Optimizing Image Registration for Interactive Applications

Riccardo Gasparini, Stefano Alletto, Giuseppe Serra and Rita Cucchiara

University of Modena and Reggio Emilia, Italy

Abstract. With the spread of wearable and mobile devices, the request for interactive augmented reality applications is in constant growth. Among the different possibilities, we focus on the cultural heritage domain where a key step in the development applications for augmented cultural experiences is to obtain a precise localization of the user, i.e. the 6 degree-of-freedom of the camera acquiring the images used by the application. Current state of the art perform this task by extracting local descriptors from a query and exhaustively matching them to a sparse 3D model of the environment. While this procedure obtains good localization performance, due to the vast search space involved in the retrieval of 2D-3D correspondences this is often not feasible in real-time and interactive environments. In this paper we hence propose to perform descriptor quantization to reduce the search space and employ multiple KD-Trees combined with a principal component analysis dimensionality reduction to enable an efficient search. We experimentally show that our solution can halve the computational requirements of the correspondence search with regard to the state of the art while maintaining similar accuracy levels.

1 Introduction

Augmented user experiences in the cultural heritage domain are in increasing demand by the new digital native tourists of 21st century. With the widespread of wearable and mobile devices, modern tourists are inherently equipped with the hardware needed to provide multimedia and augmented reality content, and while in the academic community a renewed focus rises toward this problem, consumer market is increasing its demand for fast and accurate algorithms.

In this paper, we focus on the task of obtaining precise user localization from a query image (i.e., a photo acquired by the user) with respect to a known environment. For example, a tourist visiting a cultural heritage building could acquire an image using his smartphone or a wearable camera, send it to a processing server that elaborates it registering it to a pre-built Structure from Motion (SfM) 3D model of the scene and returns the localization. A precise (6 degree-of-freedom) localization is a key step in augmented reality applications [2, 5, 21], furthermore the process of localizing an image on a 3D model can be used to propagate on the image any annotation present on the 3D point cloud such as

the presence of relevant architectural details that may be subsequently browsed by the user on a screen [1].

While the registration of a query image to a SfM model has been tackled in the past [20, 19, 13], it is usually approached from an algorithmic point of view, e.g. focusing on large-scale datasets or trying to solve issues like matching under different viewpoints and lighting conditions. Many of these works disregard execution times in favor of high accuracies under challenging conditions, but in order to be able to build interactive consumer applications based on the localization on a 3D model the correspondence search between 2D features and 3D points has to be performed in semi-real time. The standard pipeline employed in the registration of a query image is based on the extraction of local discriminative features (e.g. SIFT keypoints) from the query image and the search for correspondences of these keypoints into the 3D point cloud. This search is often addressed as a nearest neighbor problem and the most popular data structure employed to speed up this phase is the KD-Tree. Once 2D-3D correspondences are enstablished, the query is localized using standard Perspective-n-Point methods [6].

In this work, we propose to intervene on the correspondence search aiming at speeding up the most onerous component of the registration pipeline. In particular, we perform initial experiments exploiting the heavy parallelism of modern multi-core CPUs and GPGPUs. These experiments, highlighting the parallelization difficulties inherent to the structure of the problem, show that approaching the problem from an hardware point of view may not be sufficient to achieve the interactive requirements of an end-user application. Hence, we propose to address the main limitation of the KD-Tree data structure, i.e. its poor scalability under high dimensional data points. By reducing the dimensionality of the data points before constructing the KD-Tree, we can achieve matching times under half a second, which are suitable for interactive cultural heritage applications.

2 Related Work

Large-scale image-based localization is often treated as an image retrieval task, where a query image is matched with a database of geo-localized images [20, 19, 13]. Schindler et al. [20] present a method for city scale localization based on the bag of visual words representation using street side images. Hay and Efros [8] propose an approach that is able to extract coarse geographical information from a given image exploring a set of Flickr images. Recently some approaches have started addressing the problem of severe changes in lighting conditions between reference images and query. Hauagge et al. [7] propose the use of outdoor illumination models for estimating appearance and timestamps from a large set of images of an outdoor scene. Torri et al. [24] propose a place recognition method that combines synthesis of new virtual views with densely sampled image descriptors. After finding the images most similar to a query, the localization is determined using the GPS localization of those images. In most of the cases, the results are evaluated in terms of registration performance or matching quality,

but execution times or scalability are rarely considered. Furthermore, these approaches have shown high performance in scenarios of very large-scale datasets; however, the major problem is that the achieved localization accuracy cannot be better than the precision of the GPS position of the images in the database, which is not suitable for augmented reality applications.

