 252$ nonzero terms because of the Fourier domain zero padding.

To summarise, the polar matching operation between the template Fourier P-vector $\mathbf{h}$ and test image Fourier P-vector $\mathbf{f}$ about the point $\mathbf{x}$ is

$$g(\mathbf{x};\theta) \triangleq (\mathbf{f}(\mathbf{x}) \star \mathbf{h})(\theta) \triangleq \Re\left\{\mathbf{h}^H\mathbf{W}_\theta\mathbf{f}(\mathbf{x})\right\}. \tag{2}$$

Since both $\mathbf{h}$ and $\mathbf{W}_\theta$ are independent of the test image, the product $\mathbf{h}^H\mathbf{W}_\theta$ can be precomputed and stored. The operator $\Re$ also reduces computation by a factor of 2.

Choosing a sampling circle radius of 1 or 2 will cause some overlapping of test feature samples $\mathbf{f}(\mathbf{x})$ which can be exploited for a modest computational speed-up. Including multiple scale levels effectively increases the number L of P-matrix columns. Note, however, that in the polar matching operation (1) only the ℓth column of h is compared (via dot product) with the ℓth column of f. In other words, only template coefficients from scale level 3, say, are compared with test coefficients from scale level 3. Although multiple scale levels are used, no cross-scale comparisons are computed. This means that the original method is not, in general, scale invariant. In contrast, the terminology "scale tolerance" refers to the ability of the matcher to deal with the situation where the template is a zoomed in (or out) version of the test image.

In the following, the polar matching approach is strengthened so that a P-matrix constructed from a center point location $(x+\delta x, y+\delta y)$ and scale $s+\delta s$ will still obtain a large correlation score when matched with the same image centered on (x, y) at scale s.

III. Shift Tolerant Algorithm

In this section, we present an extension to the original polar matching method to incorporate an allowance for larger local spatial displacement errors δx and δy. In doing so, it will be seen that the steepness of the correlation surface about the maximum is reduced.

As shown previously, let $\mathbf{h}$ and $\mathbf{f}$ be the Fourier transforms of the polar matching features of the template subimage and test image, respectively. It is usually advantageous to normalise the features by a scalar such that $\|\mathbf{h}\| = \|\mathbf{f}\| = 1$, in order to give improved resilience to varying contrast levels between images. After normalization, the features can be considered mappings from the original pixel coordinates to the n-dimensional complex hypersphere $S^n = \{\mathbf{x} \in \mathbb{C}^n : \|\mathbf{x}\| = 1\}$. That is

$$\mathbf{h}, \mathbf{f} : \mathbb{R}^2 \mapsto S^n.$$

From (2) polar matching, denoted by $\star$, is the real part of a sesquilinear form

$$\star : S^n \times S^n \mapsto [-1, 1]^K \subset \mathbb{R}^K$$

from the complex n-sphere to the real K-cube, where K is the number of bins for θ. (We choose $K = 48$ in our experiments.) Now define $\Delta\mathbf{h} = \mathbf{h}(\cdot + \Delta\mathbf{x}) - \mathbf{h}$. We assume that $\mathbf{h}$ is linear with respect to small spatial shifts $\Delta\mathbf{x}$. That is

$$\Delta\mathbf{h} = \mathbf{J}\Delta\mathbf{x}, \qquad \mathbf{J} = \left[\frac{\partial\mathbf{h}}{\partial x}, \frac{\partial\mathbf{h}}{\partial y}\right] \in \mathbb{C}^{n\times 2}.$$

This is equivalent to a first order Taylor series expansion of $\mathbf{h}(\cdot + \Delta\mathbf{x})$, which is reasonably accurate because of the smooth bandlimited nature of the complex wavelet coefficients. Then

$$\begin{aligned}\mathbf{f} \star \mathbf{h}(\cdot + \Delta\mathbf{x}) &= \mathbf{f} \star (\mathbf{h} + \Delta\mathbf{h}) \\ &= \mathbf{f} \star (\mathbf{h} + \mathbf{J}\Delta\mathbf{x}) \\ &= \Re\left\{(\mathbf{h} + \mathbf{J}\Delta\mathbf{x})^H \mathbf{W}_\theta \mathbf{f}\right\}.\end{aligned}$$

