Appearance Based Indexing for Relocalisation in Real-Time Visual SLAM

Denis Chekhlov, Walterio Mayol-Cuevas and Andrew Calway
University of Bristol
{chekhlov,wmayol,andrew}@cs.bris.ac.uk

Abstract

Previous work on visual SLAM has shown that indexing on space and scale facilitates the use of feature descriptors for matching in real-time systems and that this can significantly increase robustness. However, the performance gains necessarily diminish as uncertainty about camera position increases. In this paper we address this issue by introducing a further level of indexing based on appearance, using low order Haar wavelet coefficients. This enables fast look up of descriptors even when the camera is lost, hence allowing efficient relocalisation. Results of experiments on a range of real world test cases demonstrate that the method is effective, including single frame relocalisation rates up to 90% using relatively low numbers of descriptor comparisons.

1 Introduction

Recent years has seen the emergence of real-time vision systems capable of tracking 3-D camera pose whilst simultaneously mapping the surrounding environment. Of particular note are those based on the probabilistic formulations which underlie the simultaneous localisation and mapping (SLAM) techniques used in robotics [4, 7]. These have demonstrated the benefit of harnessing the uncertainty relationships encoded in such formulations for focusing image processing operations when and where required, hence enabling real-time operation. Add to this their natural online processing structure and their ability to maintain covariance relationships across estimated parameters, and it is clear that these systems have the potential to provide effective mechanisms for real-time location sensing.

Nevertheless, achieving robust performance during erratic non-smooth camera motion or in visually difficult environments remains a challenge for such systems. A key element is the data association, or feature matching, problem. If uncertainty is low, then image search regions derived from a probabilistic filter will be small, constraining the spatial search for matches and hence reducing computation and likelihood of mismatch. This in turn allows the use of weaker matching techniques, e.g. template matching, in order to further reduce computational load. Of course, it also runs the risk of losing track should uncertainty increase - search regions grow and the probability of mismatch increases, resulting in bad data association and filter instability.

An effective way of gaining improved robustness is to base matching on more distinctive descriptors, such as those developed in recent years for object recognition [6, 10]. This is the approach adopted by Chekhlov *et al*. [5], who utilise the spatial gradient descriptors which form the basis of the Scale-Invariant Feature Transform (SIFT) [6]. They

BMVC 2008 doi:10.5244/C.22.37

combine this with indexing on space and scale, gating image regions and descriptor scales using the pose and uncertainty estimates from the filter. This achieves efficient matching and permits real-time operation. The more distinctive matching properties of the descriptors provides greater robustness during short periods of uncertainty caused by camera shake, for example. However, the approach is necessarily limited; if uncertainty continues to increase, as is the case when the camera is lost, for example, then the search over space and scale increases to the extent that real-time operation is no longer possible.

We address this limitation by introducing an additional level of indexing based on the appearance of image patches associated with features. For this we use low order Haar wavelet coefficients in a similar manner to Brown *et al*. [14]. These correspond to coarse estimates of spatial gradients and provide a fast means of categorising features prior to full matching with descriptors. This enables efficient matching even when uncertainty in pose becomes large. Importantly, when combined with RANSAC outlier rejection, it facilitates rapid relocalisation of the camera. The result is a robust system, capable not only of tolerating short-lived uncertainty in camera pose, but also of recovering from tracking failure caused by sustained motion blur or occlusion, for example.

Following a brief review of related work, Section 2 gives an overview of visual SLAM, and in particular the descriptor matching technique used in [5]. Section 3 describes the appearance indexing method and results of experiments on real world test cases are presented in Section 4. These demonstrate that the approach is effective, achieving single frame relocalisation rates of up to 90%, whilst using below 5% of the descriptor comparisons required for exhaustive search matching.

