Object Search Using Wavelet-Based Polar Matching for Aerial Imagery

Andy Pickering[1], Nick Kingsbury[2]

Department of Engineering, University of Cambridge
Trumpington Street, Cambridge, CB2 1PZ, UK
[1]andy.pickering@cantab.net
[2]ngk@eng.cam.ac.uk

***Abstract*—A method for finding known objects in aerial and satellite imagery was designed, implemented in MATLAB and tested. This method took a keypoint-based approach to describing objects.**

Several of the steps used in the scheme took advantage of the Dual-Tree Complex Wavelet Transform (DTCWT). These were combined with a new approach which takes the highly localised information encoded in keypoint descriptors and combines it with "global" geometrical information about the target. Many keypoints which corroborate the same approximate location as the predicted target centre can improve match confidence.

The implementation of the above was evaluated both with synthetic imagery, as a means to identify and characterise factors which degrade performance, and also real imagery as a means to demonstrate its "real-world" performance. The real imagery demonstrated success in identifying a range of objects.

I. INTRODUCTION

A. Background

The task of target matching – that is, identifying known objects within imagery – has been of interest for some time. In recent times, the amount of imagery one might want to search has far outstripped the ability of image analysts to search it manually. Thus, the ability to automatically search for known objects is highly desirable.

The use of keypoints – small regions which are to be described by a sparse representation or *descriptor* – forms the basis of one class of approaches to this problem. Such methods have a key advantage that they can easily handle occlusion. This is because, for a typical target, many keypoints will be defined. For example, in the now-famous Scale-Invariant Feature Transform (SIFT) method described by David Lowe, [1] a large number of keypoints are identified when registering a new target, but only a few are typically required to correctly identify that target within a scene.

The use of keypoints for target matching tasks requires 3 steps:

1) Keypoint detection
2) Keypoint description
3) Keypoint matching

B. The Dual-Tree Complex Wavelet Transform

Wavelet decomposition of images has proven to be highly successful for certain tasks; image compression in particular. However, despite initially showing promise for a wide range of image analysis tasks, the Discrete Wavelet Transform has been shown to be inadequate for many of these, due to a lack of shift invariance amongst other problems.

The Dual-Tree Complex Wavelet Transform (DTCWT) [2] overcomes this limitation with modest additional cost, producing complex coefficients whose amplitudes exhibit approximate shift-invariance. These coefficients may be considered to be the response of the image to a set of oriented band-pass filters. As such, the DTCWT coefficients provide a rich description of the image in terms of both spacial frequency and directionality.

Methods have been developed which use the DTCWT coefficients to accomplish all three of the keypoint-based target matching tasks described above. Indeed, much work has been done in adapting the DTCWT for use within the target matching domain. Its frequency response was made closer to rotationally symmetric in [3], while a modified version which more densely samples scale space (the *4S-DTCWT*, so called because it uses 4 interleaved DTCWTs to achieve the finer sampling) was used in [4]. Both these improvements form part of the methodology described herein.

II. THEORY AND DESIGN

A. Keypoint Detection

In order to decide which points in an image to describe, some form of keypoint detector function is required. This allows us to obtain a sparse representation of the object of interest in a way which should be consistent and invariant to the types of changes expected between images (lighting, position, orientation etc.). Usually, one would like to detect small, well-localised features like corners or "blobs", while edges should be rejected.

One such detector which has been widely used is due to Harris [5]. This detector is simple and has been used to great effect in motion tracking in particular.

However, one deficiency of the Harris detector is that it cannot reliably assign a scale to the keypoints detected. Scale information would be useful, as it tells us what size area the keypoint descriptor should be describing. In the context of many image analysis schemes, such as that used by SIFT and the method to be considered here, the scale of a feature tells us not only how large that feature is, but also where it is most prominent within a "scale-space" decomposition of the image.

Several keypoint detectors have been proposed which use the DTCWT coefficients to detect areas of high-frequency energy in multiple directions, whilst also providing scale information about the detected keypoints. Some of these are investigated by Bendale in [6]. Through experimentation, it was decided that the best detector function suited to the target matching problem presented here was a geometric mean of the DTCWT bands. It has the form

$$\tilde{E}_k(x,y) = \prod_{d=1}^{6} |\tilde{H}_k(x,y,d)|^{\frac{1}{6}} \tag{1}$$

Keypoints are defined as the maxima of this function in (x, y, k)-space where $\tilde{E}_k(x,y)$ is the is the energy at point (x, y) and scale k and $\tilde{H}_k(x,y,d)$ is the DTCWT coefficient in band d at this point.

