RANSAC with 2D Geometric Cliques for Image Retrieval and Place Recognition

Edward Johns and Guang-Zhong Yang
Imperial College London
e.johns@imperial.ac.uk

Abstract

Visual place recognition with local features often uses RANSAC to estimate a 3D transformation between images. However, degenerate cases can exist where samples drawn fit the model, but are geometrically unlikely. We propose to eliminate these by ensuring that all samples agree weakly in 2D pairwise geometry, forming a geometric clique. The pairwise geometries also enable biasing of the sampling to speed up RANSAC by early rejection of unlikely configurations. We then show that by training over a number of images for each place, the expected pairwise geometries can be learned individually for each place, accounting for the underlying scene structure and range of likely viewpoints. Experiments conducted on a new database show how our proposal outperforms similar methods in both retrieval and recognition performance, and computational efficiency.

1. Introduction

Visual place recognition has three key challenges when attempting to match a query image to a database. First, dealing with a viewpoint change between the query and database, offering robustness to translation, rotation and scale. Second, coping with environment changes that have occurred between capturing the two images, such as time of day, weather, and dynamic bodies. Third, scaling gracefully with the size of the database and allowing for fast recognition with a practical memory footprint. In this paper, we present a new method that allows for efficient recognition from a wide range of viewpoints and environment conditions, by computing a compact generative model of each place, and learning the relative displacements of local features that would be expected from a new query image.

There are two main ideologies for devising a place recognition method. Given a diverse set of training images for each place to cover its full range of viewpoints and environment conditions, place recognition can be addressed in an image-based approach, by creating a database of images and attempting to match a query image to the database images [24, 21]. Alternatively, for more efficient databasing, or to learn repeatable or discriminative properties of each place, a model-based approach can be used by learning over the range of viewpoints and environment conditions in the place's training images, and matching a query to these models in the database [8, 12, 13]. Our proposed method is an example of this second approach. Under changing environments, this approach is particularly effective at learning invariant features, such as over different times of the day [9] or different seasons [11, 10]. However, if training images covering the required range of environment conditions are not explicitly available, then alternative approaches must be adopted [16, 15].

Place recognition is closely related to the field of image retrieval [17, 23, 3]. State-of-the-art image retrieval methods typically involve extracting local features [14], encoded with Bag-Of-Words (BOW) indexing for fast retrieval [17], followed by a more robust geometric verification stage based on a RANdom SAmple Consensus (RANSAC) estimation of the Fundamental Matrix [6]. Whilst the RANSAC algorithm itself has been improved in recent years for image retrieval applications [2, 4, 5, 20, 19], it still allows for degenerate cases, whereby the best-fit model is represented by a highly-unlikely arrangement of local features in the physical scene, as in Figure 1.

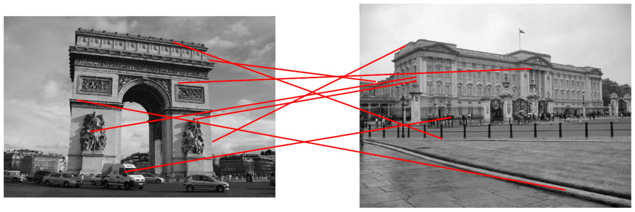


Figure 1: Feature correspondences can form degenerate solutions to a RANSAC-based estimation of the Fundamental Matrix, whereby their arrangement is geometrically unlikely even though it satisfies the model.

We propose to use the 2D pairwise geometries of features to ensure that any set of feature correspondence samples drawn by RANSAC, loosely agree with each other in terms of image distance and angle, forming a *geometric clique*.

Then, we require that all other correspondences forming a consensus with this model must agree in 2D pairwise geometry with the clique. Furthermore, extending the RANSAC algorithm for model-based place recognition has received little attention. We propose a method to learn these 2D constraints between features by observing the change in scene appearance over a range of viewpoints, to create a generative model of pairwise relationships for each place.

Closely related to our work, the SCRAMSAC algorithm [22] proposed a spatial consistency check on samples drawn by RANSAC for model hypothesis. Here, all pairs of feature correspondences used to propose a model must have similar image locations across the two images, otherwise they are rejected. We extend this concept by enforcing constraints on not just the distance, but also the angle, between features. We also show how the global geometric consistency of each correspondence can be used to bias the RANSAC sampling for faster convergence.

Learning 2D pairwise relationships across a range of viewpoints has been addressed previously in [8], although without incorporating the power of RANSAC-based 3D geometry. In [12], a RANSAC stage was forgone altogether by embedding the pairwise geometries in an inverted index, but performance was still limited compared to RANSAC approaches.

2. Pairwise feature geometries

Correspondences in our framework are based on the visual word assignments of features. However, computation of pairwise geometries grows quadratically in the number of correspondences, which can be several hunderd per image pair. We therefore initially eliminate most correspondences by considering a fast, yet weak, Hough-voting method based on [23] and similar to [18]. The top 100 correspondences are then retained for further processing.