1.1 Related Work

Se *et al*. [16] and Gordon and Lowe [11] use the SIFT, and hence the same form of descriptors, for feature matching when localising a camera within a pre-built map. They also use RANSAC outlier rejection when computing the camera pose in each frame. In these respects the methods are similar to that described here when in relocalisation mode, i.e. a known map and a lost camera. However, the SIFT is significantly more demanding in terms of computation (frame rates of 4 fps are reported in [11] for localisation alone), and its use is unnecessary during normal operation in SLAM when using scale indexing [5]. In this work we show that it is also unnecessary for relocalisation, and that this can be achieved efficiently using appearance indexing.

Williams *et al*. [2] have also recently developed a method for relocalisation in visual SLAM and report impressive results. They use template matching during normal operation and then switch to a fast version of the randomised trees classifier [13] for matching features during relocalisation. This classification approach contrasts with the descriptor matching strategy used here. Our results presented in Section 4 suggest that the two methods are similar in terms of performance and that their relative merits are also comparable. On the one hand, classification via randomised trees is almost certainly faster than descriptor comparisons, but on the other hand requires significantly greater memory resources. Also, the increased robustness provided by descriptors means that our method is less sensitive to relocalisation delays, i.e. it is able to tolerate greater uncertainty in the relocalised pose. Our results also suggest that the use of descriptors gives greater levels of precision in feature matching prior to using RANSAC, at around 50%-65% compared with the 20% reported in [2], giving faster convergence to consensus.

2 Visual SLAM Using Descriptors

In this section we provide a brief overview of visual SLAM and the specific case of using feature descriptors for matching as described in [5]. Readers are referred to the now extensive literature on such systems for more detailed information, see e.g. [1].

Our objective is to determine the 3-D pose of a moving camera whilst at the same time mapping the surrounding environment based on observations within the video stream. We represent the camera pose, $\mathbf{v} = (\mathbf{q}, \mathbf{t})$, in terms of its orientation, defined by the quaternion $\mathbf{q}$, and its position vector $\mathbf{t}$. Scene structure is defined by a map of N features, $(F_1, F_2, \ldots, F_N)$, which we assume here to be points in the scene with 3-D positions $\mathbf{m} = (\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_N)$. This gives a system state vector $\mathbf{x} = (\mathbf{v}, \mathbf{m})$ with dimension $7 + 3N$. In a probabilistic formulation, given observations $\mathcal{O}$, we seek to determine the posterior density $p(\mathbf{x}|\mathcal{O})$. Assuming Gaussian statistics and Markov state evolution, then mean and covariance estimates can be obtained using the Kalman filter (KF) and its variants [17]. This requires the definition of a process model and an observation model. The former defines the state evolution through time, i.e. $\mathbf{x}^{new} = \mathbf{f}(\mathbf{x}, \mathbf{e})$, where $\mathbf{e}$ is a zero mean Gaussian vector representing our uncertainty about the camera motion. Options for $\mathbf{f}$ include constant velocity or constant position models. We use the latter in this work. The observation model defines the relationship between the system state and observations in the current frame. We assume that each frame yields M observations, $(O_1, O_2, \ldots, O_M)$, and in the case of point maps, these will be defined by 2-D points $(\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_M)$, each potentially corresponding to the perspective projection of a 3-D map point. Denoting the position of one such point w.r.t the camera by $\mathbf{y}(\mathbf{v}, \mathbf{m}_n)$, this gives a set of observation models of the form

$$\mathbf{z}_{j_n} = \Pi(\mathbf{y}(\mathbf{v}, \mathbf{m}_n)) + \mathbf{w}_n \tag{1}$$

where the indices $(j_1, j_2, \ldots, j_N)$ define the correspondence between the map features and a subset of observations, Π denotes pin hole projection for a calibrated camera and $\mathbf{w}_n$ is a zero mean Gaussian vector representing the uncertainty in the observation. Both these models and the process model are non-linear and hence we obtain sub-optimal estimates of the state mean and covariance using the extended KF (EKF) [17]. The filter is initialised using a calibrated pattern of points in the scene and map points are initialised using the inverse depth formulation described in [15].

