Image Spatial Verification using Segment Intersection of Interest Points

Marisa Bernabeu, Antonio Pertusa and Antonio-Javier Gallego
Departamento de Lenguajes y Sistemas Informáticos,
Universidad de Alicante, San Vicente del Raspeig (Alicante) - 03690, Spain
{mbernabeu, pertusa, jgallego}@dlsi.ua.es

ABSTRACT

This work presents a new spatial verification technique for image similarity search. The proposed algorithm evaluates the geometry of the detected local keypoints by building segments connecting pairs of points and analyzing their intersections in a 2D plane. We show that these intersections remain constant with respect to different geometric transformations (translation, rotation, similarity and affine). Evaluation has been performed obtaining an initial image similarity ranking with a BOF-based methodology, and then using the proposed method for reranking. The presented algorithm (SIIP) has been compared to the RANSAC spatial verification method, showing that it is more efficient and obtains a higher performance on three different datasets.

1 INTRODUCTION

Image similarity search methods aim to obtain a ranking of the most similar images given a query. In general, the goal is to get fast algorithms that are robust to scale, rotation, noise and illumination. A classical methodology to face this task is to detect *interest point* descriptors such as SIFT [6] or SURF [7] from the query image, and match them to those of the images in the dataset. Such mappings can be computed from correspondences of salient regions between the candidate and query images. In some studies these descriptors are clustered into a *bag of features* (BOF) to increase the efficiency for similarity search. One such approach is TOP-SURF [4] which groups the visual features into a histogram obtained by selecting the highest scored visual words (the top T visual words).

However, BOF models lack spatial information. In order to consider it, spatial verification methods can be added to rerank the BOF results to improve the performance with respect to basic representations. This topic have been studied in [2], both in the transformations and in the estimations required to perform the matching.

Several spatial verification methods can be found in the literature. A well known technique is RANSAC [1], which estimates a transformation between a query and a prototype image based on how well its feature locations are predicted by the estimated transformation. RANSAC-based spatial matching has been used in several works [8] for image retrieval. A weak spatial verification alternative is proposed in [5], a fast spatial reranking method is used in [11] with a vocabulary tree, and in [10] the spatial information is obtained using a BOF.

In this work we propose a simple and efficient spatial verification algorithm to rerank the results returned from a BOF method. This technique is based on the comparison of the intersections between segments built by pairs of interest points (keypoints) from two images. The proposed approach is compared to RANSAC, obtaining better results with a smaller computational cost.

Next section describes the presented methodology. Evaluation results are detailed in section 3, followed by the conclusions and future work in section 4.

2 PROPOSED METHOD

Existing spatial verification methods can improve the performance over a basic bag of features representation, but they tend to be computationally expensive. We propose a new method based on the intersection of segments between interest points or *keypoints* that improves the performance of RANSAC with a smaller computational burden.

In a preprocessing stage, given an image query we get the most similar prototypes from a dataset using TOP-SURF [4]. This algorithm extracts the keypoints from the query image, clusters them into a BOF and compares them to the dataset of prototypes, yielding a ranking of the most similar prototypes. Then, we rerank the top K results from this stage with the proposed spatial verification algorithm.

The proposed spatial verification method consists of extracting the keypoints given a descriptor, match them and evaluate the intersections between segments. Figure 1 shows an example to explain the motivation of the

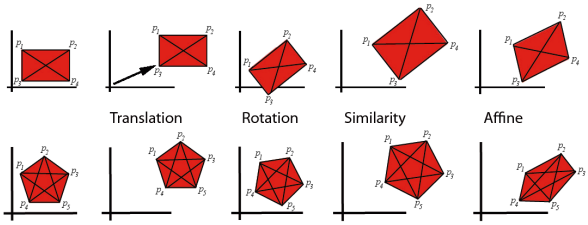

Figure 1: The intersections of segments between interest points are invariant to common transformations.

proposed approach. Given a set P_a of keypoints in an image a, we get all the possible segments and calculate their intersections. As it can be seen, the segment intersections remain invariant to common geometric transformations (scale, translation, rotation, affine, and similarity). In the example of the first row, the segment $\overline{p_1p_4}$ only intersects with the segment $\overline{p_2p_3}$ in all the transformations of the same image.

Using the number of matching intersections as a measure of distance between two images is an efficient method which is robust to 2D transformations since the number of intersection remains constant. The proof of this is trivial since the intersection points belong to the segments, thus the transformation applied to the segments will also be applied to these points.