We now consider how pairwise feature geometries can be used to eliminate correspondences that may conform to the RANSAC 3D model, but are in fact highly unlikely due to 2D constraints within the image. By computing the distances and angles between all features of correspondences in the first image, and comparing these values to those respective values in the second image, we can learn which pairs of correspondences agree with each other based on the 2D geometry. This not only allows for detection of false positive correspondences that are apparently inliers according to the estimated 3D model, but also allows for a faster estimation of this model during the RANSAC algorithm, by pre-emptively terminating a candidate model if the sample correspondences do not agree in 2D geometry.

For two images t_1 and t_2, let us define m_i as a correspondence with features u_1 and u_2 in the two images, and m_j as a second correspondence with features v_1 and v_2, as in Figure 2. We also define $\delta_{u_1 v_1}$ and $\psi_{u_1 v_1}$ as the distance and angle between the two features in the first image, and similarly $\delta_{u_2 v_2}$ and $\psi_{u_2 v_2}$ for the second image. These are then made scale-invariant and orientation-invariant, by dividing $\delta_{u_1 v_1}$ by the scale of u_1 and subtracting the orientation of u_1 from $\psi_{u_1 v_1}$, respectively, and similarly for features u_2 and v_2. Then, we compute the difference between the distance and angle of two correspondences m_i and m_j, and assign these to $d\delta_{m_i m_j}$ and $d\psi_{m_i m_j}$ respectively.

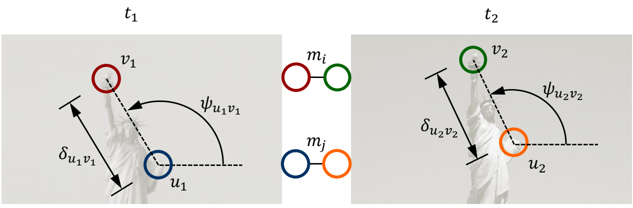


Figure 2: Notation for pairwise feature geometries. These are used to check the 2D geometric consistency of correspondences across two images.

2.1. Adjacency Matrix for Sample Rejection

For a set of n correspondences, let us now define a binary adjacency matrix A of size $n \times n$. Each element A_{ij} is set to either 0 or 1, defining whether or not the pair of correspondences m_i, m_j is geometrically consistent. This consistency is determined by whether the distance and angle differences both lie within specified thresholds $d\delta_t$ and $d\psi_t$ respectively:

$$A_{ij} = \begin{cases} 1 & \text{if } d\delta_{m_i m_j} < d\delta_t \wedge d\psi_{m_i m_j} < d\psi_t \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

In this way, matrix element A_{ij} is set to 1 if and only if the correspondences m_i and m_j agree in both distance and angle to an acceptable level. The values of $d\delta_t$ and $d\psi_t$ are free parameters and can be adjusted empirically, in reflection of both the range of expected viewpoints and the scene structure. If viewpoints are narrow and the scene is close to planar, then the allowable differences can be restricted by much smaller thresholds whilst still accepting all true correspondences.

The modified RANSAC algorithm based on the binary adjacency matrix then proceeds as follows. On each iteration, sets of sample correspondences are randomly selected, as usual, for the 8-point algorithm [7]. In this case however, after each sample is selected for a given set, the sample is compared against all correspondences currently existing in the set. If the sample is not consistent with any of these, then the whole set is discarded, and a new random set is selected. The reasoning behind this is that if any two correspondences do not agree based on pairwise geometries, then the subsequent estimated model will likely be degenerate in 2D image space even though it fits the 3D model. In this way, only those sets of samples which form a geometric clique are actually processed to completion, offering an additional boost in speed.

During model consensus, when the full set of correspondences are verified against the estimated model, we again ensure that each correspondence is consistent with the sample correspondences based on A. We experimented with requiring that all inlier correspondences must be consistent with each other, not just the sample correspondences, but this required large thresholds of $d\delta_t$ and $d\psi_t$, which in turn tended to introduce false positive correspondences.

2.2. Biased Sampling

Thus far in the RANSAC algorithm, samples for model estimation are chosen at random from the set of correspondences, until an acceptable model estimation has been achieved. However, if we can bias the sampling towards those correspondences which we know in advance are more likely to satisfy this final model, then the algorithm will converge much faster because the best set of sample correspondences will be chosen earlier. As such, we propose to use the pairwise geometries to weight each correspondence in this way, and bias the sampling accordingly.