As Fig. 6 illustrates, the problem that shift tolerant polar matching addresses is to find the maximum shifted polar matching correlation score with respect to an unknown small shift $\Delta\mathbf{x}$. Since h and $\mathbf{f}$ are both normalised to unit length, the aim is to get close to $\mathbf{W}_\theta\mathbf{f}$ by moving a distance of $\Delta\mathbf{x}$, from $\mathbf{h}$, along the surface of the hypersphere in a direction orthogonal to the radial vector $\mathbf{h}$. That is

$$\mathbf{x}_\theta = \arg\max_{\Delta\mathbf{x}} \Re\left\{\frac{(\mathbf{h} + \mathbf{J}\Delta\mathbf{x})^H}{\|\mathbf{h} + \mathbf{J}\Delta\mathbf{x}\|}\mathbf{W}_\theta\mathbf{f}\right\}, \quad \mathbf{J}^H\mathbf{h} = \mathbf{0}. \tag{3}$$

Ideally, we want $\mathbf{x}_\theta$ such that

$$\mathbf{h} + \mathbf{J}\mathbf{x}_\theta = \alpha\mathbf{W}_\theta\mathbf{f}, \quad \mathbf{J}^H\mathbf{h} = \mathbf{0} \tag{4}$$

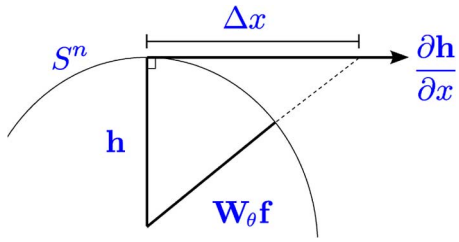

Fig. 6. Shift tolerance problem, simplified in 2-D space. Find the shift distance $\Delta\mathbf{x}$ along the surface of the hypersphere S^n, such that the unit vector $\mathbf{h}$ is rotated into $\mathbf{W}_\theta\mathbf{f}$.

for some $\alpha \in \mathbb{R}^+$. However, because S^n is a much larger dimensional space than the domain of $\mathbf{x}_\theta$, which has only two dimensions, this is an overdetermined set of equations. Instead, the solution, in the least squares sense, is

$$\begin{aligned}\mathbf{x}_\theta &= \Re\left\{(\mathbf{J}^H\mathbf{J})^{-1}\right\}\Re\{\mathbf{J}^H\mathbf{W}_\theta\mathbf{f}\} \\ &= \Re\left\{(\mathbf{J}^H\mathbf{J})^{-1}\right\}(\mathbf{f} \star \mathbf{J})(\theta).\end{aligned} \tag{5}$$

Note that, since $\|\mathbf{h}\| = 1$ and $\mathbf{h}^H\mathbf{J} = 0$, the denominator of (3) is

$$\|\mathbf{h} + \mathbf{J}\Delta\mathbf{x}\| = \sqrt{1 + \mathbf{x}_\theta^H\Re\{\mathbf{J}^H\mathbf{J}\}\mathbf{x}_\theta}.$$

Hence, substituting $\mathbf{x}_\theta$ from (5) into the right hand side of (3) gives shift tolerant polar matching

$$g^+(\mathbf{x};\theta) \triangleq \frac{(\mathbf{f} \star \mathbf{h}) + (\mathbf{f} \star \mathbf{J})^T\mathbf{A}(\mathbf{f} \star \mathbf{J})}{\sqrt{1 + (\mathbf{f} \star \mathbf{J})^T\mathbf{A}(\mathbf{f} \star \mathbf{J})}} \tag{6}$$

where $\mathbf{A} = \Re\{(\mathbf{J}^H\mathbf{J})^{-1}\}$. The term $\mathbf{J}^H\mathbf{J}$ is a 2×2 matrix and is independent of the test image term $\mathbf{f}$. Under the reasonable assumption that $\partial\mathbf{h}/\partial x$ and $\partial\mathbf{h}/\partial y$ are linearly independent, $\mathbf{J}^H\mathbf{J}$ is a positive definite matrix and is, therefore, invertible. All other terms involve polar matching operations on the test image with the template and two spatial derivatives of the template. Therefore, the template $\mathbf{h}$, Jacobian $\mathbf{J}$, and matrix $\mathbf{A}$ should be precomputed and stored in memory.