2.1 Data Association

A critical issue in all SLAM systems, and one that is central to this work, is the data association problem, i.e. determining the indices j_n in eqn (1). Poor assignment of observations to features leads to inconsistent estimation and ultimately filter instability. Ideally we would wish to optimise assignments over all frames, but online operation requires that we fix them for each iteration of the filter. The common approach to this is to adopt a nearest neighbour strategy such that for each map feature F_n, we assign the closest observation according to a distance metric, i.e. set j_n to the j that minimises $\mathrm{DIST}(F_n, O_j)$. For example, many existing systems base this on template matching [1].

A more robust approach is to base data association on more distinctive patch descriptors, as described by Chekhlov *et al*. [5]. Their method makes use of multiresolution descriptors and can be formally defined as follows. Given an image point $\mathbf{u}$, we define a function $\mathrm{PATCH}_{a\times a}(\mathrm{I}, \mathbf{u})$, which extracts from the frame I an image patch of size $a \times a$

pixels about the point $\mathbf{u}$, i.e. $\text{PATCH}_{a\times a} : (\ \text{I}, \mathbf{u}) \mapsto\ \text{P} \in \mathbb{R}^{a\times a}$. We define a further function $\text{SCALE}_{a\times a} : \mathbb{R}^{b\times b} \mapsto \mathbb{R}^{a\times a}$, which takes an input patch of size $b \times b$ and converts it to one of size $a \times a$ using subsampling or upsampling methods as appropriate. Finally, based on the pixel values within a patch P of size $a \times a$, we build an r element descriptor $\mathbf{d} = \text{DESC}(\ \text{P})$, where $\text{DESC} :\ \text{P} \in \mathbb{R}^{a\times a} \mapsto \mathbb{R}^r$. In [5], this is based on the distribution of spatial gradients within sub-blocks as in the descriptors used for the SIFT [6]. These have proved to have good invariance properties w.r.t affine transformations of the patch [12].

The method then proceeds as follows. At the initialisation of a new map feature F_n in frame I_0, say, with associated image point $\mathbf{u}_n$, a set of L multiresolution descriptors $\mathbf{d}_n = (\mathbf{d}_{n1}, \mathbf{d}_{n2}, \ldots, \mathbf{d}_{nL})$ are built as follows

$$\mathbf{d}_{nl} = \text{DESC}(\text{SCALE}_{a\times a}(\text{PATCH}_{s_{nl}.a\times s_{nl}.a}(\ \text{I}_0, \mathbf{u}_n))) \tag{2}$$

where $(s_{n1}, s_{n2}, \ldots, s_{nL})$ denotes the set scales for the representation. The generation of descriptors at multiple resolutions enables subsequent matching to take account of changes in scale, avoiding the computational overhead of determining local maxima in scale space as is done in the SIFT [6]. Given a set of observations in a subsequent frame I_k, the data association problem for the map feature F_n then becomes

$$j_n = \arg\min_j \left[\min_l \ \rho(\mathbf{d}_{nl}, \mathbf{d}'_j) \right] \tag{3}$$

where $\mathbf{d}'_j = \text{DESC}(\text{PATCH}_{a\times a}(\ \text{I}_k, \mathbf{z}_j))$ is the descriptor for the jth observation computed within a patch of size $a \times a$ about the point $\mathbf{z}_j$ and $\rho(\mathbf{d}_1, \mathbf{d}_2)$ is a distance metric between descriptors, e.g. normalised Euclidean distance [6]. As an additional guard against mismatch, we also require that $\rho(\mathbf{d}_{nl}, \mathbf{d}'_{j_n})$ is below a suitable threshold T_d. Thus, amongst the observations, we seek the one whose descriptor is closest to one of those within the multiresolution set associated with the map feature. In practice, it is necessary to normalise descriptors w.r.t the dominant orientation within a patch and also to allow for multiple descriptors per patch as described in [6].