B. Keypoint Description and Matching

Once keypoints in an image can be reliably detected, we would like some robust description of the keypoint region. Perhaps the most famous method for keypoint description is that used by SIFT [1]. This method is based on finding the distribution of oriented gradients at the keypoint.

The design of a descriptor must also include a method for computing the correlation or similarity between two keypoints, by some computation involving the two keypoints' descriptors.

Polar Matching: A DTCWT-based method for keypoint description and matching has been developed by Kingsbury [3]. This method is based on band-limited sampling from a given scale of the DTCWT in a circular pattern. The descriptors produced by this method are represented by a matrix (the polar matching 'P'-matrix) which is formed with columns which correspond each to a different circularly symmetric sampling pattern. The Discrete Fourier Transform (DFT) of these columns is taken.

To match two keypoints, the element-wise product of the two P-matrices is taken and summed row-wise. The inverse-DFT of the resulting vector produces a vector which gives the correlation of the two keypoints at many (typically 48) evenly-spaced relative rotations. Finding the maximum of this vector tells us both the strength of the correlation and also the relative rotation at which it occurs. For the target matching scheme used here, this is very important.

III. Target Matching

A. Single-Keypoint Target Matching

As a starting point, we could imagine a target which contains a single keypoint at location $\hat{\mathbf{x}}$. That keypoint could be found anywhere on the target, so we define the target as having a centre at location $\mathbf{t}$. This means, given the location $\mathbf{x}$ of the same keypoint on a matching object, the *implied target centre* $\mathbf{t}'$ for that object is given by

$$\mathbf{t}' = \mathbf{x} + \mathrm{R}_\theta \mathbf{d} \tag{2}$$

where $\mathbf{d} = \mathbf{t} - \hat{\mathbf{x}}$ and R_θ is the rotation matrix corresponding to the rotation between target and the matching object.

Using the polar matching method described in [3], both a correlation score and rotation are obtained. Thus, in an image with K keypoints, we could compare every keypoint in the image with the keypoint which describes our target. Using the apparent rotation from each comparison in equation 2, it is then possible to find K implied target centres. These can then be ranked by the correlation score of each keypoint with the target keypoint in order to determine the most likely matches.

While this method will work for simple targets in benign environments, in practical situations there is likely to be a high density of spurious matches.

B. Target Template

Of course, real targets will typically contain many keypoints. To have an effective target matcher, it is necessary to combine the information from these keypoints in such a way that the global geometric information about the target is captured, as well as the local information encoded in each descriptor. To describe a target with multiple keypoints, these keypoints are brought together in a set.

The keypoints (with their associated descriptors) which form this set are defined using a spatial constraint. Now, taking the set of keypoints on the target itself, a single target centre is defined.[1] Thus, every keypoint j is assigned a displacement $\mathbf{d}_j = \mathbf{t} - \mathbf{x}_j$ – that is, the displacement of the target centre from that keypoint, given the correct keypoint orientation.

The set of keypoints with displacement vectors forms the *target template*.

C. Multi-Keypoint Target Matching

The method presented here is merely an extension of that described in section III-A.

If, on a matching object, all the keypoints are detected in the same relative positions as in the original target, and with the same rotations, all the implied target centres for those keypoints will be in exactly the same position. However, for any real image, there are measurement errors, and the predicted centres even for an almost identical matching object will not perfectly align. This means there needs to be some way to cluster together implied target centres which do not perfectly coincide.

If the errors in the positions of each of the implied target centres may be considered identically Gaussian distributed[2] then each implied target centre may be treated as a Gaussian blob, scaled by the correlation score of the keypoints.

By summing all of these Gaussians together, the result is a smooth surface which may be thought of as expressing

[1] In practice this is achieved by the user clicking to select the centre of the target, and then dragging to indicate a radius defining the spatial extent of the target.

[2] This is almost certainly not true, which may be seen if we decompose the error into discretization error and error in the value of θ, the first of which is likely to be Gaussian and the second of which certainly isn't. However, this assumption simplifies the analysis and processing, and gives reasonable results.

the likelihood that a matching object has its centre at any given point. Of course, it is not known in advance which keypoints on a matching object and target correspond. Thus, it is necessary to compare every keypoint in the target template with every keypoint in the image. The value of the matching surface M at a point $\mathbf{w}$ for an image containing I keypoints and target template containing T keypoints is then given by:

$$M(\mathbf{w}) = \frac{1}{2\pi\sigma_{sp}^2} \sum_{i=1}^{I} \sum_{j=1}^{T} c_{ij} \exp\left\{ \frac{|\mathbf{w} - \mathbf{t}'_{ij}|^2}{2\sigma_{sp}^2} \right\} \tag{3}$$

$$\text{where} \qquad \mathbf{t}'_{ij} = \mathbf{x}_i + \mathrm{R}_{\theta_{ij}} \mathbf{d}_j \tag{4}$$

and c_{ij} and θ_{ij} are, respectively, the correlation and rotation obtained by the comparison of keypoints i (in the search image) and j (in the target template). σ_{sp} is the standard deviation of the Gaussian used to represent each implied target centre.

