Geometric Indexing for Recognition of Places

Carlos Lara-Alvarez, Alfonso Rojas, and Eduardo Bayro-Corrochano

CINVESTAV Unidad-Guadalajara
Av. del Bosque 1145, Colonia el Bajo,
Zapopan, Jalisco, México
calara@gdl.cinvestav.mx

Abstract. The *Place Recognition* (PR) problem is fundamental for real time applications such as mobile robots (e.g. to detect loop closures) and guidance systems for the visually impaired. The Bag of Words (BoW) is a conventional approach that calculates a histogram of frequencies. One of the disadvantages of the BoW representation is that it loses information about the spatial location of features in the image. In this paper we study an approximate index based on the classic q–gram paradigm to recover images. Similar to the BoW, our approach detects interest points and assigns labels. Each image is represented by a set of q–grams obtained from triangles of a Delaunay decomposition. This representation allows us to create an index and to recover images efficiently. The proposed approach is path independent and was tested with a publicly available dataset showing a high recall rate and reduced time complexity.

Keywords: Place recognition, Bag of Words, Geometric indexing.

1 Introduction

Visual place recognition is helpful and sometimes is the only alternative for GPS-denied areas. Many approaches estimate the location of the robot from a sequence of images instead of a single image. For example, [1] compares short sequences of images to recognize coherent sequences; showing good results even when the perceptual change in the datasets is extreme. FAB-MAP [2] defines a probabilistic model; an image is represented by a binary vector indicating whether a word is present in the image or not. Analyzing sequences of images is powerful but for some applications it is inadequate since the camera does not necessarily follow the same path.

The place recognition problem can also be stated as an *image retrieval system*: the query image is the current sensory measurement, and the database contains measurements recorded along the robot trajectory. Many representations have been proposed but the standard way of representing an image is by using interest points –so called keypoints– . Ideal keypoints must be invariant to viewpoint changes, illumination and occlusions; keypoint detectors such as SIFT [3], SURF [4], MSER [5] have been successfully used for many applications.

Keypoints and their local descriptors are commonly used within the context of the *Bag-of-Visual Words* (BoVW) framework. The BoVW approach requires

E. Bayro-Corrochano and E. Hancock (Eds.): CIARP 2014, LNCS 8827, pp. 548–555, 2014.

a dictionary that is usually generated by clustering a set of descriptors obtained from a generic–image database. When a new image is analyzed, its keypoints are labeled by finding the most probable cluster in the dictionary. That is, if the dictionary clusters are represented by their centroids then a descriptor point is labeled according to its closest centroid. In order to create a scalable indexing and retrieval scheme, histograms of visual–words frequencies are represented in a more compact way by hashing. The major drawback of this approach is that there are many similar images for large databases; hence a verification step is time consuming.

This paper focuses on solving the PR problem from the image retrieval perspective. Our approach preserves the minimal geometrical information required for an efficient system. Our contributions are (i) based on the Delaunay triangulation of labeled keypoints we define geometric q–grams, and, (ii) we propose an inverted index to quickly locate an image without having to search every image in the dataset.

The rest of this work is organized as follows: Section 2 reviews the most relevant work related to our technique. Section 3 introduces the geometrical q–grams based on Delaunay Triangulation. Section 4 describes the q–gram index (for $q \in \{2, 3\}$), and the indexing and querying process. Section 5 presents the methodology and results of the experimental evaluation. Finally, Section 6 concludes this work and presents some ideas for future work.

2 Related Work

There are several approaches based on the Delaunay triangulation (DT) for pattern recognition. For example, to find affine transformations of point patterns, [6] uses the largest maximal clique of the consistency graph for each triangle to obtain the largest set of mutually consistent point pairs. Hence, it allows additions and deletions of points and some random perturbations in their relative locations. Instead of the RANSAC approach to find keypoints correspondences, [7] proposes to analyze DT of keypoints to detect robust matches even for large viewpoint changes. For fingerprint identification [8] creates an index based on DT; invariants of side lengths and angles of the minutiae triangulation are calculated to create an index that reduces memory requirements without sacrificing recognition accuracy.

In many problems of pattern recognition, objects in an image may be efficiently represented by a set of labeled points. The min–Hash algorithm [9] describes a small image patch by selecting independently visual words as global descriptors. Unlike the bag-of-words approach, min–Hashing algorithms combine visual appearance (visual words) with semi-local geometric information to find small objects. Often, a label assigned to a keypoint can vary from scene to scene. This variability arises from many sources: image noise, varying scene illumination, instability in the feature detection process and non–affine changes in the measurement regions. To reduce quantization problems [10] maps each visual region to a weighted set of words, allowing the inclusion of features which were

lost in the quantization stage. Our approach does not use a large vocabulary, but it uses combinations of labels obtained from a reduced–size vocabulary aiming to alleviate some feature labeling problems but maintaining a high selectivity.