A score α_i is assigned to correspondence m_i, by summing the elements in row i of A, indicating the number of correspondences which agree in pairwise geometry:

$$\alpha_i = \sum_{j=1}^{n} A_{ij} \tag{2}$$

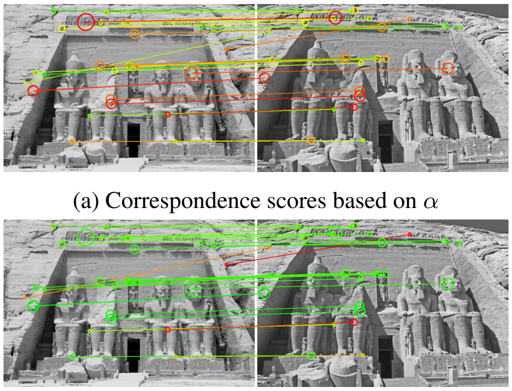
Now, the value of α considers all correspondences independently when computing the summation and inferring a correspondence's global geometric compatibility. However, a more robust score would be gained from giving more importance to those correspondences which themselves have global compatibility. In this way, a false positive correspondence, which happens to agree with only one other correspondence, will not increase that correspondence's score as much as a true positive correspondence which agrees with many other correspondences. As such, we define a second correspondence score β as the summation over the row of B as before, but with each element weighted by its own α score:

$$\beta_i = \sum_{j=1}^{n} \alpha_i A_{ij} \tag{3}$$

Figure 3 shows the difference in using scores based on α and β. The strengths of α shown in 3a are in general roughly reflective of the correspondences global geometric compatibility, but some obvious inliers are assigned to a low score, and similarly outliers to a high score. The strengths of β in 3b are a much better reflection, and create a very strong division between inliers and outliers.

The weighted RANSAC algorithm then proceeds as follows. Correspondences are first ranked in order of their score β. Then, samples are drawn from a biased distribution, such that each correspondence has a probability of selection related to its ranking. The biased distribution is created by adding $\{n + 1 - l_i\}$ copies of correspondence m_i, where l_i is the ranking of that correspondence, such that each correspondence has one more copy in the pool of correspondences than the next highest-ranked correspondence. The remainder of the algorithm proceeds as before using B to reject inconsistent samples.

If two images are of similar viewpoint as in Figure 3, it would be acceptable to simply sample the top correspondences based on their scores, without any random sampling. However, this strategy rapidly degrades in performance as the imaging conditions differ and the ratio of inliers-to-outliers decreases. We also investigated weighting each correspondence by its score rather than its ranking, but this similarly placed too much emphasis on the top correspondences and offered poor flexibility.



(a) Correspondence scores based on α

(b) Correspondence scores based on β

Figure 3: Correspondence scores based on global compatibility with all other correspondences, where green indicates a high score. The correspondence scores defined by β are a much better reflection of global geometric consistency than those defined by α. Correspondences with higher scores are then given more bias in the RANSAC sampling.

3. Generative Place Models

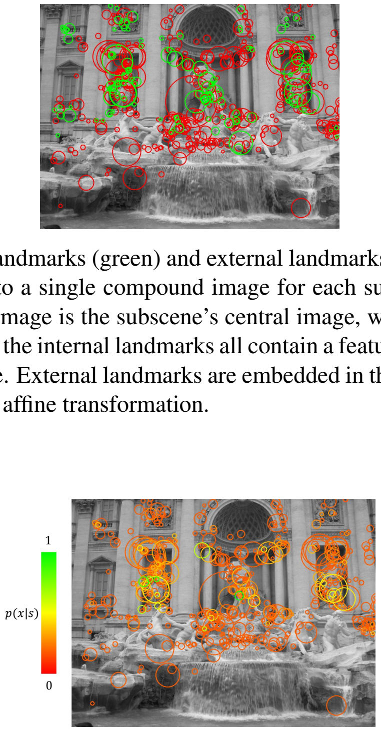
We now propose to build a generative model of a place's appearance to explicitly learn pairwise geometries individually for each scene, without having to decide on the parameters $d\delta_t$ and $d\psi_t$ empirically as before. If all images captured of a place were superimposed on each other about a common point, then the range of distances and angles between feature pairs represent the expected ranges if a further image was captured of the place from anywhere within that range of viewpoints. Thus, we can learn $d\delta_t$ and $d\psi_t$ for that particular place by simply observing the relative geometries of feature pairs in the training images. However, if the set of viewpoints in these images is too great, then $d\delta_t$ and $d\psi_t$ will be too large to offer discriminative capacity, and so the first step is to decompose each place into smaller clusters of images representing similar viewpoints.

3.1. Subscenes and Compound Images

Let us first define a database $\mathcal{R}$ of places, with each place $r \in \mathcal{R}$ assigned a set of training images $t \in \mathcal{T}_r$. For each place r, a set of subscenes $\mathcal{S}_r$ are established, with each subscene $s \in \mathcal{S}_r$ assigned a subset of training images $\mathcal{T}_s \subset T_r$. Each subscene s is then associated with a *compound image* $\mathbf{t}_s$, which is a "synthetic" image formed as a composition of the subscene's training images. Each place is thus represented by several such compound images, one for each subscene, and each of which is responsible for a distinct combination of viewpoint and illumination conditions as defined by the subset's training images.