Compared with the original polar matching method, which just requires computation of $\mathbf{f} \star \mathbf{h}$, this shift tolerant version also requires $\mathbf{f} \star \mathbf{J} = \mathbf{f} \star [\partial\mathbf{h}/\partial x, \partial\mathbf{h}/\partial y]$. This comprises two weighted inner products, each of similar complexity to $\mathbf{f} \star \mathbf{h}$. Alternatively, using the inverse fast Fourier transform to compute the polar matching operations, we now require three FFTs (rather than one) per test location $\mathbf{x}$. Since an upsampling of four is used, these are 48-point FFTs. Once $\mathbf{f} \star \mathbf{J}$ has been computed, there is a minor additional cost for each $\theta = 0, \ldots, 47$ and $\mathbf{x}$, of six multiplications and two adds to compute the quadratic form $(\mathbf{f} \star \mathbf{J})\mathbf{A}(\mathbf{f} \star \mathbf{J})^T$, one add for the numerator, one add and a square root for the denominator, and a division.

However, the overall computation of both the original and tolerant method is often dominated by calculating the features $\mathbf{f}$ at each point $\mathbf{x}$. This overhead involves a DTCWT to decompose the test image. More crucially, at each $\mathbf{x}$ it also requires bandpass interpolations to generate the coefficients around the sampling circles. As we will illustrate with some experiments in Section V, the extra costs of the new method are likely to be worthwhile in order to reduce the sensitivity to displacement error.

IV. Scale Tolerant Algorithm

Shift tolerant matching can be extended quite naturally to shift and scale tolerant matching. We introduce a dilation variable ψ, such that

$$\mathbf{h}(\mathbf{x};\psi) \triangleq \mathbf{h}(\psi\mathbf{x}).$$

Now

$$\mathbf{h} : \mathbb{R}^3 \mapsto \mathbb{C}^n.$$

Define $\Delta\mathbf{h} \triangleq \mathbf{h}(\mathbf{x}+\Delta\mathbf{x};\psi+\delta\psi) - \mathbf{h}(\mathbf{x};\psi)$. Then, for small $\Delta\mathbf{x}$ and $\delta\psi$, assume

$$\Delta\mathbf{h} = \mathbf{J}\begin{bmatrix}\Delta\mathbf{x}\\ \delta\psi\end{bmatrix}, \qquad \mathbf{J} = \left[\frac{\partial\mathbf{h}}{\partial x}, \frac{\partial\mathbf{h}}{\partial y}, \frac{\partial\mathbf{h}}{\partial\psi}\right] \in \mathbb{C}^{n\times 3}.$$

i.e., this is a first order Taylor series expansion of the shifted and dilated template $\mathbf{h}(\mathbf{x}+\Delta\mathbf{x};\psi+\delta\psi)$. Similar to the shift tolerant case, we want

$$\mathbf{h} + \mathbf{J}\begin{bmatrix}\mathbf{x}_\theta\\ \psi_\theta\end{bmatrix} = \alpha\mathbf{W}_\theta\mathbf{f}, \quad \mathbf{J}^H\mathbf{h} = \mathbf{0}. \tag{7}$$

The solution, in the least squares sense, is

$$\begin{aligned}\begin{bmatrix}\mathbf{x}_\theta\\ \psi_\theta\end{bmatrix} &= \Re\left\{(\mathbf{J}^H\mathbf{J})^{-1}\right\}\Re\{\mathbf{J}^H\mathbf{W}_\theta\mathbf{f}\}\\ &= \Re\left\{(\mathbf{J}^H\mathbf{J})^{-1}\right\}(\mathbf{f}\star\mathbf{J})(\theta).\end{aligned} \tag{8}$$

The shift and scale tolerant polar matching takes the same form as (6). The only difference is that $\partial\mathbf{h}/\partial\psi$ has been appended as an extra column to $\mathbf{J}$, (4) and (5) are replaced by (7) and (8), and $\mathbf{A}$ is now a 3×3 matrix.

V. Experiments

Caltech's "PP_Toys_03" full resolution dataset [14] was used to investigate the effectiveness of the shift and scale tolerant methods. The 73 image dataset mostly comprises various toys against a grassy or stony background. To simplify reproducibility of results, no attempt was made to find suitable template center points. Instead, templates of size 128×128 pixels were simply taken from the center of each image. For simplicity, prior to any further processing, the RGB values were converted to intensity via $0.3R + 0.6G + 0.1B$. Fig. 4 illustrates the three levels of DTCWT sampling used.