Image graph representations are common in the literature; for example, [11] uses a graph of interest points clusters and a matrix of commute times between the different nodes of the graph to obtain a description of their relative arrangement that is robust to large intra class variation, which partially preserves the spatial information. In the context of the 3D world applied to the problem of non-rigid shape retrieval in large databases [12] shows that considering pairs of geometric words ("geometric expressions") allows one to create spatially-sensitive bags of features that are more discriminative.

We analyze triangulations of keypoints to describe images. To the best of the authors' knowledge this is the first paper presenting an index for place recognition based on triangulations of labeled keypoints. Our work is inspired by the success of algorithms used for the approximate string matching problem. This problem consists in finding a query string in a larger text allowing a limited number of errors in the matches [13].

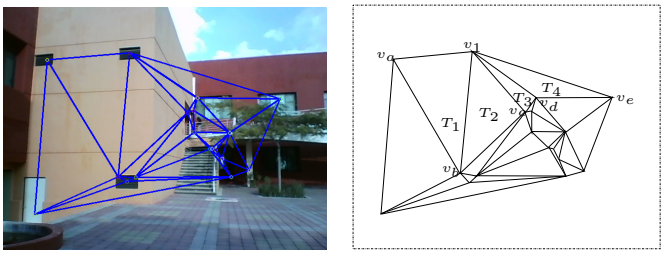
3 Geometrical 2–Grams and 3–Grams from DT

In the context of the approximate text searching, a q–gram is a contiguous sequence of q items from a given sequence. There are many ways of measuring errors between two sequences; the most used is the Levenshtein distance defined as the minimum number of single-character operations (insertion, deletion, substitution) required to change one sequence into the other [13]. Indices based on q–grams are widely used for approximate string matching because they are easily scalable for large databases. Specifically, we use some ideas of the q–gram filters; a q–gram is a subsequence of q symbols from a given sequence. Intuitively, a string containing an approximate match must contain a minimum number of q–grams.

To find similar images in a big dataset, the definition of sequential q–grams must be extended to q–grams in the plane. As sequential q–grams are obtained by analyzing sequences of characters within a text, q–grams in a plane are obtained by analyzing subgraphs of a planar graph. In the following discussion we focus on bigrams and trigrams, for $q > 3$ we could use other subgraphs. For $q \in \{2, 3\}$, the grams can be easily obtained by iterating over the edges (for $q = 2$) or faces (for $q = 3$). Let us review the definition of Delaunay Triangulation:

Definition 1 (Delaunay Triangulation). *Let P be a set of points in the Euclidean plane, with $\mid P \mid \geq 3$. Let us assume that no three points are collinear and that no four points are cocircular. A Delaunay triangulation is a triangulation* $\mathrm{DT}(P)$ *such that no point in P is inside the circumcircle of any Delaunay triangle.*

The *empty circumcircle* property implies that the insertion of a new point in a Delaunay triangulation affects only the triangles whose circumcircles contain



(a) (b)

Fig. 1. Finding 3–grams representation of an image: (a) the DT is calculated from locations of keypoints (b) using the label $l(v)$ of each keypoint v, iterate faces of the DT to find the 3–grams; for example, $t_1 = [l(v_1), l(v_a), l(v_b)]$ is obtained from face T_1, $t_2 = [l(v_1), l(v_b), l(v_c)]$ is obtained from face T_2 and so forth

that point. Hence, noise only affects locally and the DT is stable under single point perturbations [14]. On the other hand, it is known that there exists a unique Delaunay triangulation for the non–degenerate case; this property implies that DT can be used for indexing.

As shown in Figure 1, given an image I we find its q–gram representation $\text{grams}_q(I)$ by four steps:

1. find keypoints and descriptors,
2. assign a label $l(v)$ to the keypoint v based on its descriptor and the available dictionary,
3. calculate a Delaunay Triangulation based on the coordinates of keypoints, and,
4. iterate over edges (or faces) to find 2–grams (or 3–grams).

4 Image Indexing

An *Inverted Index* (II) is a data structure that improves the speed of data retrieval operations on a dataset at the cost of increasing the storage complexity. An image indexing structure is built to support fast access to images previously stored. An II consists of two major components: terms and posting lists. The set of terms, T, consists of all terms $t_j \in T$ seen previously; each term t_j maps to a posting list. Each posting is in the format (I_i, n_i) where n_i is the number of occurrences of the term t_j within image I_i. Figure 2 shows an example of a 3–gram index;

When indexing a place the system simply includes the terms $t \in \text{grams}_q(I)$ and its count in the inverted index. For querying an image, the system recovers the posting list for every $t \in \text{grams}_q(I)$. The union of all images in the posting list produces a set of indexed images that contains at least one term of the query image I. To reduce the search space, we only want to retrieve images that are closely similar to the image query I. We use the Jaccard similarity coefficient