Each compound image itself is based upon one of the subscene's training images denoted the *central image*, $\hat{t}_s$. Then, for each subscene s, a set of landmarks $\mathcal{X}_s$ are generated by tracking features across the subscene's training images, with each landmark representing the same real-world point observed along the feature track, similar to [8, 12]. Each landmark is defined as either an *internal* or *external* landmark, and embedded in the compound image accordingly. Internal landmarks are those whose feature track contains a feature in the central image, otherwise the landmarks are classed as external. For the internal landmarks, the estimated location is simply the original location of the central image's feature from the landmark's feature track. For the external landmarks, we estimate an affine transformation between all images along the landmark's feature track, and the central image. The landmark's position in the central image is then taken as the median of all these transformations. Note that these positions are not directly used for our place recognition method, but are necessary for the baselines to which we compare, and for subscene visualisation. For our method, the landmark positions are in fact a range rather than a single point, the computation of which will be discussed in Section 3.3.

To determine the scale and orientation of each landmark, the features in the feature track are first scaled and rotated according to the image scale ratio and orientation difference between the image containing the feature, and the central image, and then the median is taken over the track. The landmark scale and orientation is then taken as the median across these adjusted features. Finally, each landmark $x \in \mathcal{X}_s$ is assigned an observation probability $p(x|s)$, reflecting the stability of the landmark across the subscene, defined as the number of images containing the landmark's feature track, divided by the total number of images in the subscene. This is then used to add further bias to the RANSAC sampling stage, by giving more weight to those feature-to-landmark correspondences who's associated landmarks are stable and hence likely to be true positive correspondences. To achieve this, we weighted each correspondence's β score in Equation 3 by $p(x|s)$ for that landmark. Figure 4 shows the content of a subscene, with all internal and external landmarks embedded in the compound image, each with an associated observation probability.



(a) Internal landmarks (green) and external landmarks (red) are all combined into a single compound image for each subscene. The background image is the subscene's central image, where the feature tracks of the internal landmarks all contain a feature within the central image. External landmarks are embedded in the compound image via an affine transformation.

(b) Each landmark is assigned an observation probability $p(x|s)$, reflecting the landmark's stability across the subscene.

Figure 4: A typical compound image for a subscene, reflecting landmark positions and observation probabilities.

3.2. Image Clustering

To cluster each place's training images into a set of subscenes, several methods exist including k-means clustering [21], agglomerative clustering [24], and kernel vector quantization [13]. Our framework is a special case for clustering in that all training images representing a compound image must have a valid affine transformation specifically with respect to the central image, such that the positions of the external landmarks can be determined. Furthermore, if we want the generative pairwise geometries to cover the full range of viewpoints in the training dataset, then every image that has formed at least one affine transformation with another image must be included, otherwise the particular viewpoint for that image may be excluded from the model. However, there is no optimum solution that will generate clusters satisfying this constraint without including overlap between clusters.

The proposed solution to achieving this particular structure aims to minimise this cluster overlap in a graph-cut procedure, by pruning out large sets of similar images first and repeatedly subdividing the scene until all images are a member of at least one subscene. To do this, every image is designated a score according to the number of images with which it has formed an affine transformation, where the number of feature correspondences between the two images is at least 15. To speed up the clustering stage, a weak affine transformation was first estimated by the 4 DOFs represented by each single correspondence, as proposed in [18]. Then, the RANSAC stage with pairwise geometries was run by sampling from inliers from the estimated affine transformation. Once scores have been assigned to all images, the algorithm recursively chooses the image with the highest score, and designates it as the central image for a new subscene, which is then composed of this central image and all images with which it forms an affine transformation. Every image in that subscene is then removed from the list of available central images, because the particular viewpoint of that image has now been included in the model. The algorithm continues in this way until all images in the dataset have been included in at least one subscene.

3.3. Pairwise landmark geometries

If we make the assumption that the query image falls within the range of viewpoints represented by the subscene's training images, then the expected range of pairwise landmark positions can be directly observed by overlaying the training images on the subscene's central image, and noting all the positions of each pair of landmarks. Now, for a subscene with n landmarks, explicitly learning the pairwise distances and angles for every pair would result in memory requirements of the order O(n^2), and so this is not a scalable solution. Instead, we propose to calculate the ranges in $x-y$ image space, rather than polar distance-angle image space as before, and store these ranges as fixed image positions on the central image, rather than storing ranges for each landmark pair. In this way, pairwise geometry ranges for landmarks are calculated at runtime by comparing the extents of these $x-y$ ranges, eliminating the large memory footprint that would be required to store them offline.

We learn these geometry ranges by aligning each subscene training image with the central image, and calculating the disparity between the two images in the distance and angle between each landmark pair. First, we roughly scale and rotate each training image with respect to the central image, by taking the median of the scale ratios and orientations across all correspondences between the two images. Then, we denote a *pivot feature* as a feature from the central image's set of features which has formed a correspondence with a feature in the training image, i.e. the two features share a feature track. The training image is then aligned with the central image by translating and rotating all features, such that the pivot feature and its correspondence are located at the same position in the central image. Figure 5 illustrates this concept.