2.1 Notation and definitions

Consider a given set $P = \{p_1, p_2, \ldots, p_n\}$ of keypoints, where P_a denotes the set of points from the image a, and $\overline{p_1p_2}$ the segment formed by points p_1 and p_2. Let's also denote $\mathscr{P}_2(P)$ as the power set of P with cardinality 2 which contains all the subsets of P with two elements, or in other words, the set of all possible segments between two points in the set P.

The intersections of segments in a set $\mathscr{P}_2(P)$ can be calculated using different methods. In this study, we use an efficient algorithm for line segment intersection from Chazelle et al [12] that has a complexity of $\mathscr{O}(n\log n + k)$, where n is the number of segments in the set, and k the number of intersections.

Given a set of segments $S_a = \{s_1, s_2, \ldots, s_n\}$ belonging to an image a, we define $I = s_1 \cap s_2$ as the intersection function between s_1 and s_2. This function returns whether two segments intersect or not ($\emptyset$). Let also denote I_a as the set containing all the pairs of segments which intersect each other:

$$I_a = \{(s_n, s_m) : s_n, s_m \in S_a, s_n \cap s_m \neq \emptyset, n \neq m\}$$

Spatial verification is performed by comparing the set of intersections. We define the distance $d(a,b)$ between two images a and b as the number of common intersections divided by the maximum number of intersections from both images:

$$d(a,b) = 1 - (|I_a \cap I_b|)/(\max(|I_a|, |I_b|))$$

2.2 Algorithm

Data: Images a, b
Result: Distance $d_{a,b}$
$P_a = \text{SURF}(a)$; $P_b = \text{SURF}(b)$;
$M_{a,b} = \max_N\{(p_a, p_b) : p_a \in P_a, p_b \in P_b,$
$dist(p_a, p_b) < \varepsilon\}$;
for ***each*** *image* i ***in*** a,b **do**
  $P'_i = \{p_i : p_i \in f_i(M_{a,b})\}$;
  $I_i = \{(s_n, s_m) : s_n, s_m \in \mathscr{P}_2(P'_i), s_n \cap s_m \neq \emptyset, n \neq m\}$;
end
$d(a,b) = 1 - (|I_a \cap I_b|)/(\max(|I_a|, |I_b|))$;
return $d(a,b)$

Algorithm 1: Distance between two images a and b.

Algorithm 1 describes the proposed spatial verification method that calculates the distance between two images a and b. We first extract the sets of keypoints P_a and P_b from the input images. Then, these sets are matched to get the subset of related points $M_{a,b}$ that are common to both images. Matching is performed using the Euclidean distance between the feature vectors of each keypoint, defined as $dist(p_a, p_b)$. For efficiency, these sets are ordered by relevance using the inverse of the Euclidean distance between each pair of matched points in order to keep as a maximum only the first (the most correlated) N points to build segments between them.

It is important to note that adding keypoints makes the number of possible intersections to increase. In general, for a set of n segments, there can be up to n^2 intersections in the worst case. This is the reason to keep as a maximum only the N keypoints that are most similar from both images.

From the initial sets of keypoints P_a and P_b, the algorithm selects the subsets of keypoints P'_a and P'_b that are present on the set of corresponding pairs $M_{a,b}$ in order to keep only the matching points for the next stage. Mathematically, we define a bijective function $f_a : M_{a,b} \to P_a$ which given an element of the matching set $M_{a,b}$ returns the corresponding point of the set P_a. Analogously, we define f_b for the set of keypoints P_b.

Then, the segments between all filtered keypoints P'_a and P'_b are independently built for images a and b respectively. Finally we calculate the intersections between these segments in both images, and we define a distance $d(a,b)$ that takes into account the number of common intersections.

For instance, consider the first and last rectangles from the first row in Figure 1, denoted as image a and image b respectively. The interest points p_1, p_2, p_3 and p_4 from image a are matched to the points p_1, p_2, p_3 and p_4 from image b. The segment intersections in a are $\overline{p_1p_3} \cap \overline{p_2p_4}$, that are common to the intersections $\overline{p_1p_3} \cap \overline{p_2p_4}$ in b. Therefore, as there is only one com-

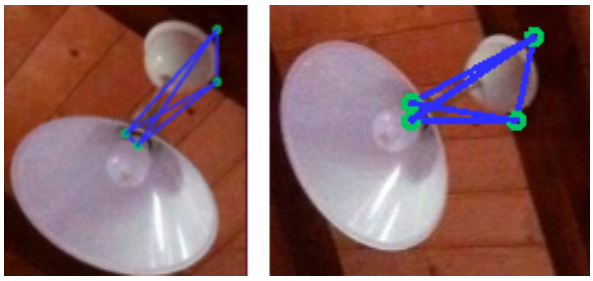

Figure 2: Top interest points and their segments from two images in the MirBot dataset.

mon intersection and each image only contains one intersection, their distance is $d(a,b) = 0$.