2.2 Space and Scale Indexing

The above formulation of the data association problem takes no account of the fact that given we have knowledge of the camera pose, we can constrain the possible observations to associate with a given map feature. In the extreme case, if a feature projects outside of the image frame given a confident estimate of the pose, then we may conclude that we have no observation for that feature in the current frame. More generally, using predicted mean observations and their associated covariances provided by the filter, we can gate, or index, observations in terms of spatial position: for a given feature, we need only consider matching to those observations within the region about its predicted mean projection as defined by the predicted covariance, as illustrated in Fig. 1a. This spatial indexing reduces both the search space, over j in eqn (3), and the number of image processing operations, and is central to the emergence of real-time visual SLAM systems over recent years.

An additional indexing strategy is used by Chekhlov *et al.* to also enable efficient search over scale in eqn (3). Given knowledge of the camera pose, an estimate of the change in scale that has occurred since a feature was first observed can be obtained. This is based on the estimated change in distance between the camera and the feature point in

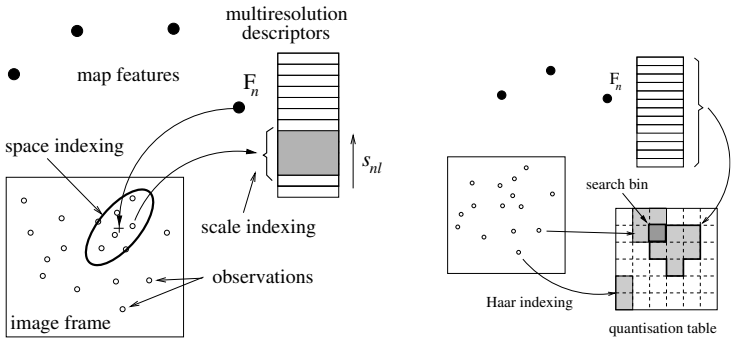

Figure 1: Indexing for efficient descriptor matching: (a) space and scale indexing during normal operation; (b) appearance indexing during relocalisation.

the map [5], with the uncertainty in both pose and the 3-D position of the feature defining a search window about the predicted scale as shown in Fig. 1a. It is this guided search strategy which enables use of the multiresolution descriptors within a real-time system. Data association with both space and scale indexing can thus be formulated as follows. Let Λ_n denote the subset of observations within the current frame whose 2-D positions are within the 'covariance bounding box' about the mean projection of feature F_n and let Γ_n denote the range of spatial scales within which feature F_n is expected to be observed. Efficient data association can then be achieved using

$$j_n = \arg\min_{j \in \Lambda_n} \left[\min_{l \in \Gamma_n} \rho(\mathbf{d}_{nl}, \mathbf{d}'_j) \right] \tag{4}$$

where again we further reduce the chance of mismatch by requiring $\rho(\mathbf{d}_{nl}, \mathbf{d}'_{j_n}) < T_d$. This formulation gives reduced search, and hence reduced image processing operations, as the certainty in camera pose increases.

3 Appearance Indexing and Relocalisation

The difficulty with the data association in eqn (4) is its reliance on good pose estimates. As uncertainty increases, then the search over space and scale increases. In systems using weak matching techniques, this increases the likelihood of mismatch and hence bad data association. For descriptor based matching, mismatches are considerably less likely, but the increased search results in significant reductions in frame rate. In the case of tracking failure, caused by sustained motion blur or visual occlusion, for example, then the time required to achieve good data association is likely to exceed video frame rate significantly, hence prohibiting relocalisation.