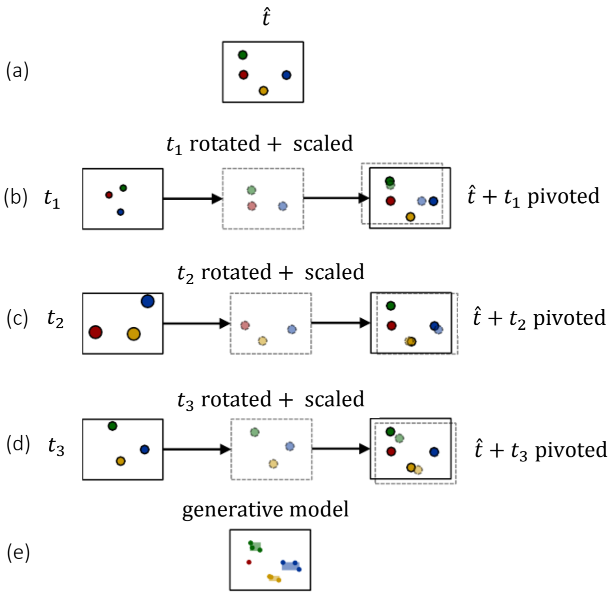
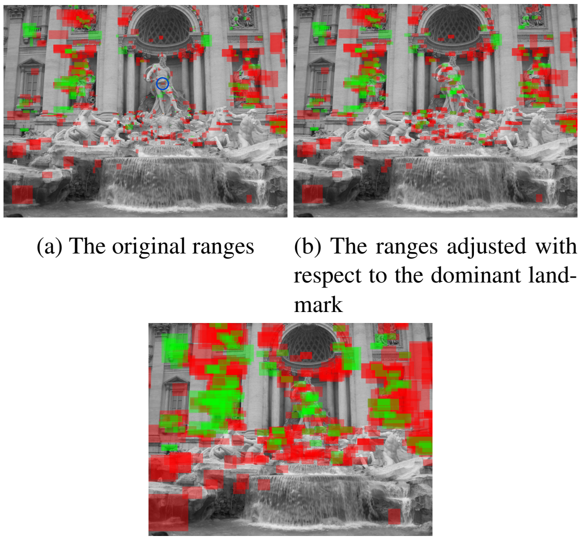


Figure 5: In order to learn the generative pairwise geometries for a subscene, all training images must be aligned with the subscene's central image. This is achieved by first scaling and rotating the training images accordingly, and then ”pivoting” the resulting image with respect to one of the features in the central image, denoted the pivot feature. (a) shows the central image, and (b), (c) and (d) demonstrate the alignment process. In (b) and (c), the pivot feature is the red feature, whereas in (d), the training image does not contain this feature, and thus pivoting is via the blue feature. In (e), the resulting range of landmark positions are shown based on these alignments.

The choice of pivot feature is an important one in ensuring a good image alignment. The first consideration is that it is unlikely that a single feature in the central image will form a correspondence with all of the subscene's other training images. But a pivot feature must be available for each of these images, such that every image in the subscene can be aligned with the central image. The second consideration is that we want to minimise the $x-y$ ranges to help discriminate between subscenes, whilst still accurately fitting the generative model. If the subscene is perfectly planar and we make an affine assumption about the camera projection properties, then the $x - y$ ranges would all be the same, regardless of which pivot feature we choose. However, if the subscene is planar except for one feature, and that feature is chosen as the pivot feature, then the $x - y$ ranges would be much larger than if a planar feature was chosen as the pivot feature.

In essence, we want to maximise the overall "planarity" between the pivot feature and all other features to be aligned. If we assume that a subscene is represented by a dominant plane, with all landmarks spatially located at some distance to that plane, we want the pivot feature to be representative of a landmark that lies on this plane. To achieve this, we rank features in the central image in order of the number of correspondences which the feature forms with the other images in the subscene. The rationale behind this is that landmarks lying on a dominant plane are much more likely to form correspondences across two images, due to the smaller discrepancy in relative image positions. After ranking features in this way, every training image in the subscene is assigned the highest-ranked feature which forms a correspondence with that image, and the image is aligned with the central image via this pivot feature.



(a) The original ranges

(b) The ranges adjusted with respect to the dominant landmark

(c) The ranges adjusted with respect to a scaling factor of $k = 2$

Figure 6: Generative pairwise landmark geometries are represented by $x - y$ ranges in the compound image. The original ranges in (a) are adjusted in (b) to account for landmarks not being observed in all the subscene's training images, and adjusted in (c) to allow for recognition outside the range of viewpoints in these images. The most commonly-used pivot feature is highlighted in blue in (a). Note that as the landmarks move further from the pivot, the $x - y$ ranges increase due to non-planarity and perspective effects.

