Place recognition from disparate views

Rob Frampton
http://www.cs.bris.ac.uk/~frampton
Andrew Calway
http://www.cs.bris.ac.uk/~andrew

Visual Information Laboratory
University of Bristol

Abstract

Visual place recognition methods which use image matching techniques have shown success in recent years, however their reliance on local features restricts their use to images which are visually similar and which overlap in viewpoint. We suggest that a semantic approach to the problem would provide a more meaningful relationship between views of a place and so allow recognition when views are disparate and database coverage is sparse. As initial work towards this goal we present a system which uses detected objects as the basic feature and demonstrate promising ability to recognise places from arbitrary viewpoints. We build a 2D place model of object positions and extract features which characterise a pair of models. We then use distributions learned from training examples to compute the probability that the pair depict the same place and also an estimate of the relative pose of the cameras. Results on a dataset of 40 urban locations show good recognition performance and pose estimation, even for highly disparate views.

1 Introduction

Place recognition is the ability of a system to identify its current location with respect to a set of previously visited places. Vision-based place recognition techniques are becoming popular due to the cheapness and flexibility of cameras. For vision systems, this involves determining, given an input image, which other image in a database is most likely to depict the same physical location in the world.

Consider the top row of images in Figure 1. It is clear that these two images show the same place, albeit with a small change in viewpoint. Existing work in this area has largely focussed on situations such as this, in which there is a large amount of overlap between the views. These systems usually rely on image retrieval-type techniques for comparing places, such as the extraction of local or global image features, with approaches based on the bag-of-words representation [25] having shown to be particularly successful. Notably, FAB-MAP [6] has demonstrated the ability to work on datasets of over 100,000 images in real-time.

Now consider the bottom row of Figure 1. Since the images appear to be quite different, it is not immediately obvious that they depict the same location. However, humans are able to use their understanding of the world to look for more subtle clues such as the style of the area, the objects in the scene, features on the ground, and the geometric arrangement of all of these. Thus, most people should be able to say that the two images are consistent with each other, and can there be fairly confident that they represent the same place - even though there has been a significant change in viewpoint.

Figure 1: Pairs of input images and the output from our system. Top row: images of a similar viewpoint; Bottom row: images of disparate viewpoints.

Although humans are capable of recognising places in this way, existing image matching techniques make assumptions about the visual similarity of separate observations of the same place, making them unsuitable for this type of recognition. The reliance on finding common local features between the images can make them susceptible to occlusion, lighting conditions and environmental changes, and most significantly, means that their viewpoints must have substantial overlap. This may not be a problem if the place database already has thorough coverage of the area in question, but this is not always practical. We also may not be able to assume that when we revisit a place we will see it from a similar viewpoint as before.

Suppose that rather than simply representing a place by the image features which describe its appearance, we used a higher level representation of the scene. If we were to model a scene using some understanding of semantic features and their geometric arrangement, we could recognise places from an arbitrary viewpoint, as long as enough information is visible - closer to the way in which humans appear to interpret a scene. In this paper we take initial steps towards building such a system, and present results of a place recognition method which uses detected objects. By estimating the geometry of the scene and extracting features we can determine the probability that two images depict the same place. We also gain some semantic understanding of the relationship between the two views by estimating the position of each object, the ground plane and the relative pose of the cameras, shown in the final column of Figure 1. Although our method is restricted to scenes which contain familiar objects, we are not aware of any existing work which performs recognition from disparate views such as this, so we aim to demonstrate that there is potential for systems which can recognise places from images with little or no visual overlap.

2 Related Work

Although there has been some successful place recognition work using global features [27], the majority of recent work has involved the extraction and matching of local image features such as SIFT [13] or SURF [3]. The bag-of-words model proposed by Sivic and Zisserman [25] has been particularly popular [1, 6, 19, 24], and recent work has demonstrated the real-time recognition of very large databases [7, 17] and systems combined with odometry to represent topological relationships between images [4, 16]. Some works have also used features associated with 3D information from stereo [5] or laser range data [21] to