We address this problem by introducing a further level of indexing based on the appearance of image patches associated with map features and observations. This enables efficient searching on descriptors when pose information is not available or is unreliable, hence facilitating fast relocalisation. The indexing is based on low order Haar wavelet coefficients [3] and is motivated by work of Brown *et al*. [14]. These correspond to coarse estimates of the spatial gradients in an image patch. For a patch P of size $a \times a$, the first

4 Haar coefficients can be computed using $\mathrm{H}_4 = \mathrm{H}\,\mathrm{P}\,\mathrm{H}^T$ such that

$$\mathrm{H}_4 = \begin{bmatrix} h_0 & h_1 \\ h_2 & h_3 \end{bmatrix} \quad \mathrm{H} = \frac{1}{\sqrt{a}} \begin{bmatrix} 1 & 1 & \cdots & 1 & 1 & \cdots & 1 & 1 \\ 1 & 1 & \cdots & 1 & -1 & \cdots & -1 & -1 \end{bmatrix} \tag{5}$$

where the matrix H is of size $2 \times a$. For indexing we only use the 3 coefficients $\mathbf{h} = (h_1, h_2, h_3)$; the coefficient h_0 being unsuitable for discrimination since it corresponds to the scaled mean of the patch. To account for the lack of rotation invariance, we also need to normalise patches w.r.t their dominant orientation prior to computing the Haar coefficients in a similar manner to that used when generator descriptors.

Indexing is based on a quantisation table $Q = \{b_i\}$ consisting of a set of bins b_i. We define a quantisation function $\mathrm{QUANT} : \mathbb{R}^3 \mapsto B \subset Q$, which produces a mapping between the Haar index $\mathbf{h}$ and a subset of bins B. For a mapped feature F_n, we assign pairs (n,l) to bins according to the quantisation of the Haar coefficients for each patch at scale s_{nl}, i.e. if $\mathbf{h}_{nl}$ denotes the Haar coefficients for the patch at scale s_{nl}, then $b_i = \{(n,l) | b_i \in \mathrm{QUANT}(\mathbf{h}_{nl})\}$. Data association is then achieved by only considering those observations whose appearance indexing maps to a bin associated with F_n

$$j_n = \arg\min_j \left[\min_{l,(n,l) \in b_i, b_i \in B_j} \rho(\mathbf{d}_{nl}, \mathbf{d}'_j) \right] \tag{6}$$

where $B_j = \mathrm{QUANT}(\mathbf{h}_j)$ and $\mathbf{h}_j$ are the Haar coefficients for the patch surrounding the jth observation located at point $\mathbf{z}_j$.

In practice, we switch to the above data association in the event of tracking failure, as indicated by sustained inability to associate observations with mapped features via eqn (4). The resulting matched observations are then used to relocalise. However, the lack of pose information does mean that matching is less reliable, especially when dealing with large maps. Consequently, there are likely to be significant numbers of outliers (in the experiments we recorded levels of around 35%-50%) and thus further processing is required for reliable relocalisation. For this we use RANSAC in a similar manner to that in [16, 11, 2], based on the original formulation by Fischler and Bolles [9]. Briefly, for each match, we have a mean 3-D position for the map feature. Given three such matches, we can obtain a pose estimate [9]. Thus, using RANSAC, we seek a consensus set and the associated pose by testing multiple hypotheses using the 3 point algorithm. For relocalisation, we take the same approach as Williams *et al*. [2], reinitialising the filter with the pose returned by RANSAC and a large artificial covariance, and using the consensus set as the observations for the next update. On successful data association in subsequent frames the filter then returns to normal operation.

4 Experimental Results

We analysed performance by running the system live in the laboratory and offline on several real world test sequences. The latter were captured in an office environment using smooth 'trackable' motion, enabling us to compare frame by frame relocalisation performance with a ground-truth provided by normal SLAM running alongside. Two of the sequences were captured using simple translational or rotational motion, whilst the others were captured using general motion in order to build a reasonably sized feature map. Map

Sequence	Method	Frames			Comps, %
		Success, %	Wrong, %	No Result, %	Mean± Std Dev
Wall Trans	Ext	97.27	0.16	2.56	100±0
	Ind	96.70	0	3.2967	3.14±0.32
Wall Rot	Ext	95.56	0.42	4.00	100±0
	Ind	93.34	0.03	6.61	2.58±0.24
Office 1	Ext	94.66	1.54	3.78	100±0
	Ind	91.53	0.89	7.57	4.11±0.54
Office 2 A	Ext	70.96	2.24	26.79	100±0
	Ind	59.84	1.03	39.12	2.66±0.25
Office 2 B	Ext	89.77	1.00	9.18	100±0
	Ind	74.02	0.40	25.56	3.24±0.33
Office 2 C	Ext	90.59	1.09	8.30	100±0
	Ind	78.96	0.69	20.33	2.86±0.19