Fig. 7 shows correlation surfaces obtained by performing polar matching between the template and test image regions centered about the template. We compute $g(\mathbf{x};\theta) = \mathbf{h}(\mathbf{x})\star\mathbf{h}(\mathbf{0})$ about a local neighborhood of $\mathbf{x} = (x, y) = \mathbf{0}$. As discussed earlier, the polar matching output $g(x, y;\theta)$ is a function of space (x, y) and orientation θ. Denote the shift tolerant matcher output as g^+, and the shift and scale tolerant output as g^{++}. For each template, the test images are shifted, rotated, and dilated versions of the template. Rotations were performed on the test images to show that the rotational invariance property of polar matching is not significantly affected by the tolerant methods. For each rotation $\Theta = 0, 7.5^\circ, \ldots, 90^\circ$ and dilation $\Psi = 1, 1.05, \ldots, 1.5$, and for each of the 73 test images, we obtain a correlation output $g_{\Theta,\Psi}(x, y;\theta)$. The shift tolerant matcher was applied to the shifted and rotated test images to give the correlation surfaces $g^+_{\Theta,1}(x, y;\theta)$. The shift and scale

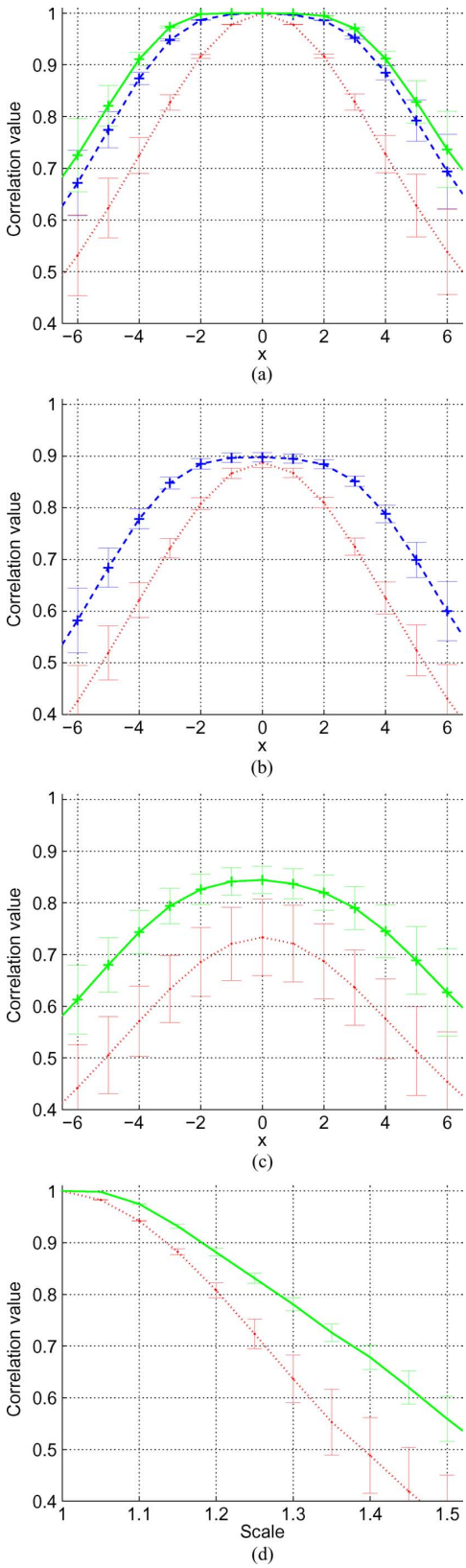

Fig. 7. Mean correlation values over 73 image dataset, with respect to x. (a) Unrotated and undilated. (b) Mean over rotations. (c) Mean over dilations. (d) Mean correlation values with respect to dilation. The original polar matching is plotted with a dotted red line, the shift tolerant method with a blue dashed line with crosses, and the shift and scale tolerant method with a green solid line with crosses. The error bars denote variance over all experiments. (a) $g_{0,1}(x, 0; 0)$. (b) $\underset{0\le\theta\le\pi}{\text{mean}}\, g_{\theta,1}(x, 0;\theta)$. (c) $\underset{\psi=1,1.05,\ldots,1.4}{\text{mean}}\, g_{0,\psi}(x, 0; 0)$. (d) $g_{0,\psi}(0, 0; 0)$.

tolerant matcher was applied to the shifted and dilated test images to give $g^{++}_{0,\Psi}(x,y;\theta)$.