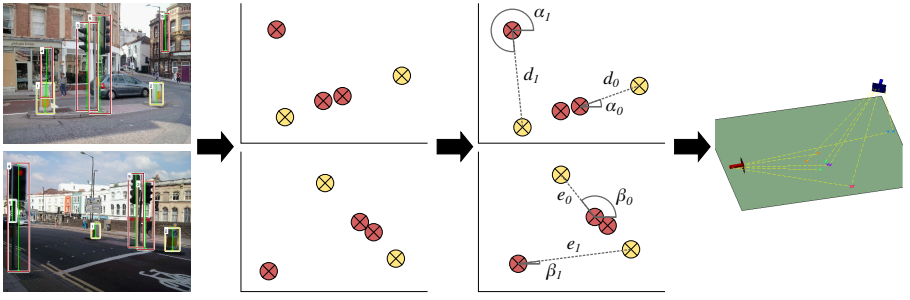


Figure 2: Overview of the primary stages of our system

improve recognition; nonetheless, these still suffer from the limitations of image-matching approaches.

There is also a growing body of work exploring the interpretation of semantic features in images and their geometry. This includes semantic segmentation [9], depth estimation from a single image [13, 23], generation of a "blocks world" model through physical reasoning [10], estimating planar structure [11, 12], and modelling 3D scenes using familiar objects [26]. Bao and Savarese [2] present a structure from motion system which, like us, uses the notion of geometric consistency of detected objects across two views. However they also rely on local image features which means disparate viewpoints would not work. We take inspiration from Hoiem et al [14] as we use a similar model of object geometry, but their goal is to improve object detection.

Perhaps the most similar work to ours is that of Ranganathan and Dellaert [22], who use objects as a feature in place recognition, but as they also model the positions of local features in the objects their system does not appear to have any ability to recognise places from arbitrary viewpoints. They also only demonstrate results over 6 indoor locations.

3 Overview

We have chosen to use objects as the basic feature in our system. They were chosen as a starting point for a number of reasons: they have clear, distinct classes (e.g. “car”, “traffic light”) which eases correspondence across widely separated views; we are able to make some inference about the (relative) size of known objects, and therefore infer their depth; and for our purposes, most simple objects do not extend arbitrarily, and thus can be treated as point features in 3D space.

The overall approach we take is to use knowledge about the objects in the scene to generate a geometric place model, then extract features from a pair of place models, and compute the probability that the places are the same. Figure 2 illustrates the main stages of our system. The remainder of this section will briefly overview our method, with the following sections providing more detail.

The first column in Figure 2 shows a pair of typical input images. We begin by detecting objects in an image using a small set of pre-trained object classifiers. We used the detector implemented by Felzenszwalb *et al*. [8] and classifiers for five classes of common street object were learned: ‘traffic light”, “round sign”, “bollard”, “small bollard” and “belisha beacon”.

By assuming that we know the relative world height of each object class, we estimate the depths of the objects as well as the plane they sit on. Ultimately we generate a model like

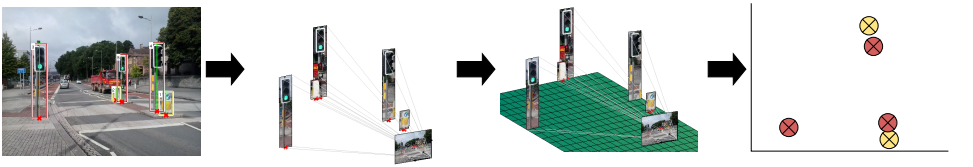


Figure 3: Illustration of the geometry estimation

that shown in the second column of the figure - essentially a top-down 2-dimensional view of the scene. This is the place model which we use to perform place recognition.

If the place model is a good approximation of the scene geometry, it should be the case that when we are given two images of the same place, their models look very similar; indeed, if we had perfect measurements, they would be related by just a rotation and translation. With this in mind, we extract a set of features $\mathcal{F}$ from the place models and compute the probability $p(C|\mathcal{F})$, where C is the event that the two images depict the same place. The problem is now treated as a machine learning problem; distributions $p(C|\mathcal{F}_i)$ for each feature are estimated from training data and are used to compute the final probability during testing.

A significant issue to be addressed is the fact that we do not know the correspondence of objects between images. The problem is compounded by the fact that the set of detected objects often contains a significant amount of noise - both false positive detections and true positive detections of objects which are only visible in one view. Indeed, the number of objects which are common to both views may be smaller than the number of uncommon or incorrect detections, so existing point correspondence algorithms are not applicable. In addition, the number of possible correspondences grows very large with the number of detected objects, meaning that excess noise can generate an intractable number of hypotheses.