In calculating $M(\mathbf{w})$, the vast majority of keypoint comparisons will not be between keypoints which correspond to the same locality on matching objects. As a result, most of the $I \times T$ implied target centres will not represent anything meaningful. However, there are two reasons why M, which is constructed from these implied target centres does help us find possible locations for the search object:

1) For a matching object, a large number of implied target centres are expected to be found clustered around the corresponding centre of that object. After representing these as Gaussian functions and summing, they will reinforce each other.
2) Keypoint descriptors which do not describe similar areas should have a low correlation score.

Candidate locations (*matches*) within the image may be found as the maxima of M. Though there may be a very large number of maxima, these may be sorted by weight, where the *match weight* is the value of M at that location. In this manner, the matches are sorted by relevance. When looking for matches to a given target, one may start at the top of the sorted list and simply stop looking when the matches become obscure.

D. Match Histogram

In practice, M (see equation 3 on page 3) is not calculated exactly. Instead, it is sufficient and far less costly to compute the *match histogram*. This is a sampled version of M which is computed by accumulating correlation scores at the implied target centres in spacial bins.

This histogram is then smoothed using a discrete Gaussian kernel.[3] For simplicity, a 2D grid the same size as the image is used. This means that, effectively, the positions of implied target centres are quantised to an integer pixel position. Thus, given a target template and a search image, the procedure to produce the match histogram is:

1) Compute correlation score between every keypoint in the target template and every keypoint in the image
2) For each comparison, find the implied target centre in pixel coordinates using the relative rotation and displacement vector
3) Add the correlation score for each comparison to the match histogram at the location of the implied target centre
4) Smooth the match histogram using a Gaussian kernel
5) Find maxima of the smoothed match histogram

An example of a match histogram is given in figure 1.

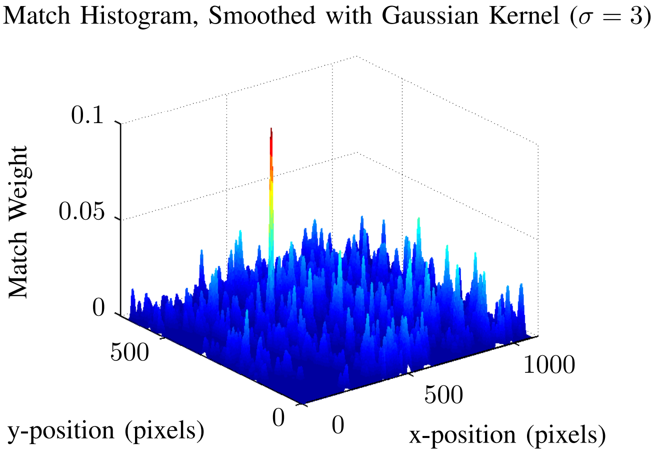

Fig. 1. The match histogram for a small target. The very tall peak corresponds to the original target, the smaller peaks correspond to possible matches

IV. Rotational Coherence

As the system was tested with real imagery, it became apparent that spurious matches were arising which bore no resemblance to the target. Investigation showed that, typically, the detected rotations of the various keypoints contributing to the match were random. This is to be expected – whilst a rotation of the original target (or a rotated "true" match) should result in an equal rotation of all the keypoints, a spurious match generated by coincidental implied target centres would not be expected to have correlated keypoint rotations.

One way that was investigated to use this information to remove the spurious matches was using "rotational coherence". This is a measure which quantifies how similar a set of angles are. Since angle is a circular quantity, an ordinary mean and standard deviation are not suitable to estimate how similar the rotations of the various keypoints contributing to a given peak

[3] The Gaussian kernel is truncated to discard all samples which are less than $\frac{1}{1000}$ of the peak value

are. Instead, a variant of the circular mean given in [7] is employed.