(JC) to measure the similarity of the query image and every recovered image. Given two sets A and B, the Jaccard coefficient is defined as

$$\mathrm{JC}(A,B) = \frac{\mid A \cap B \mid}{\mid A \cup B \mid} \tag{1}$$

The JC is 0 when the two sets are disjoint, 1 when they are equal, and strictly between 0 and 1 otherwise. It is a commonly used indicator of the similarity between two sets: two sets are more similar when their Jaccard index is closer to one, and more dissimilar when their Jaccard index is closer to 0. A multiset $\underline{U}$ is defined as a 2-tuple (U, m) where U is some set and $m : U \to \mathbb{N}_{\geq 1}$ is a function from U to the set $\mathbb{N}_{\geq 1} = \{1, 2, 3, \ldots\}$ of positive natural numbers representing the number of occurrences of elements of U in the multiset $\underline{U}$.

Analogously to (1); for two multisets $\underline{U} : (U, m_U), \underline{V} : (V, m_V)$, the Jaccard coefficient can be stated as

$$\mathrm{JC}(\underline{U},\underline{V}) = \frac{\sum_{e \in U \cap V} \min\{m_U(e), m_V(e)\}}{\sum_{e \in U \cup V} \max\{m_U(e), m_V(e)\}} \tag{2}$$

After the query, we have some images and their corresponding multisets that represent the q–grams common with the query. Then equation 2 is used to calculate the similarity between the query and each recovered image. An image is considered a putative match when the Jaccard coefficient is bigger than a given threshold value. The output of the query is a list of images ranked in decreasing order of Jaccard Index. Finally, a spatial validation step is performed in order to find the set of images that match the query.

term	posting list
$[0,0,1]$	$(I_1, 1)$; $(I_2, 2)$
$[0,0,2]$	$(I_1, 2)$; $(I_2, 1)$
$[2,4,8]$	$(I_2, 1)$
$[5,6,7]$	$(I_2, 1)$

Fig. 2. An inverted index for two images I_1 and I_2 represented by 3–grams $[[0,0,1],[0,0,2],[0,0,2]]$ and $[[0,0,1],[0,0,1],[0,0,2],[2,4,8],[5,6,7]]$, respectively

Spatial Verification

We apply the verification to small sets of images obtained by querying the index, this step improves the precision. For each image we check geometric consistency with the current observation by means of RANSAC [15]. Candidate interest point correspondences are derived from the vertices used to create the geometrical q–grams, hence they are already computed. The spatial verification is applied to those images returned from the index that have a minimum Jaccard Coefficient.

5 Experimental Evaluation

To evaluate the performance of our approach we use the New College Dataset [2]. This dataset was obtained by a mobile robot traversing a complex trajectory of 1.9 km with multiple loop closures. This dataset contains images of the left and right camera of the robot. Since many applications only use a single camera, we only use images obtained by the left camera. The ground truth is provided by the authors of [2] and this information was verified by visual inspection. A Gaussian filter with $\sigma = 9$ was included in the preprocessing step. For each image we find SURF keypoints with 2 octaves and 1 layer and select those features within the second octave. We obtain the Delaunay triangulation for the keypoints and use their labels to find representations based on 2 and 3–grams. To find the label of each vertex we use a dictionary of 256 words obtained from images of random city locations.

Figure 3 shows some images with partial match but correctly found by our approach. Note that Figures 3a and 3b cannot be matched by path–dependent techniques because they follow different paths.

(a) (b) (c) (d)

Fig. 3. Example scenes with partial match but correctly found by Del–Map from the New College dataset: (a) and (b) Occlusion by pedestrian, (c) and (d) Different point of view

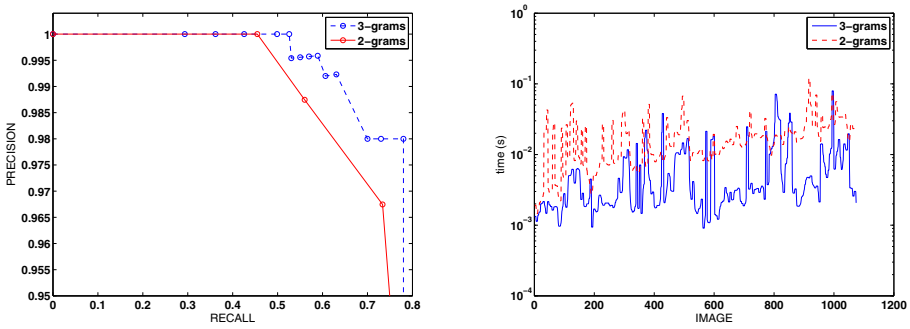
We obtain the precision–recall curve shown in Figure 4(a) by varying the minimum Jaccard coefficient. For many applications –including loop closure– the precision must be 100%; for this precision, the best performance was obtained by the Del–Map algorithm based on 3–grams with a recall of 0.55 (table 1). Figure 4(b) shows the timing results for each image in the route. For the purpose of comparison, timing results exclude the time required for detection of the keypoints.