We take a relatively simple approach to the correspondence problem. First, we only consider 5-object correspondences between views. This necessitates the presence of five objects which are common to both images, but helps robustness to noisy detections. Second, we only consider the top 10 ranking objects, according to the confidence scores generated by the object detector, in each image. We find that this limits the number of correspondence hypotheses to a tractable figure, whilst still generally retaining 5 common objects between views. We will evaluate $p(C|\mathcal{F})$ for all possible 5-object correspondences and use the highest hypothesis probability as the probability that the images depict the same place.

4 Geometry Estimation

We use a geometry model inspired in part by the work of Hoiem *et al*. [14], which assumes the relative world height of each object class is known. We also assume that we know the height of the camera from the ground in the same units - like [14], we estimate this *a priori* by observing that all of our images are taken at eye level.

Given that for each object i we have the world height H_i, image height h_i in pixels, and the focal length of the camera f in pixels, we can use similar triangles to estimate the depth of the object $d_i = \frac{fH_i}{h_i}$. The result is shown in the second column of Figure 3.

We now approximate the point in the image at which each object contacts the ground by taking the point at the base of each bounding box, shown in red in Figure 3. Since we have an estimate of the depth of each object, these ground points yield a set of 3D points which represent the base (i.e. ground contact point) of each object, in the camera frame.

Ideally, we would like to estimate the ground plane by fitting a plane to these points, however some of the points may be erroneous detections. Alternatively, we could estimate a

different ground plane for each set of 5 objects, which would be consistent with our decision to use 5-object correspondence hypotheses when comparing places, but having different object geometry for each hypothesis would prevent us from precomputing features as described in the following section. Instead, we compute the ground plane for all possible sets of 5 objects, and take the average of all planes whose residual is below a threshold. We find that a fixed threshold determined experimentally is sufficient. The third column of Figure 3 shows the ground plane estimate. It is more convenient for our purposes to orient the ground plane flat along the X-Z plane, so we instead treat the camera as being rotated relative to the ground, rather than the ground being sloped relative to the camera.

Finally, we re-estimate the position of the objects by intersecting the ground ray of each object with the X-Z (ground) plane. We can now represent each object as a 2D point on the plane, essentially giving a top-down view of the scene, as in the final column of Figure 3. This is the place model which we use for comparing places in the following section.

5 Hypothesis scoring

Given a pair of images, we can compute a geometric place model for each as described above. We can also compute the full set of 5-object correspondence hypotheses between the views. We now wish to compute the probability of each hypothesis being correct.

Intuitively, if the models do represent the same place, we would expect them to be related by just a translation and rotation. In reality there is obviously a reasonable amount of error on the object positions, particularly in the "depth" direction. We require a way of comparing the models, given a correspondence hypothesis, in a rotation and translation invariant manner.

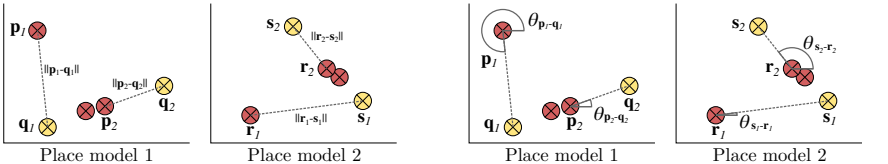
We approach this by considering *pairs of correspondences* - that is, a hypothesis $(i \rightarrow i', j \rightarrow j')$, where object i in the first image corresponds to object i' in the second image, and likewise j and j', with $i \neq j$ and $i' \neq j'$. Since each pair defines a transform between the coordinate frames of the place models, we can extract features which characterise it in a rotation invariant manner. Significantly, this approach has a substantial speed advantage: since the number of objects in each image is quite small (in this our case, no more than 10), the number of pairs of correspondences is not very large - easily small enough to evaluate them all. Thus, the amount of computation required for each *full* correspondence hypothesis is minimised, as features for the relevant pairs-of-correspondences are precomputed and looked up in a table.

We extract 3 types of feature from the models for each pair of correspondences which will be used to compute $p(C|\mathcal{F})$, the probability that some full correspondence hypothesis is correct. In addition, we use two further scores - the residual on the ground plane estimate described in Section 4 and the confidence scores given by the object detector.