Here, angles are converted to points on the unit circle and the Euclidean mean of these points calculated:

$$\mathbf{x}_n = \begin{bmatrix} \cos\theta_n \\ \sin\theta_n \end{bmatrix} \tag{5}$$

$$\boldsymbol{\alpha} = \frac{1}{N}\sum_{n=1}^{N} \mathbf{x}_n \tag{6}$$

The mean of the angles θ_n is then given as the angle which $\boldsymbol{\alpha}$ makes with the x-axis. The length of the vector $\boldsymbol{\alpha}$ lies in the range $[0, 1]$. It is equal to 1 only if all angles θ_n are the same, and is equal to 0 if they are evenly distributed around the unit circle. As a result, the length of $\boldsymbol{\alpha}$ is a measure of how "concentrated" the angles are, and is referred to as *rotational coherence* herein.

For the purpose of assigning rotational coherence to matches (that is, peaks in the match histogram), the measure of equation 6 is altered to take account of the contribution of the different implied target centres to the match weight. Now, for the match at position $\mathbf{p}$,

$$\hat{\boldsymbol{\alpha}} = \sum_{i=1}^{I}\sum_{j=1}^{T} \phi_{ij}\mathbf{x}_{ij} \tag{7}$$

$$\text{where} \quad \phi_{ij} = \frac{1}{2\pi\sigma_{sp}^2 M(\mathbf{p})} c_{ij} \exp\left\{\frac{|\mathbf{p}-\mathbf{t}'_{ij}|^2}{2\sigma_{sp}^2}\right\} \tag{8}$$

$$\text{and} \quad \mathbf{x}_{ij} = \begin{bmatrix} \cos\theta_{ij} \\ \sin\theta_{ij} \end{bmatrix} \tag{9}$$

Here, ϕ_{ij} represents the contribution of the implied target centre at $\mathbf{t}'_{ij}$ to the weight of the match at location $\mathbf{p}$, i.e. $\sum_{ij}\phi_{ij} = 1$ and $M(\mathbf{p})$ is the matching surface from equation 3. $\hat{\boldsymbol{\alpha}}$ is the weighted circular mean, and $|\hat{\boldsymbol{\alpha}}|$ is the rotational coherence. The average rotation for a given match may also be calculated as[4] $\hat{\theta} = \text{atan2}(\hat{\alpha}_2, \hat{\alpha}_1)$, and this can be used as an estimate of how much a match is rotated relative to the target.

Of course, as in the case of the calculation of the match histogram, this calculation can be performed much more easily than is implied by equation 7 (which would, for every peak in the match histogram require a summation with $I \times T$ terms). In this case, two "layers" are added to the match histogram. As the correlation score for each implied target centre is accumulated in the first layer (as described in section III-D), the quantities $c_{ij}\cos\theta_{ij}$ and $c_{ij}\sin\theta_{ij}$ are accumulated in the other two layers. These layers then have the same Gaussian smoothing applied to them as to the match histogram. If we refer to these two layers as $M_{\cos}(\mathbf{w})$ and $M_{\sin}(\mathbf{w})$, the rotational coherence at the location $\mathbf{p}$ of a peak in the match histogram is then found as

$$|\hat{\boldsymbol{\alpha}}| = \sqrt{\frac{M_{\cos}(\mathbf{p})^2 + M_{\sin}(\mathbf{p})^2}{M(\mathbf{p})^2}} \tag{10}$$

This adds to the computational complexity of the system only slightly.

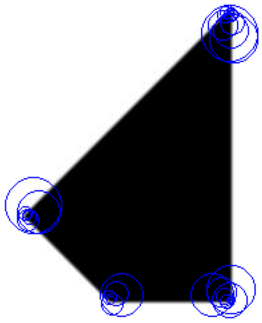

Fig. 2. The "kite" image used to investigate the effect of rotation, as in figure 3. The detected keypoints are marked as blue circles.

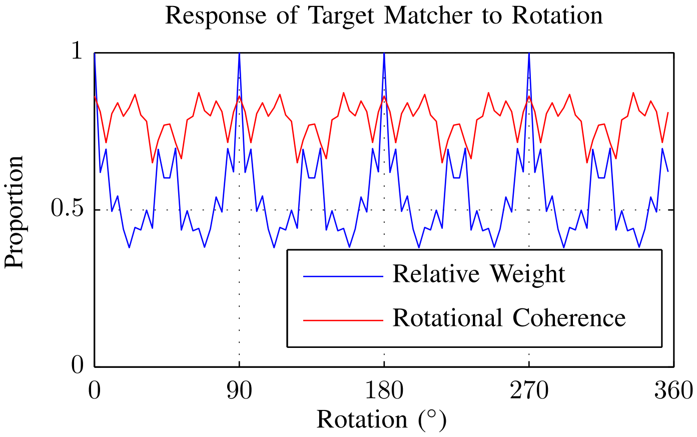

Fig. 3. Variation in match weight with rotation between target and matching object

V. Results

A. Synthetic Imagery

Synthetic imagery was used to investigate the effects of various image transformations on the target matcher's performance. One transformation of interest is that of rotation. It is desirable that the target matcher should not be affected by the orientation of matching objects, but it was found that for this method some degradation of performance did occur.