Table 1: Relocalisation success rates for exhaustive and appearance indexing search. The right-hand column shows the average percentage ratio of descriptor comparisons made per frame using indexing to that made using exhaustive search.

sizes were approximately 20-25 features for the former and between 40-70 for the latter. The sequences consisted of between 2000 to 2500 frames. We used a calibrated hand-held camera with 320×240 pixels and both narrow-angled (43° FOV) and wide-angled (81° degrees FOV) lens. Observations were obtained using a fast saliency operator [8], giving on average 300-400 detected observations per frame for 5 of the sequences and around 700 per frame for the sequence 'Office 2C'. Descriptors and Haar coefficients were computed within regions of size 23×23 pixels around an observation point.

Results of frame by frame relocalisation are shown in Table 1. This compares the performance for two cases: using exhaustive search to match descriptors as in eqn (3); and using appearance indexing as in eqn (6). Note the high rate of successful relocalisation and the closeness of the indexing performance to that of exhaustive search. The last column shows the average percentage ratio of descriptor comparisons made per frame using indexing to that made using exhaustive search. For indexing this is consistently below 5% and since the additional computational overhead of computing Haar coefficients for indexing is relatively low, this demonstrates that we can significantly reduce computational cost whilst maintaining comparable performance. This is further illustrated in Fig. 2a, which shows the average percentage ratios per frame for two of the sequences. The plot in Fig. 2b shows the average percentage ratio of descriptors created per frame using indexing to that created using exhaustive search. Note the initial increase in the number of descriptors created as the map is built and that once the map has stabilised the rate at which descriptors are built is similar for both indexing and exhaustive search. This indicates that indexing gains us efficiency primarily through reducing the search space, rather than avoiding the creation of descriptors.

As further evidence of the effectiveness of the descriptors for relocalisation, Table 2 shows precision and recall rates achieved prior to using RANSAC for both indexing and exhaustive search. These results also illustrate an additional benefit of indexing - by providing a further level of coarse classification, it reduces the level of mismatch as indicated by the significant increase in precision when using indexing compared to that

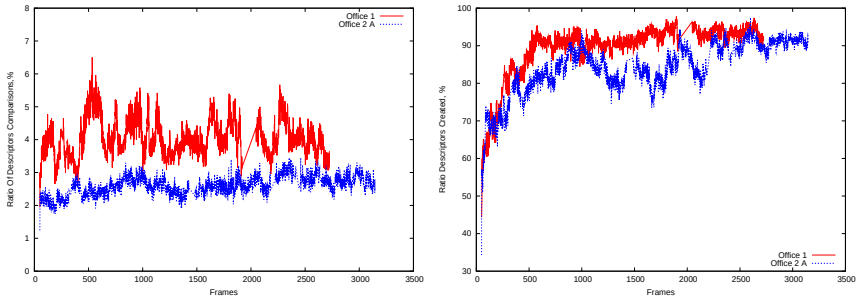

Figure 2: Average percentage ratio of descriptors compared (left) and created (right) per frame using indexing to that compared (created) using exhaustive search.