Discussion

The results presented in the Table 1 demonstrate that the Del–Map algorithm achieves a high recall. Our strategy of using a reduced–size dictionary to alleviate some feature labeling problems gives good performance as shown in the experimental results. In recognizing places, the 3–gram representation outperforms the 2–gram representation. We argue that this result is an effect of the better distinctiveness of the 3–grams; hence, the index recovers fewer places but

Table 1. Comparison of the proposed approach against FAB–Map. Timing results are for a 1.7 GHZ Intel Core i5.

Algorithm	Del–Map		FAB Map
	2–grams	3–grams	
Precision	100 %	100 %	100 %
Recall	45.5 %	55.01 %	47 %
Run Time (ms/place)	12	2.4	1.94
path dependent	NO		YES



(a) (b)

Fig. 4. (a) Precision–Recall curve for the proposed approach (b) Elapsed time for each place

the recognition is more precise. and because the verification step is costly, the time complexity is also reduced.

Another advantage of Del–Map is that it could detect places even when there exists occlusions or a partial view of the indexed scene. Del–Map could recover images obtained from the same place even when the camera does not follow the same path. A path–independent algorithm such as the proposed one can be an advantage for mobile robots moving in real environments, and other applications where the camera does not follow the same path. Finally, Figure 4b shows that the time complexity grows linearly when the number of indexed images increases. This bounded complexity may be an advantage for robots that move in larger environments.

6 Conclusions and Future Work

We show that composed representations obtained from a DT of labeled keypoints are a good choice to solve place recognition problems because they give high selectivity. An important research question about the n–gram index is: which is the optimum value of n to improve the performance of the proposed

algorithm without reducing the recall rate? This issue will be tackled in future work; we plan to use *Fan Graphs* for generating q–grams ($q > 3$) that represent images because they allow redundancy. Although we use SURF in this work, we are working on keypoints that are more robust to illumination changes, we are working on replacing the current verification step for one based on the DT such as the one proposed in [7].

References

1. Milford, M., Wyeth, G.: SeqSLAM: visual route-based navigation for sunny summer days and stormy winter nights. In: Papanikolopoulos, N. (ed.) IEEE International Conferece on Robotics and Automation (ICRA 2012), pp. 1643–1649. IEEE, River Centre (2012)
2. Cummins, M., Newman, P.: FAB-MAP: Probabilistic Localization and Mapping in the Space of Appearance. The International Journal of Robotics Research 27(6), 647–665 (2008)
3. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60, 91–110 (2004)
4. Bay, H., Tuytelaars, T., Van Gool, L.: SURF: Speeded up robust features. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006, Part I. LNCS, vol. 3951, pp. 404–417. Springer, Heidelberg (2006)
5. Matas, J., Chum, O., Urban, M., Pajdla, T.: Robust wide baseline stereo from maximally stable extremal regions. In: British Machine Vision Conference, pp. 384–393 (2002)
6. Ogawa, H.: Labeled point pattern matching by delaunay triangulation and maximal cliques. Pattern Recognition 19(1), 35–40 (1986)
7. Dou, J., Li, J.: Robust image matching based on SIFT and delaunay triangulation. Chinese Optics Letters 10, S11001 (2012)
8. Bebis, G.: Fingerprint identification using delaunay triangulation. In: Proceedings of the International Conference on Information Intelligence and Systems, pp. 452–459 (1999)
9. Chum, O., Perdoch, M., Matas, J.: Geometric min-hashing: Finding a (thick) needle in a haystack. In: CVPR, pp. 17–24. IEEE (2009)
10. Philbin, J., Isard, M., Sivic, J., Zisserman, A.: Lost in quantization: Improving particular object retrieval in large scale image databases. In: CVPR (2008)
11. Behmo, R., Paragios, N., Prinet, V.: Graph commute times for image representation. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2008)
12. Bronstein, A., Bronstein, M., Ovsjanikov, M., Guibas, L.: Shape google: geometric words and expressions for invariant shape retrieval. ACM Trans. Graphics (TOG) 30/1, 1–20 (2011)
13. Navarro, G.: A guided tour to approximate string matching. ACM Comput. Surv. 33(1), 31–88 (2001)
14. Tuceryan, M., Chorzempa, T.: Relative sensitivity of a family of closest-point graphs in computer vision applications. Pattern Recognition 24(5), 361–373 (1991)
15. Fischler, M.A., Bolles, R.C.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. ACM Commun. 24(6), 381–395 (1981)