5.1 Feature Extraction

For this section, we define the following notation. For a given pair of images, assume that we have the 2D object position vectors $\mathbf{p}_i$ and $\mathbf{q}_i$, which are the positions of the objects in the first place model, and $\mathbf{r}_i$ and $\mathbf{s}_i$ which are the objects in the second place model, for the i^{th} pair of correspondences. See Figure 4 for examples.

Object distance score. (Figure 4(a)). For a true pair-of-correspondence hypothesis, it should be that the length between the pairs of objects in each image is very similar - i.e, the magnitude of $\|\mathbf{p}_i - \mathbf{q}_i\| - \|\mathbf{r}_i - \mathbf{s}_i\|$ is close to zero. Thus, we use this as a feature and express the object distance score for pair of correspondences i, as $\mathcal{D}_i = \sqrt{\left|\|\mathbf{p}_i - \mathbf{q}_i\| - \|\mathbf{r}_i - \mathbf{s}_i\|\right|}$.



(a) Object distance score (b) Rotation consensus score

Figure 4: Illustrations of some place model measurements used as features.

Rotational transform consensus score. (Figure 4(b)). Each pair of matches defines a rotational transform between the coordinate frames of the two place models. Let $\theta_{\mathbf{v}}$ be the anticlockwise angle of 2D directional vector $\mathbf{v}$, relative to some fixed reference direction. For a particular pair of correspondences, the value $\theta_{\mathbf{q}_i-\mathbf{p}_i} - \theta_{\mathbf{s}_i-\mathbf{r}_i}$ is an approximation of the rotational relationship between place models. If a hypothesis is correct, it should be the case that all the pairs of correspondences defined by the hypothesis agree on this value. We therefore measure the agreement between the relevant pairs by computing a histogram of estimated rotation. The largest bin value in the histogram is the feature score.

Formally, let the function hist(θ) produce a 10-bin histogram of the input angle, with the value being counted in two bins to reduce boundary effects. Thus, this will be a vector with eight zeros and two ones. We can precompute the value of the hist function for each pair of correspondences. Then, when scoring a full hypothesis, we simply look up the relevant histograms and sum them, thus giving a histogram of estimated rotation. The feature score is simply the maximum bin count, given by $\mathcal{R} = \max\left(\sum_i \text{hist}(\theta_{\mathbf{q}_i-\mathbf{p}_i} - \theta_{\mathbf{s}_i-\mathbf{r}_i})\right)$. A high value of $\mathcal{R}$ indicates consensus and is correlated with correct hypotheses.

Edge orientation histogram score. Since the majority of our features characterise the geometry of places, we have also used some appearance information from the input images in the form of edge orientation histograms. Again, the scores are computed for pairs of correspondences. As we have estimated the relative pose of the camera and ground plane, we can rectify the image so that we are looking down at the ground plane. We then look at the rectangular strip of ground plane which joins two objects in one view (with detected objects masked out), and compare it with a similar strip in the other view, as shown in Figure 5. We use a thresholded canny edge detector to find edge features and compute a normalised histogram of their orientations. As shown in the figure, we divide the strips into 3 segments and compute a histogram for each one, which gives a rough notion of the position of edge features. These histograms are precomputed for every pair of objects in each view. Then, we

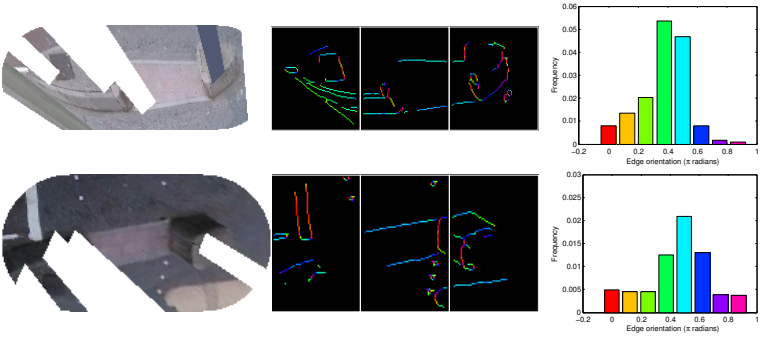


Figure 5: An example of a matching pair of ground plane strips from different viewpoints, the extracted edge features, and corresponding edge orientation histograms.

precompute the Euclidean distance between the histograms for each pair of correspondences.