Sequence	Indexing		Exhaustive	
	Precision, %	Recall, %	Precision, %	Recall, %
Wall Trans	63 ± 8	80 ± 17	61 ± 9	95 ± 9
Wall Rot	67 ± 11	84 ± 17	53 ± 10	87 ± 12
Office 1	51 ± 13	80 ± 18	39 ± 14	85 ± 14
Office 2 A	66 ± 18	76 ± 20	35 ± 15	87 ± 15
Office 2 B	60 ± 18	70 ± 19	40 ± 15	89 ± 15
Office 2 C	70 ± 18	75 ± 22	39 ± 18	94 ± 9

Table 2: Precision and recall rates prior to using RANSAC for descriptor matching using indexing and exhaustive search.

for exhaustive search. Moreover, it does this with only a small reduction in recall rates. These results contrast with the low levels of precision (20%) reported by Williams *et al.* for their system based on randomised trees [2]. This will have the effect of giving quicker convergence to consensus within RANSAC when computing the relocalised pose.

Examples of relocalisation performance achieved during live tests are shown in Fig. 3. This shows the camera view indicating spatial search regions, matched (green) and unmatched (red) features and feature searching for relocalisation (orange). Also shown is the external view of the camera pose and map estimates, with associated uncertainty. In the top example, after the map has been built, the camera is suddenly occluded and moved to a different part of the map. This causes the method to switch to relocalisation mode as indicated in the middle frames. Note the increase in pose uncertainty. After several frames of motion blur the system is able to quickly relocalise and resume normal tracking operation (right-hand frame). The second example shows even more impressive relocalisation capability, with the camera initially rotated by 180° and then suddenly rotated back at the same time as moving to a different part of the map. Again successful and fast relocalisation is achieved.

5 Conclusions

We have presented a new approach to achieving relocalisation and hence greater robustness in real-time visual SLAM. The key contribution is the introduction of appearance indexing alongside indexing on space and scale, which allows efficient use of highly