Figure 3 shows the variation in height of the peak in the match histogram (the *match weight*) of a rotated version of the simple "kite" image shown in figure 2.

For this simple example, the match weight drops by as much as 60% depending on the relative rotation of the target as used to define the target template and the rotated version.

[4] atan2(y, x) is the two argument version of arctan, which is the angle in radians between the positive x-axis and the position vector to the point (x, y).

The effect that this has on the target matcher's performance depends on the "unrotated" weight of the match, as compared with spurious matches which will be present in the image, since it is only the ranking of matches which is important.

Close inspection found that this effect was due to instability in the locations of keypoints. As shown in [3], the keypoint descriptor used is affected very little by keypoint rotation, however, its reliable operation does depend on having an accurate location for the keypoint. It is in this area that the greatest opportunities for improvement in the procedure exist.

B. *Real Imagery*

The target matcher was successfully tested with a range of targets using publicly-available aerial and satellite imagery. For several targets (mostly road vehicles and boats), it was found that where a good match could be identified by eye, this would be identified by the target matcher as the top match. As mentioned, a large number of peaks occur in the match histogram (this can be seen in figure 1), so only the best 5 matches are shown in figure 4. However, it is clear that the correct locations of all the boats in the image have been found within these top 5 matches, though one false positive is present.

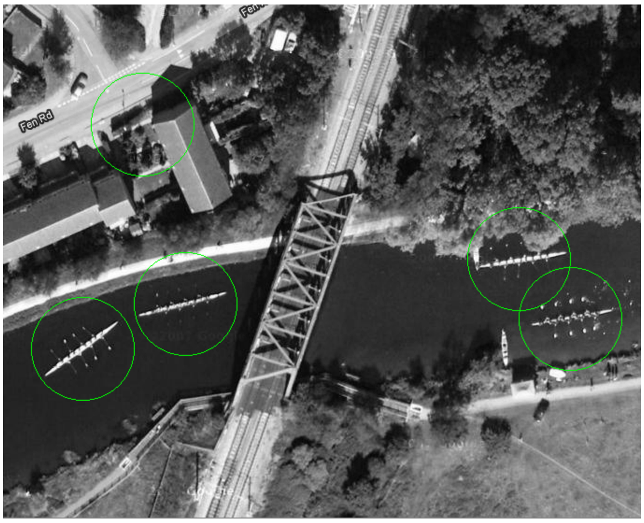

Fig. 4. Rowing eights on the river Cam (imagery from Google Maps). The left-most boat was used to define the target template, the best 5 matches in the image are shown.

VI. Conclusion

The method of the "match histogram" has been shown to be successful, applied as an approach to the target matching problem.

In addition, it has been demonstrated that fairly simple techniques based on the Dual Tree Complex Wavelet Transform may be used to efficiently perform the key intermediate steps of target matching – namely keypoint detection, description and matching. These measures take advantage of some of the DTCWT's important properties: approximate shift invariance and rotational symmetry in particular, to compute the required intermediate results with a minimum of redundancy.

The analysis presented in section V-A in particular indicates that there is scope for considerable improvement, particularly with respect to rotation, if the keypoint detector can be made more robust and reliable in estimating keypoint location and scale. However, this remains a difficult problem.

References

[1] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proc. International Conference on Computer Vision*, Corfu, Greece, Sep. 1999, pp. 1150–1157.

[2] N. G. Kingsbury, "Complex wavelets for shift invariant analysis and filtering of signals," *Journal of Applied and Computational Harmonic Analysis*, vol. 10, no. 3, pp. 234–253, May 2001.

[3] ——, "Rotation-invariant local feature matching with complex wavelets," in *Proc. European Conference on Signal Processing (EUSIPCO)*, Florence, Italy, Sep. 2006, paper 1568982135.

[4] B. T. Pashmina Bendale and N. G. Kingsbury, "Multiscale keypoint analysis based on complex wavelets," in *Proc. British Machine Vision Conference (BMVC)*, Aberystwyth, UK, Sep. 2010.

[5] M. S. C. Harris, "A combined corner and edge detector," in *Proc. 4th Alvey vision conference*, Manchester, UK, Sep. 1988.

[6] P. Bendale, "Development and evaluation of a multiscale keypoint detector based on complex wavelets," Ph.D. dissertation, University of Cambridge, Cambridge, UK, Jan. 2011.

[7] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 2006.