We also observe that for objects which are very close together, the direction of the vector between them becomes very unstable, making the histogram distance unreliable. To account for this we weight the score by the mean of the object vector lengths in each view. Formally, let the function $\text{edge}(j,\mathbf{v},\mathbf{u})$ return the edge orientation histogram for the j^{th} segment of ground plane strip between 2D vectors $\mathbf{v}$ and $\mathbf{u}$. Then, the edge orientation feature score for a pair of correspondences i is given by $\mathcal{E}_i = \sum_{j=1}^{3} w_i \|\text{edge}(j,\mathbf{p}_i,\mathbf{q}_i) - \text{edge}(j,\mathbf{r}_i,\mathbf{s}_i)\|$ where $w_i = \frac{1}{2}(\|\mathbf{p}_i - \mathbf{q}_i\| + \|\mathbf{r}_i - \mathbf{s}_i\|)$.

Additional scores. As well as extracting the 3 features described, we use two additional scores. We define $\mathcal{G}_1$ as the residual of the ground plane estimate for the set of 5 objects from the first image, $\mathcal{G}_2$ as the residual for the second image, and $\mathcal{O}$ as the sum of the object detection confidence scores for all 10 objects in the hypothesis (5 from the first view, and 5 from the second). These scores are included due to the observation that a correct hypothesis is likely to have a low ground plane residual and high confidence scores for the objects.

5.2 Probabilistic result

Having extracted the features described above, we wish to compute the probability that a hypothesis is correct. For simplicity, we assume independence between all feature scores, yielding a naive Bayes classifer:

$$p(C|\mathcal{F}) = p(C|\mathcal{R})p(C|\mathcal{G}_1)p(C|\mathcal{G}_2)p(C|\mathcal{O})\prod_{j=1}^{\binom{5}{2}} p(C|\mathcal{D}_j)p(C|\mathcal{E}_j) \tag{1}$$

Note that the product is over every pair of correspondences; since we have 5 correspondences, there are $\binom{5}{2} = 10$ pairs in our system.

We take a machine learning approach to estimate each of the above the $p(C|score)$ terms. In each case, the distributions for $p(score|C)$ and $p(score|\neg C)$ are estimated from training examples. This necessitates knowledge of C in the training data, so all matching places in our dataset have been hand-labelled with the correct object correspondence. Then, Bayes' theorem is used to estimate the value of $p(C|score) = \frac{p(score|C)}{p(score|C)+p(score|\neg C)}$. Note that it does not make sense to assume a prior for C, the event that we encounter a correct hypothesis, so the equation is given assuming that $p(C) = 0.5$.

6 Results

To assess the performance of our system, we collected a dataset of 40 locations, each with between 2 and 4 images from widely different viewpoints. We then trained the system as described in Section 5.2. Note that this data is independent from the data used to train the object classifiers. Since we are simply learning distributions over *comparisons* of places, not about the places themselves, we decided to train the dataset on a subset of the test dataset to maximise use of the data. To verify that the results were not biased, we tried repeatedly training the system on a random 50% subset of the dataset and running the test again. We found that the learned probability distributions were very similar each iteration, and that the recognition performance did not change by more than about 2%.

A place recognition experiment was then performed. Each image from the dataset was compared against every other image to compute the the posterior probability $p(C|\mathcal{F})$ that the images depict the same place. Figure 7 illustrates the ranked positions of hypotheses according to this probability for each test image.

Table 1 states the performance of our system under several conditions. The "grouped" score is simply the percentage of test images for which an image from the same place was

Figure 6: Some examples of our system's output. In each column, the first two rows are the input images and the last row is the hypothesis given by our system. In all cases, the first image is represented by the red camera.

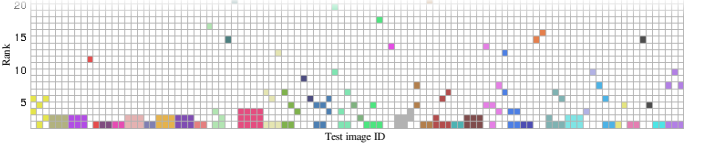


Figure 7: Illustration of the rank of hypotheses for each test image. Coloured squares are images from the same place as the test image (i.e. correct hypotheses), and their y position shows their rank. The colours indicate which location the test image belongs to.

chosen as the most likely match, simulating a place recognition scenario in which we have made a small number of previous observations of each place. It is interesting however to consider a harder case in which, for each test image, there is only a single matching image in the database. The "pairwise" score simulates this situation by removing all but one of the matching images for each test image.