Figure 2 shows an example of the segments between matching keypoints of two images. The perspective change is well approximated by a 2D affine transformation and occlusion has little importance.

2.3 Regularization term

The distance $d(a,b)$ was used to perform a fair comparison with RANSAC, which only considers matching keypoints. However, this distance does not take into account the number of unmatched points, which can be a problem in some cases; for example, in Figure 1, if the rectangle is compared to the pentagon, as their geometry is coherent their distance will be 0.

This is the reason to add a regularization term to the original distance in order to consider the proportion between matched and unmatched keypoints. The modified distance $d'(a,b)$ is defined as:

$$d'(a,b) = d(a,b)\frac{|M_{a,b}|}{\max(|P_a|,|P_b|)}$$

3 EVALUATION

This section describes the experimental results. To serve the purpose of comparison, we have evaluated the performance with three datasets commonly used in spatial verification tasks:

Mirbot [3] contains photographs taken by users using smartphones, so they are low-medium quality with variable dimensions (max. 640×640px). The dataset version from October 2014 has been used, with 16327 images distributed in more than 1000 classes.

Oxford 5K [8] contains 5062 high resolution (1024×768px) images divided in 11 classes. Images are assigned to four possible labels: *Good*, *OK*, *Junk* or *Bad*. We have used only the pictures labeled as *Good* or *OK* as positive results, discarding the *Junk* images, and using the *Bad* images as negative results.

Paris [9] contains 6300 high resolution (1024×768px) images collected from Flickr with Paris landmarks. Similarly to Oxford, images are divided in 11 classes. Labels are assigned as in Oxford dataset: *Good, OK, Junk or Bad*. We have also ignored those images labeled as *Junk*.

For the experiments we first use the TOP-SURF method [4] with the default dictionary of 100k words to get a list of the most similar images to a query. Then, only the top K images are taken to be reranked, both using RANSAC and the proposed spatial verification technique for comparison.

In the reranking stage, keypoints could be obtained using any local descriptor. We have chosen SURF [7] to evaluate both RANSAC and the proposed SIIP method.

Performance is assessed using leaving-one-out cross-validation with values of K between 20 and 50. Then the accuracy and the Mean Average Precision at k (MAP@k) are computed. Accuracy (Top-1) measures the ratio between true positives TP and the number of images in the dataset Q:

$$\text{Acc} = \frac{1}{|Q|}\sum_{q\in Q}\text{TP}(q)$$

To calculate the Mean Average Precision at k (MAP@k) we first calculate the Average Precision at k (AP@k) for a query q, and then the MAP is obtained as the mean of the APs for all queries:

$$\text{AP@}k(q) = \frac{1}{N_R}\sum_{n=1}^{N_R} P_k(q)$$

$$\text{MAP@}k(Q) = \frac{1}{|Q|}\sum_{q\in Q}\text{AP@}k(q)$$

where N_R is the minimum between k and the total number of retrieved results, and $P_k(q)$ is the precision at cut-off k in the results list.

3.1 Results

The table 1 shows the results after reranking the $K = 20$ most similar images from the Mirbot dataset with RANSAC and the proposed method (Segment Intersection of Interes Points, SIIP). Different values of N were evaluated to measure the impact on the performance of the maximum number of keypoints, but changing N does not alter significantly the accuracy neither in RANSAC nor with SIIP with the distance function d. A value $N = 24$ has been chosen in the following as it obtains the best MAP@10 and a good accuracy.

The table 2 shows the results obtained in the Mirbot, Oxford 5K and Paris datasets. The accuracy using only TOP-SURF without rerank is shown as the baseline. Then, RANSAC and SIIP (both using SURF features) accuracy and MAP@10 are obtained with $N = 24$. SIIP results are given with the distance function d and also adding the regularization term d'. As it can be seen, SIIP outperforms RANSAC in all the experiments.

		N=8	N=16	N=24	N=32
SIIP (d)	Accuracy	0.3140	0.3148	0.3149	**0.3150**
	MAP@10	0.1782	0.1786	**0.1786**	0.1781
RANSAC	Accuracy	0.3018	0.3017	0.3022	0.3023
	MAP@10	0.1770	0.1774	0.1776	0.1776

Table 1: Results in the MirBot dataset reranking the $K = 20$ first images varying the number of keypoints N.