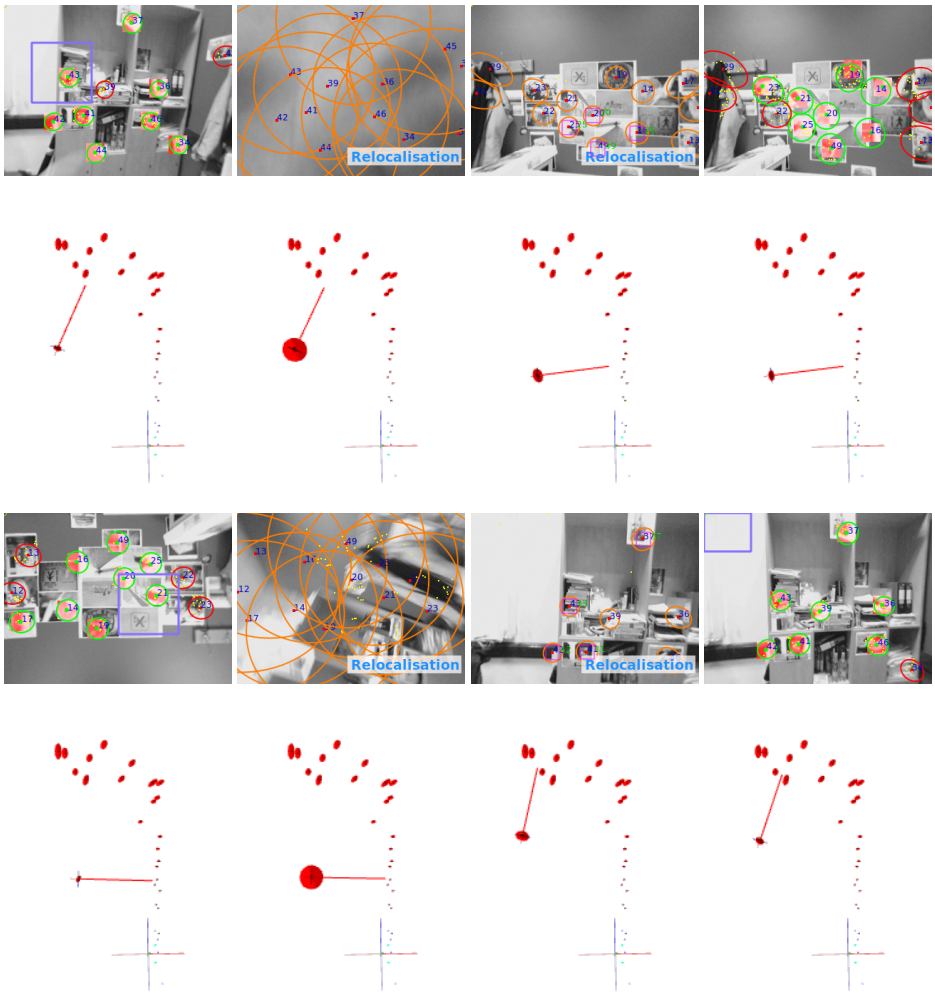

Figure 3: Examples of successful relocalisation during live tests in the laboratory, showing the views through the camera with matched and unmatched features and the external views of the estimated camera pose and map with associated uncertainties.

discriminatory descriptors and hence fast relocalisation. Initial results obtained on test sequences and during live runs in the laboratory suggest that the method is effective and has significant potential. It compares well with the other relocalisation method recently reported in [2]. In future work we intend to further develop the use of indexing strategies, particularly in terms of achieving robust relocalisation within very large maps and under severe changes in viewing perspective.

Acknowledgements: This work was funded in part by ORSAS UK. We are also grateful to Andrew P. Gee for discussions and contributions to software development.

References

[1] A.J.Davison, I.D.Reid, N.D.Molton, and O Stasse. Monoslam: Real-time single camera slam. *IEEE Trans on Pattern Analysis and Machine Intelligence*, 29(6):1052–1067, 2007.

[2] B.Williams, G.Klein, and I.Reid. Real-time slam relocalisation. In *Proc Int. Conf. Computer Vision*, 2007.

[3] C.K.Chui. *An Introduction to Wavelets*. Academic Press, 1992.

[4] A.J. Davison. Real-time simultaneous localisation and mapping with a single camera. In *Proc Int. Conf. Computer Vision*, 2003.

[5] D.Chekhlov, M.Pupilli, W.Mayol-Cuevas, and A.Calway. Real-time and robust monocular slam using predictive multi-resolution descriptors. In *Proc Int. Symp. on Visual Computing*, 2006.

[6] D.Lowe. Distinctive image features from scale-invariant keypoints. *Int Journal of Computer Vision*, 60(2):91–110, 2004.

[7] E.Eade and T.Drummond. Scalable monocular slam. In *Proc. IEEE Int Conf on Comp Vision and Patt Recog*, 2006.

[8] E.Rosten and T.Drummond. Machine learning for high-speed corner detection. In *Proc. European Conf on Computer Vision*, 2006.

[9] M.A. Fischler and R.C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[10] H.Bay, T.Tuytelaars, and L.Van Gool. Surf: Speeded up robust features. In *Proc. European Conference on Computer Vision*, 2006.

[11] I.Gordon and D.Lowe. Scene modelling, recognition and tracking with invariant image features. In *Proc. Int Symp on Mixed and Augmented Reality*, 2006.

[12] K.Mikolajczyk and C.Schmid. A performance evaluation of local descriptors. *IEEE Trans on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, 2005.

[13] V. Lepetit and P. Fua. Keypoint recognition using randomized trees. *IEEE Trans on Pattern Analysis and Machine Intelligence*, 28(9):1465–1479, 2006.

[14] M.Brown, R.Szeliski, and S.Winder. Multi-image matching using multi-scale oriented patches. In *Proc. IEEE Int Conf on Comp Vision and Patt Recog*, 2005.

[15] J.M.M. Montiel, J.Civera, and A.J.Davison. Unified inverse depth parametrization for monocular slam. In *Proc Robotics: Science and Systems Conf.*, 2006.

[16] S.Se, D.Lowe, and J.Little. Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks. *Int Journal of Robotics Research*, 21(8):735–758, 2002.

[17] Y.Bar-Shalom, T.Kirubarajan, and X.-Rong Li. *Estimation with Applications to Tracking and Navigation*. Wiley, 2002.