We also observed that some discriminative ability of the system is provided by the different object classes - so a place with objects of class "sign" and "bollard" cannot possibly match with a place containing only "traffic light" objects. Whilst this is a legitimate place recognition scenario, we wanted to observe the discriminative ability of the features alone. Thus, we also tested the system on a "restricted class" subset of the dataset with 30 locations, all of which contained the same two object classes, meaning that almost every image was capable of valid object correspondences with every other image. Clearly this is a harder case, however Table 1 shows that performance was still reasonable.

We have compared our performance against that of the GIST descriptor [20], which attempts capture the general appearance of the scene and therefore does not necessarily require images with overlapping viewpoints. In this way it seems to be the closest work for recognising the types of images we are dealing with, although their approach is significantly different and GIST does not perform well on our dataset.

Table 1: Our system's place recognition performance, and a comparison with GIST

	Grouped	Pairwise
Restricted class dataset	67.9%	54.5%
Full dataset	73.1%	61.8%
GIST	19.2%	21.4%

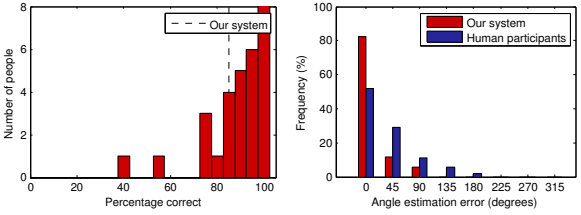


Figure 8: The results of the human experiment for humans and our system.

(a) Recognition performance (b) Orientation error

Figure 6 demonstrates some examples of successful and unsuccessful place recognition. For display purposes the pose of the cameras is computed using Horn's absolute orientation method [15] on the given hypothesis. In general, we found that the system is able to find a good hypothesis for pairs of matching places - however in cases where it failed, there was a *similar-looking* place (at least according to the features we extracted) which caused confusion. Intuitively, this is not a surprising mode of failure. All but one of our features are based on the estimated geometry of the scene, and we have only 5 points with a reasonable amount of error. The error in many cases may be larger than the difference between similar-looking places, so the system is prone to confusion in these situations.

6.1 Human experiment

As we are not aware of any directly comparable place recognition system, we wished to obtain some kind of benchmark to assess the performance of our method. Thus, we conducted a reduced test to compare our system against human performance. 30 participants were presented with 20 images of places, and were asked for each one to choose which of 5 other images represents the same place in the world. We also asked the participants to estimate the relative pose of the cameras by choosing one of eight rotational transforms around a circle. We then tested our system on the same data.

In terms of place recognition ability we found that the average person performed slightly better than our system (Figure 9(a)), however the results seem to verify that the problem is not trivial - about a third of participants performed as well or worse than our system. Interestingly however, when the camera pose estimates were compared with an expert-labelled ground truth, our system made much better estimates than people (Figure 9(b)). This, along with opinions gathered from participants during the experiment, indicated that humans are often using other, more specific semantic cues such as markings and structure on the ground or the style of the area to recognise places, rather than trying to align 3D maps of the scenes.

7 Conclusion

We have presented a system which performs place recognition using objects, and have demonstrated encouraging results on a dataset where matching images may have completely different viewpoints of the same scene. The results revealed a moderate amount confusion between places, which combined with the neccessity of some restrictive, although not unreasonable assumptions mean that the system is not very general and is unlikely to scale well to larger datasets. Nonetheless, we believe that this work has shown that it is certainly possible to perform recognition using semantic features when there is no overlap of image features at all, on a dataset which even some humans find difficult.

The most common failure of our system is when the geometry of a pair of non-matching places was more similar than the amount of error on the measurements - although, with only five points and as few as one or two object classes in the scene, this does not seem

surprising, so the system appears to do a good job with the features we are using. For future work, motivated by the limitations of geometry and by the human results which showed the importance of other semantic features, we intend to investigate the use of more appearance-based features to accompany the edge features which we are already using, particularly those on the ground, as well as ways of characterising the style or architecture of a place to aid recognition.

Acknowledgements

This work was funded by the UK EPSRC.

References

[1] Adrien Angeli, David Filliat, Stephane Doncieux, and Jean-Arcady Meyer. Fast and incremental method for loop-closure detection using bags of visual words. *IEEE Transactions on Robotics*, 24(5):1027–1037, October 2008.