		Accuracy			MAP@10		
DB	K	RANSAC	SIIP (d)	SIIP (d')	RANSAC	SIIP (d)	SIIP (d')
[3]	20	0.302	0.315	0.305	0.178	**0.179**	0.177
	30	0.304	0.319	0.306	0.177	0.177	0.175
	40	0.305	**0.322**	0.305	0.176	0.176	0.174
	50	0.304	0.319	0.305	0.176	0.175	0.173
[8]	20	0.920	0.923	0.928	0.823	0.826	0.830
	30	0.922	0.925	0.931	0.826	0.829	0.835
	40	0.920	0.925	0.933	0.827	0.831	0.836
	50	0.917	0.925	**0.933**	0.825	0.830	**0.837**
[9]	20	0.796	0.806	0.812	0.621	0.631	0.633
	30	0.800	0.811	0.817	0.629	0.638	0.644
	40	0.803	0.814	0.826	0.631	0.640	0.649
	50	0.803	0.816	**0.828**	0.631	0.643	**0.652**

Table 2: Accuracy and MAP@10 with MirBot [3], Oxford 5K [8] and Paris [9] datasets. The baseline (only TOP-SURF) accuracy without reranking is 0.247 in MirBot, 0.896 in Oxford 5K, and 0.744 in Paris.

In MirBot the function d' did not improve the results of d, but in the other datasets it was consistently better. Contrary to Paris and Oxford, in MirBot each class contains different objects of the same type, instead of the same object from different perspectives. This fact, along with the very large number of classes, explains these accuracy differences.

Besides each image had a very different runtime that depends on the number of keypoints and intersections, SIIP is consistently 3 times faster than RANSAC for all the images, independently of the dataset.

4 CONCLUSIONS

This work presents a new algorithm (SIIP) for image spatial verification based on the comparison of the intersections between segments built by pairs of interest points from two images. It can be used for reranking a subset of images already ranked with a similarity search algorithm such as a BOF.

Evaluation has been performed by obtaining the most K similar images with TOP-SURF, and then reranking the filtered prototypes using SURF interest points. SIIP has been compared to RANSAC, given both superior efficiency (3 times faster) and performance on the three datasets evaluated: Oxford 5K, Paris and MirBot.

As a future work, the proposed methodology could be applied with other descriptors such as SIFT [6], different equations for the distance $d(a,b)$ could be explored, and a comparison with more recent spatial verification methods should be made.

Acknowledgment

This work is supported by the TIMUL project (TIN2013-48152-C2-1-R) and the Instituto Universitario de Investigación en Informática (IUII) from the Universidad de Alicante.

5 REFERENCES

[1] 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, Vol. 24(6):381–395, 1981.

[2] R. Hartley and A. Zisserman. Multiple View Geometry in Computer Vision. Cambridge University Press, 2nd edition, 2004.

[3] A. Pertusa, A.J. Gallego, and M. Bernabeu. MirBot: A multimodal interactive image retrieval system. Lecture Notes in Computer Science, 7887:197–204, 2013.

[4] B. Thomee, E.M. Bakker, and M.S. Lew. TOP-SURF: a visual words toolkit. In 18th ACM Int. Conf. on Multimedia, pp. 1473–1476, 2010.

[5] H. Jégou, M. Douze, and C. Schmid. Improving bag-of-features for large scale image search. Int. Journal of Computer Vision, 87(3), 2010.

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

[7] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-Up Robust Features (SURF). Computer Vision and Image Understanding, 110(3):346–359, 2008.

[8] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. IEEE Conf. on Computer Vision and Pattern Recognition, 2007.

[9] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Lost in Quantization: Improving Particular Object Retrieval in Large Scale Image Databases. IEEE Conf. on Computer Vision and Pattern Recognition, 2008.

[10] S.A. Chatzichristofis, C. Iakovidou, and Y.S. Boutalis. Content Based Image Retrieval Using Visual-Words Distribution Entropy. 5th Int. Conf. MIRAGE, Rocquencourt, pp. 204–215, 2011.

[11] S.S. Tsai, D. Chen, G. Takacs, V. Chandrasekhar, R. Vedantham, R. Grzeszczuk, and B. Girod. Fast geometric re-ranking for image-based retrieval. 17th IEEE Int. Conf. on Image Processing (ICIP), pp. 1029–1032, Sept. 2010.

[12] B. Chazelle and H. Edelsbrunner. An Optimal Algorithm for Intersecting Line Segments in the Plane. Journal of the ACM, 39(1), 1992.