Fig. 7(a) shows the original correlation output g, the shift tolerant output g^{+}, and the shift and scale tolerant correlation output g^{++} with rotation, dilation, and y-shift set to zero. It can be seen that g^{+} and g^{++} show more tolerance to shifts than g. For example, g shows a correlation value of 0.9 and higher is obtained over the interval ± 2 pixels, whereas the tolerant versions obtain 0.9 and higher for the interval ± 4 pixels. This also occurs in the y direction. A similar improvement is evident at correlation values over 0.8 or 0.7.

Interestingly, we see that the shift-scale tolerant method seems a little more tolerant to shifts than the shift tolerant method. The reasons for this not completely clear and are a subject for further investigation.

In Fig. 7(b), g and g^{+} are compared by averaging over all rotations and fixing dilation and y to zero. This shows that g^{+} retains better shift tolerance even when there is a rotational difference between the template and test image. Fig. 7(c) compares g with g^{++} by fixing rotation and y to zero and averaging over all dilations from 1 to 1.4. In Fig. 7(d), the rotations and location are set to zero to compare g with g^{++} over the dilations 1, 1.05, …, 1.5. Fig. 7(c) and (d) show that g^{++} is more tolerant to scale than the original method. Fig. 7(d) also shows that the test image needs to be scaled by a factor of more than 1.35 before the shift and scale tolerant correlation score falls below 70% of the maximum.

Further experiments were carried out on 64×64 base image patch sizes. The second, third, and fourth finest scale levels were used and the radii were kept the same as the original experiments. As might be expected, when we halve the scale of the subimages, the shift tolerance approximately halves. However, the shape of the curves is consistent with the trend of those depicted in Fig. 7.

A sum over all (x,y) values of the correlation surfaces that are above 90% of the theoretical maximum of 1 is computed for the original and tolerant matchers. A ratio of the tolerant matcher score over the original score is then computed for comparison. That is, for the shift tolerant matcher, the ratio

$$\mathcal{M}g^{+}_{\theta,1}(\mathbf{x};\theta) \triangleq \frac{\sum_{g^{+}>0.9} g^{+}_{\theta,1}(\mathbf{x};\theta)}{\sum_{g>0.9} g_{\theta,1}(\mathbf{x};\theta)}$$

gives a measure of the area of the region within 0.9 of the height of the maximum. The mean and standard deviation of $\mathcal{M}$, taken over all experiments are given in Table I for subimage sizes of 128×128 and 64×64.

The behavior with respect to nontargets was also investigated. Ten random points were taken from each of the 73 images at a minimum distance of 64 pixels from the center. Images of size 128×128 centered at these points were correlated by polar matching with each of the 73 templates (also of size 128×128). The histograms of the resulting 53290 correlation scores are shown in Fig. 8. By inspection, a small proportion of the image pairs in this experiment resemble scaled, rotated, and shifted versions of each other. Hence, the histogram overestimates the number of false positives. As might be expected, the discriminative ability tends to diminish a little as more tolerance is added.

The P-matrix of the template features is a 12-by-L complex matrix. In our experiments, $L = 21$. For the shift tolerant method, three complex 12-by-L matrices are required: one for $\mathbf{h}$ and two for $\mathbf{J}$. For the shift-scale tolerant method 4 complex 12-by-L element matrices are required. To sacrifice memory for speed, one could precompute and store $\mathbf{h}^H\mathbf{W}_\theta \in \mathbb{C}^{1\times n}$, with at most $n = 12L$ nonzeros for each θ. For this implementation, the tolerant methods would, in addition, require storing $\mathbf{J}^H\mathbf{W}_\theta \in \mathbb{C}^{m\times n}$, for each θ, where $m = 2$ for the shift tolerant method and $m = 3$ for the shift-scale tolerant method. To extract the template features $\mathbf{h}$ from a 128×128 subimage takes 0.06 s for the original method, 0.18 s for the shift tolerant method, and 0.25 s for the scale tolerant method. To extract the test features from a 128×128 subimage over a 33×33 pixel neighborhood and compute the polar matching operation between the template and all 33×33 test features takes 3.0 s for the original method, 3.6 for the shift tolerant method and 3.8 for the shift-scale tolerant method.[1] For the experiments shown in Fig. 7, the shift tolerant matcher requires 20% more computation time than the original polar matching method. The shift-scale tolerant version requires 27% more time than the original method.