3.4. Regularisation

One of the goals of learning generative pairwise landmark geometries in this way, is for every subscene to cover the full range of viewpoints represented by its training images. However, this is only satisfied if every landmark is tracked across each training image; otherwise, the particular viewpoint for that image is not represented when the landmark's position on the central image is estimated. This may be acceptable when the estimated position is within the landmark's $x - y$ range as determined by the other features in the track, and hence the image's viewpoint is already covered, but when the image represents a more unusual viewpoint then it is not truly represented in a landmark unless the landmark is tracked from that particular image.

We therefore propose to expand each landmark's $x - y$ range on the central image to include those viewpoints which have not been represented. First, let us consider the landmark with the greatest number of features in its track, and denote this the "dominant" landmark. We can make a rough assumption that this landmark is represented by the greatest range of viewpoints. If we now assume that all landmarks lie on an affine plane, then the size of the $x - y$ range for each landmark should be the same if each landmark is represented by the same viewpoints. Therefore, we can estimate the true $x - y$ range of a landmark by scaling it to the size of the dominant landmark's range. In this way, all landmarks cover roughly the same range of viewpoints in the central image, regardless of how many viewpoints are actually represented in the landmark's feature track. If a landmark's $x - y$ range is already greater than that of the dominant landmark, despite it having fewer features in its track, then it can be assumed that this landmark is significantly out-of-plane with respect to the scene's dominant plane, which causes the large range of expected positions on the central image. As such, we retain the original $x - y$ range for these landmarks as they are already greater than the theoretical range if the scene was perfectly planar.

Whilst each subscene can then be considered representative of the viewpoints of its training images, the entire set of subscenes in the database may still not represent all viewpoints that could be encountered. If a query image is captured from a viewpoint not represented by those in the training images, then its landmarks will be observed at positions outside those reflected in the subscenes. Therefore, we introduce a scaling factor k, which scales the $x - y$ image range for each landmark, relative to the adjusted range we have just discussed. k acts as a free variable, the effects of which are presented in Section 4.

Finally, when matching a query image to a compound image, to compute compatibility between a query feature pair and landmark pair, the relative displacement in the query image must be within the relative displacement in the compound image. Figure 6 illustrates these $x - y$ displacement ranges of a set of landmarks for a typical subscene.

4. Experiments

We evaluated our proposed method in terms of both the image retrieval (Section 2) and place recognition (Section 3) applications. Our own dataset was compiled due to the need for representation of a large number of distinct places, together with a large number of training images per place, necessary for building the generative scene models. Most existing datasets consist of only a small number of images per place, and are more suited to retrieval rather than recognition applications. For our dataset, we used Flickr [1] to acquire images of 50 well-known buildings, such as the Eiffel Tower and the Tower of London, with 500 images taken per building. 10 of these images were than taken per building as query images for testing, with all other images for training. Some of the images obtained from Flickr were poor representations of the place of interest, such as being severely occluded, or simply irrelevant, and it was ensured that these images were not used as queries. All images had the largest dimension set to 1000 pixels.

For the image retrieval application, we extracted SIFT features [14] and used a BOW framework with $10k$ visual words and soft assignment [17]. Based on the cosine similarity score with tf-idf weighting, the top 100 databse images per query were passed on to geometric verification. For the place recognition application, all database compound images were considered for geometric verification, due to the smaller number of images stored per place. For each query image, database images were then ranked in order of the number of inliers found to generate precision-recall curves. In the place recognition application, only the compound image with the top-ranked score across all of that place's compound images, was included in the precision-recall scores. In this way, the precision at 100% recall is the effective recognition rate across all places. Unless otherwise specified, implementations of our method included both the biased sampling and geometric clique check.

4.1. Baselines

We compared our geometric cliques method to two baselines for RANSAC-based geometric verification in the image retrieval application. First, the Locally-Optimized RANSAC (LO-RANSAC) method [2], and second, a method inspired by the SCRAMSAC algorithm [22], which we call the Spatial Consistency Check (SCC). In LO-RANSAC, a second stage of sampling is conducted from the inliers of a first stage of sampling, in increase tolerance to image noise. In SCC, a compatibility test is done on correspondence pairs, similar to our method, except that only pairwise distance is computed, ignoring angular information. Both the two baselines use uniform sampling rather than our proposed biased-sampling algorithm. Both our method and SCC additionally use LO-RANSAC in their pipeline, with all RANSAC implementations terminating at an inlier probability threshold of 0.99.

For place recognition, we compared against two baselines for learning scene models from a set of training images. First, inspired by the method [21] (Iconic Images), we retained the set of central images for each subscene and matched directly to these. Second, inspired by the method of [13] (Localised Landmarks), we matched directly to the compound images for each subscene using the landmark positions as computed in Section 3.1. Both the place recognition baselines used our geometric cliques method for image-to-image matching using the best empirically-gauged parameters for compatible pairwise geometries.