[2] Sid Yingze Bao and Silvio Savarese. Semantic structure from motion. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, 2011.

[3] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (SURF). *Comput. Vis. Image Underst.*, 110(3):346–359, June 2008.

[4] S. Bazeille and D. Filliat. Combining odometry and visual loop-closure detection for consistent topo-metrical mapping. *RAIRO - Operations Research*, 44(4):365–377, January 2011.

[5] C. Cadena, D. Gálvez-López, J.D. Tardós, and J. Neira. Robust place recognition with stereo sequences. *Robotics, IEEE Transactions on*, 28(4):871–885, 2012.

[6] Mark Cummins and Paul Newman. FAB-MAP: Probabilistic localization and mapping in the space of appearance. *The International Journal of Robotics Research*, 27(6): 647–665, June 2008.

[7] Mark Cummins and Paul Newman. Highly scalable appearance-only SLAM - FAB-MAP 2.0. In *Robotics Science and Systems*, pages 1–8, Seattle, USA, 2009.

[8] Pedro Felzenszwalb, Ross Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010.

[9] Stephen Gould, Richard Fulton, and Daphne Koller. Decomposing a scene into geometric and semantically consistent regions. In *IEEE International Conference on Computer Vision*, pages 1–8. Ieee, September 2009.

[10] Abhinav Gupta, Alexei A Efros, and Martial Hebert. Blocks world revisited: Image understanding using qualitative geometry and mechanics. In *ECCV*, volume 6314 of *Lecture Notes in Computer Science*, pages 482–496. Springer, 2010.

[11] Osian Haines and Andrew Calway. Detecting planes and estimating their orientation from a single image. In *British Machine Vision Conference*, September 2012.

[12] Derek Hoiem, Alexei a. Efros, and Martial Hebert. Recovering surface layout from an image. *International Journal of Computer Vision*, 75(1):151–172, February 2007.

[13] Derek Hoiem, Andrew N. Stein, Alexei a. Efros, and Martial Hebert. Recovering occlusion boundaries from a single image. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[14] Derek Hoiem, Alexei a. Efros, and Martial Hebert. Putting objects in perspective. *International Journal of Computer Vision*, 80(1):3–15, April 2008.

[15] Berthold K. P. Horn. Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America A*, 4(4):629–642, 1987.

[16] K. Konolige, J. Bowman, J. D. Chen, P. Mihelich, M. Calonder, V. Lepetit, and P. Fua. View-based maps. In *Proceedings of Robotics: Science and Systems*, Seattle, USA, June 2009.

[17] M. Labbe and F. Michaud. Appearance-based loop closure detection for online large-scale and long-term operation. *Robotics, IEEE Transactions on*, PP(99):1–12, 2013.

[18] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, November 2004.

[19] David Nistér and Henrik Stewénius. Scalable recognition with a vocabulary tree. In *CVPR (2)*, pages 2161–2168, 2006.

[20] Aude Oliva and Antonio Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42:145–175, 2001.

[21] R. Paul and P. Newman. FAB-MAP 3D: Topological mapping with spatial and visual appearance. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 2649–2656, 2010. doi: 10.1109/ROBOT.2010.5509587.

[22] A. Ranganathan and F. Dellaert. Semantic modeling of places using objects. In *Proceedings of Robotics: Science and Systems*, Atlanta, GA, USA, June 2007.

[23] Ashutosh Saxena, Min Sun, and Andrew Y Ng. Make3D: Learning 3D scene structure from a single still image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(5):824–840, 2009.

[24] Grant Schindler, Matthew Brown, and Richard Szeliski. City-scale location recognition. *CVPR*, pages 1–7, June 2007.

[25] Josef Sivic and Andrew Zisserman. Video Google : A text retrieval approach to object matching in videos. *ICCV*, pages 2–9, 2003.

[26] E.B. Sudderth, A. Torralba, W.T. Freeman, and A.S. Willsky. Depth from familiar objects: A hierarchical model for 3D scenes. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2410–2417, 2006.

[27] Antonio Torralba, Kevin P Murphy, William T Freeman, and Mark A Rubin. Context-based vision system for place and object recognition. In *Ninth IEEE International Conference on Computer Vision*, volume 1, pages 273–280, 2003.