TABLE I
ENERGY IMPROVEMENT RATIOS

	128 × 128		64 × 64	
method/experiment	mean	std	mean	std
$\mathcal{M}g^{+}_{0,1}(\mathbf{x};0)$	3.02	0.53	3.57	1.57
$\mathcal{M}g^{+}_{\theta,1}(\mathbf{x};\theta)$	2.39	0.53	2.52	0.93
$\mathcal{M}g^{++}_{0,1}(\mathbf{x};0)$	4.62	1.91	5.73	2.76
$\mathcal{M}g^{++}_{0,\psi}(\mathbf{x};0)$	2.30	0.77	2.57	1.23

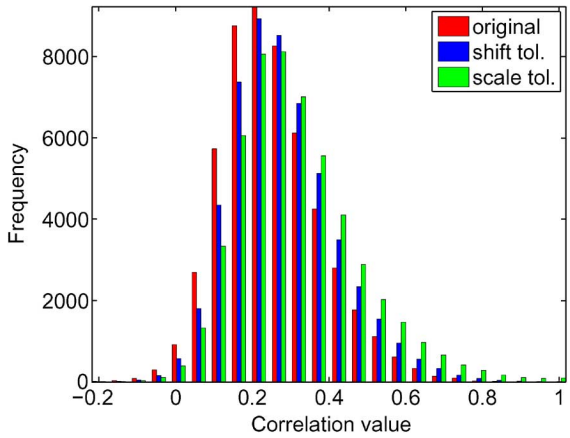

Fig. 8. Histograms of nontarget correlation scores.

VI. CONCLUSION

Fig. 7 and Table I show that the shift and scale tolerant methods behave as intended. Both the shift and shift-scale tolerant methods extend the polar matching shift tolerance by approximately four pixels in both the horizontal and vertical direction when DTCWT levels 3–5 are used for polar matching as in Fig. 4. Furthermore, the scale tolerant method extends the tolerance of polar matching to dilations of around 1.3:1. If, however, the distance between sensor and known object is available then scale tolerance is less important. In this case, shift-rotational invariance and shift tolerance is still important

[1]Using Matlab 7.3.0 with Windows XP and a 2.4 GHz dual core processor.

and the shift tolerant polar matcher may be more appropriate than the shift-scale tolerant version.

In practice the Jacobian $\mathbf{J}$ is approximated by a finite difference. For example, the partial derivative with respect to x is estimated by

$$\frac{\partial \mathbf{h}}{\partial x} \approx \frac{1}{\epsilon} \left(\mathbf{h}(x+\epsilon, y) - \mathbf{h}(x, y) \right).$$

The value $\epsilon = 1/10$ of a pixel was used in all experiments performed in this paper. It is conceivable that the finite difference increment ϵ could be optimised. Furthermore, the finite difference approximation could potentially be replaced with a more sophisticated discrete derivative.

It is also important to note that the shift and scale tolerant approach implied by (3) could be applied to any other correlation operation that can be expressed as a weighted inner product, or sesquilinear form, $\mathbf{h}^H \mathbf{W} \mathbf{f}$. A simple example would be the classic matched filter with $\mathbf{W} = \mathbf{I}$, where $\mathbf{h}$ and $\mathbf{f}$ are simply normalised intensity values of two images.

Investigations of shift and shift-scale tolerant matching to specific applications like keypoint matching and target detection and particle filter tracking in video should make for interesting further work as would an investigation into the choice of scale levels and the number and radii of sampling circles.

REFERENCES

[1] P. X. Bo, M. Brady, R. Highnam, and J. Declerck, "The use of multiscale monogenic signal on structure orientation identification and segmentation," in *Proc. LNCS Int. Workshop Dig. Mammogr.*, 2006, vol. 4046, pp. 601–608.

[2] R. R. Coifman and D. Donoho, "Wavelets and statistics, lecture notes in statistics," in *Translation-Invariant De-Noising*, A. Antoniadis and G. Oppenheim, Eds. New York: Springer-Verlag, 1995, pp. 125–150.

[3] S. Derrode, M. Daoudi, and F. Ghorbel, "Invariant content-based image retrieval using a complete set of Fourier–Mellin descriptors," in *Proc. IEEE Multimedia Comput. Syst.*, 1999, vol. 2, pp. 877–881.