4.2. Results

4.2.1 Image Retrieval

For the image retrieval application, we first evaluated the effect of varying the parameters $d\delta_t$ and $d\psi_t$, i.e. the thresholds on allowable distance and angle difference between two feature correspondences for a pass in compatibility. Rather than sweep the two parameters independently, we instead calculated the distance and angle differences between $10k$ feature correspondences from our dataset based on a standard LO-RANSAC method, ranked these values in order of size, and determined the values at various percentiles. Figure 7 shows how the mean Average Precision (mAP) varies as a function of this percentile, p. A peak is found at $p = 80$, such that 80% of all the $10k$ feature correspondences would have passed as compatible at these values of $d\delta_t$ and $d\psi_t$, which were 35 pixels and 29 degrees, respectively. Smaller values placed too harsh a restriction on the inlier set of samples, whereas larger values allowed false positive samples which created degenerate models. As p approaches infinity, the algorithm effectively reverts to LO-RANSAC because the thresholds do not impose any constraints. The values at $p = 80\%$ were then used for the remainder of the experiments.

Figure 8 shows the precision-recall curves for our method and both baselines, by averaging the precision-recall scores of all query images. We see an improvement in performance for our method over both baselines, with better results than SCC owing to the involvement of both angle and distance geometry in the pairwise compatibility check. Mean average precision, over all query images, for LO-RANSAC, SCC and our Geometric Cliques, are 0.546, 0.572 and 0.587. As with our method, the SCC results here represent the best over a range of thresholds on the pairwise feature distances.

The average time for image-to-image geometric verification for LO-RANSAC, SCC and our Geometric Cliques

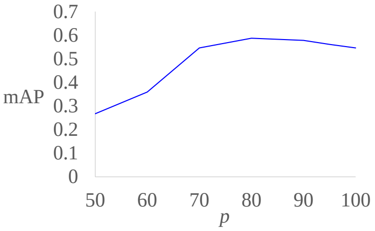


Figure 7: The effect of increasing $d\delta_t$ and $d\psi_t$ on the mean Average Precision of our method in the image retrieval application. p is the percentile at which these values are taken from an existing set of geometries calculated from pairs of inlier feature correspondences.

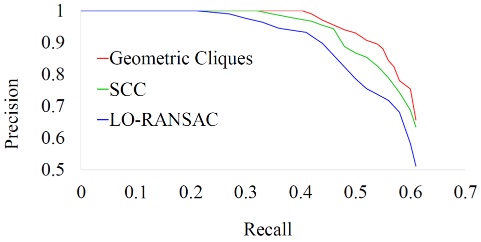


Figure 8: Precision-recall curves for our image retrieval method and the two baselines, averaged over all query images.

method was 48 ms, 22 ms and 12 ms respectively, with our method faster than SCC due to both the stronger geometric constraints and hence greater capacity to detect and early reject degenerate cases, and the ability to bias the sampling based on global consistency of correspondences. Whilst our method requires greater time to extract the pairwise relationships on the query image (22 ms vs. 14 ms with SCC), this is a fixed time and does not scale with the database. Given very large datasets, the query time is of far more importance in judging practical efficiency. Furthermore, our implementation could be speeded up dramatically by discretising image space and reading pairwise geometries from a lookup table.

4.2.2 Place Recognition

For the place recognition application, we evaluated the the scaling factor k of the $x - y$ ranges of landmarks. Figure 9 shows the effect of varying k on the mean Average Precision of our method, where a peak is found at $k = 3$. Similarly to when varying $d\delta_t$ and $d\psi_t$ before, k is a compromise between overfitting and underfitting, and its optimum value is dependent on the scene structure, the range of viewpoints available in the training images, and the range of viewpoints expected in the query images.

Figure 10 then compares two implementations of our method (with two values of scaling factor k) to the two place recognition baselines. We see that whilst our method performs well without any additional scaling ($k = 1$), its performance degrades when a high recall is required, because many of the query images captured from unusual viewpoints have RANSAC samples rejected that are in fact true positives. The Average Precision of the Iconic Images, Localised Landmarks, Geometric Cliques ($k = 1$) and Geometric Cliques ($k = 3$) implementations were 0.733, 0.769, 0.789 and 0.803, respectively. The average time for geometric verification for these four methods was 67 ms, 81 ms, 20 ms and 28 ms, respectively, with our method offering superior speed due to both early rejection of incompatible samples, and biased sampling.

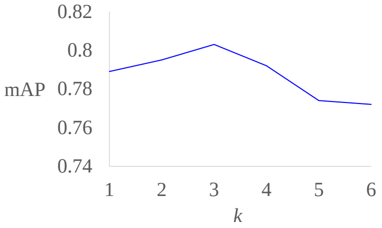


Figure 9: The effect of landmark $x - y$ range scaling factor k on the mean Average Precision of recognition.

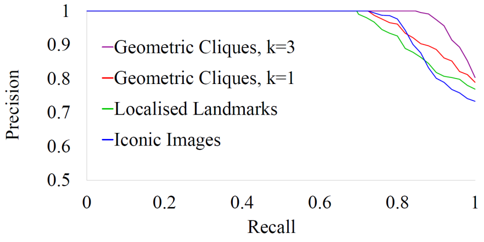


Figure 10: Precision-recall curives for two implementations of our place recognition method and the two baselines, averaged over all query images.

5. Conclusions

In this paper, we have presented a new method for RANSAC-based geometric verification by considering pairwise relationships between feature correspondences, and ensuring that all correspondences that fit the 3D model are also globally consistent in 2D geometry. We name this the method of *Geometric Cliques*, and show how it can be used in both an image retrieval and place recognition application. We introduce a new dataset that provides a significant number of training images per place, necessary for training model-based place recognition systems, and we show how our method outperforms similar baseline techniques both in retrieval and recognition performance, and computational efficiency in geometric verification.

References

[1] Flickr, http://www.flickr.com.

[2] O. Chum, J. Matas, and S. Obdrzalek. Enhancing RANSAC by generalized model optimization. In *Proceedings of the Asian Conference on Computer Vision*, pages 812–811, 2004.

[3] O. Chum, A. Mikulik, M. Perdoch, and J. Matas. Total recall II: Query expansion revisited. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 889–896, 2011.

[4] O. Chum, T. Werner, and J. Matas. Two-view geometry estimation unaffected by a dominant plane. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 772–771, 2005.

[5] J.-M. Frahm and M. Pollefeys. RANSAC for (quasi-) degenerate data (QDEGSAC). In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 453–460, 2006.

[6] R. I. Hartley. In defense of the eight-point algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:580–591, 1997.

[7] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004.

[8] E. Johns and G.-Z. Yang. From images to scenes: Compressing an image cluster into a single scene model for place recognition. In *Proceedings of the IEEE International Conference on Comptuer Vision*, pages 974–881, 2011.

[9] E. Johns and G.-Z. Yang. Feature co-occurrence maps: Appearance-based localisation throughout the day. In *Proceedings of the International Conference on Robotics and Automation*, pages 3212–3218, 2013.

[10] E. Johns and G.-Z. Yang. Generative methods for long-term place recognition in dynamic scenes. volume 106, pages 297–314, 2013.

[11] E. Johns and G.-Z. Yang. Dynamic scene models for incremental, long-term, appearance-based localisation. In *Proceedings of the International Conference on Robotics and Automation*, pages 2731–2736, 2014.

[12] E. Johns and G.-Z. Yang. Pairwise probabilistic voting: Fast place recognition without RANSAC. In *Proceedings of the European Conference Conference on Computer Vision*, pages 504–519, 2014.

[13] Y. Kalantidis, G. Tolias, Y. Avrithis, M. Phinikettos, E. Spyrou, P. Mylonas, and S. Kollias. VIRaL: Visual image retrieval and localization. *Multimedia Tools and Applications*, 51:555–591, 2011.

[14] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal on Computer Vision*, 60:91–111, 2004.

[15] W. Maddern, A. Stewart, C. McManus, B. Upcroft, W. Churchill, and P. Newman. Illumination invariant imaging: Applications in robust vision-based localisation, mapping and classification for autonomous vehicles. In *Proceedings of the International Conference on Robotics and Automation*, pages 504–519, 2014.

[16] M. J. Milford and G. F. Wyeth. Seqslam: Visual route-based navigation for sunny summer days and stormy winter nights. In *Proceedings of the International Conference on Robotics and Automation*, pages 1643–1649, 2012.

[17] J. Philbin, O. Chum, M. Isard, and A. Z. J. Sivic. Lost in quantization: Improving particular object retrieval in large scale image databases. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–1, 2008.

[18] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–1, 2007.

[19] R. Raguram, O. Chum, M. Pollefeys, J. Matas, and J. Frahm. Usac: A universal framework for random sample consensus. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35:2022–2038, 2013.

[20] R. Raguram and J.-M. Frahm. RECON: Scale-adaptive robust estimation via residual consensus. In *Proceedings of the IEEE International Conference on Computer Vision*, 2011.

[21] R. Raguram, C. Wu, J.-M. Frahm, and S. Lazebnik. Modeling and recognition of landmark image collections using iconic scene graphs. *International Journal of Computer Vision*, 95:213 – 231, 2011.

[22] T. Sattler, B. Leibe, and L. Kobbelt. Scramsac: Improving RANSAC's efficiency with a spatial consistency filter. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2090–2097, 2009.

[23] G. Tolias and Y. Avrithis. Speeded-up, relaxed spatial matching. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1653–1660, 2011.

[24] Y. Zheng, M. Zhao, Y. Song, H. Adam, U. Buddemeier, A. Bissacco, F. Brucher, T.-S. Chua, and H. Neven. Tour the world: Building a web-scale landmark recognition engine. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1085–1092, 2009.