[4] J. Fauqueur, N. Kingsbury, and R. Anderson, "Multiscale keypoint detection using the dual-tree complex wavelet transform," in *Proc. IEEE Conf. Image Process.*, 2006, pp. 1625–1628.

[5] N. Gotze, S. Drue, and G. Hartmann, "Invariant object recognition with discriminant features based on local fast-Fourier Mellin transform," in *Proc. IEEE Int. Conf. Pattern Recognit.*, 2000, vol. 1, pp. 948–951.

[6] X. Guo, Z. Xu, Y. Lu, and Y. Pang, "An application of Fourier–Mellin transform in image registration," in *Proc. IEEE Int. Conf. Comput. Inf. Technol.*, 2005, pp. 619–623.

[7] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proc. Alvey Vis. Conf.*, 1988, pp. 147–151.

[8] P. R. Hill, D. R. Bull, and C. N. Canagarajah, "Rotationally invariant texture features using the dual-tree complex wavelet tranform," in *Proc. IEEE Conf. Image Process.*, 2000, pp. 901–904.

[9] N. G. Kingsbury, "Complex wavelets for shift invariant analysis and filtering of signals," *J. Appl. Comput. Harmon. Anal.*, vol. 10, no. 3, pp. 234–253, 2001.

[10] N. G. Kingsbury, "Rotation-invariant local feature matching with complex wavelets," in *Proc. Eur. Conf. Signal Process.*, 2006, pp. 901–904.

[11] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[12] J. D. B. Nelson, S. K. Pang, S. J. Godsill, and N. G. Kingsbury, "Tracking ground based targets in aerial video with dual-tree complex wavelet polar matching and particle filtering," *Fusion*, 2008.

[13] S. C. Olhede and G. Metikas, "The monogenic wavelet transform," *IEEE Trans. Signal Process.*, vol. 57, no. 9, pp. 3426–3441, Sep. 2009.

[14] P. Perona, Caltech's 'PP_Toys_03' Full Resolution Dataset Website, 2003 [Online]. Available: http://www.vision.caltech.edu/pmoreels/Datasets/PP_Toys_03/FullResolution/TestSingleObjects.tar

[15] I. W. Selesnick, R. G. Baraniuk, and N. G. Kingsbury, "The dual-tree complex wavelet transform," *IEEE Signal Process. Mag.*, vol. 22, no. 6, pp. 123–151, Jun. 2005.

[16] Y. Sheng and H. H. Arsenault, "Experiments on pattern recognition using invariant Fourier–Mellin descriptors," *J. Opt. Soc. Amer.*, vol. 3, no. 6, pp. 771–776, 1986.

[17] E. P. Simoncelli, W. T. Freeman, E. H. Adelson, and D. J. Heeger, "Shiftable multi-scale transforms," *IEEE Trans. Inf. Theory*, vol. 38, no. 2, pt. 2, pp. 587–607, Mar. 1992.

James D. B. Nelson received the Ph.D. degree in applied harmonic analysis from the Mathematics Department at Anglia Polytechnic University, Cambridge, U.K., in 2001.

He held Postdoctoral positions at the University of Cranfield (2001-2004), the University of Southampton (2004-2006), and the University of Cambridge (2006-2010). In 2010, he joined University College London as a Lecturer in the Department of Statistical Science. His research interests include: wavelet analysis; detection, enhancement, and classification for signal and image processing; and machine learning.

Nick G. Kingsbury (M'87) received the honours degree in 1970 and the Ph.D. degree in 1974, both in electrical engineering, from the University of Cambridge, Cambridge, U.K.

From 1973 to 1983 he was a Design Engineer and subsequently a Group Leader with Marconi Space and Defence Systems, Portsmouth, U.K., specializing in digital signal processing and coding, as applied to speech coders, spread spectrum satcomms, and advanced radio systems. Since 1983 he has been a Lecturer in Communications Systems and Image Processing at the University of Cambridge and a Fellow of Trinity College, Cambridge. He was appointed to a Readership in Signal Processing at Cambridge in 2000, and to the position of Professor of Signal Processing in 2007. He is currently head of the Signal Processing and Communications Research Group. His current research interests include image analysis and enhancement techniques, object recognition, motion analysis and registration methods. He has developed the dual-tree complex wavelet transform and is especially interested in the application of complex wavelets and related multiscale and multiresolution methods to the analysis of images and 3-D datasets.