To obtain a better localization accuracy, various techniques have explored the use of the 3D structure of the environment [18, 14]. The significant progress achieved in Structure from Motion (SfM) makes it possible to build models on a city-scale. Essentially, localization is obtained by identifying correspondences between 2D local features in the query image and 3D points of the point-cloud. A common strategy to address this matching problem is to use local-invariant features, such as SIFT [15]. In [14], the authors deal with the task of registering images to multiple 3D point clouds, possibly modeling places in different parts of the world. To address the ambiguity in the nearest neighbor search resulting from having millions of 3D points, they propose to employ co-occurrence priors and incorporate them into the RANSAC loop, effectively sampling hypotheses in a statistically significant way. Once 2D-3D correspondences are identified and filtered from outliers, a standard 3-point pose solver is used to compute the camera location. Sattler et al. [18] propose to exploit the advantages of direct 2D-3D matching by creating a codebook of SIFT visual words and use it to limit the search space. Instead of matching every descriptor in the query image to the entire space of the point cloud, they approximate this search by clustering descriptor into visual words and, once a query descriptor is identified as belonging to a cluster, exhaustive linear search is performed inside the cluster to retrieve the exact correspondence.

3 Proposed Method

Structure from Motion (SfM) techniques aim at building a 3D point cloud of a scene from an unordered set of images exploiting keypoint correspondences (e.g. SIFT features) between pictures capturing similar viewpoints of the same object [22, 4, 23, 17]. After finding a set of geometrically consistent matches, their projecting points into the 3D space and the camera parameters (intrinsic and extrinsic, i.e. focal length, location and orientation) are jointly estimated running the Bundle Adjustment (BA) algorithm [23]. Iteratively, after each new image is matched and added to the point cloud, the algorithm minimizes the sum of the distances between the projection of the 3D points and their relative 2D keypoints (i.e. the reprojection error). This allows to use non-linear least squares solvers reducing the risk of incurring in local minima, an otherwise significant issue when trying to find the minimum cost in a single run. Figure 1 shows an example of 3D model obtained using the aforementioned process.

While using SfM to recover the 3D structure of an environment provides accuracies often used as ground-truth for other methods [12], performing the feature extraction, the exhaustive matching and the iterative BA often requires computation times beyond any real-time applications. Hence, given a pre-built

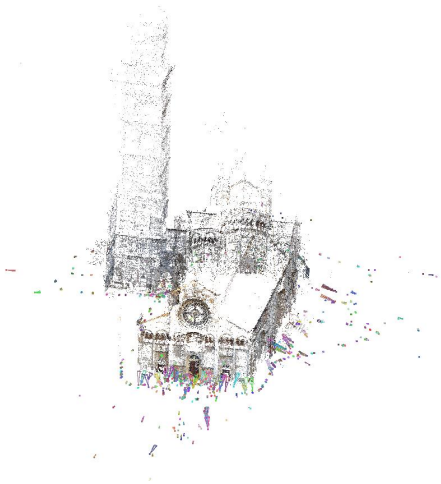

Fig. 1. 3D sparse reconstruction using image acquired by wearable cameras.

3D model, we analyze how to perform the registration of a single image under the constraints of a cultural heritage interactive applications, in which the user can only wait for a few seconds of processing time.

3.1 Image registration under time constraints

The task of registering or localizing a single image to a pre-built 3D point cloud has been recently addressed in literature [18, 20, 19, 13]. In particular, the standard pipeline used to obtain precise localization of a query image is the following. First, SIFT keypoints are extracted from it: the usage of this descriptor allows to match with the 3D points of the model, where every point is described with the list of SIFT descriptors that matched to produce that point in the SfM phase. To obtain a set of 2D-3D correspondences, the SIFT features on the query image are matched with the 3D point cloud. A match is considered correct if the distance ratio (distance from the closest neighbor to the distance of the second closest) is lower than a fix threshold (often set at 0.8). Once that 2D-3D correspondences are established, Perspective-n-Point (PnP) algorithm can be used to retrieve the extrinsic camera parameters [6, 9]. In particular, the PnP algorithm finds the 6 degrees-of-freedom (DoF) pose of a calibrated camera by exploiting the projective model:

$$\mathbf{p} = \mathtt{K}[\mathtt{R}|\mathbf{t}]\mathbf{P} \tag{1}$$

where $\mathbf{p}$ is the 2D point on the image place (in homogeneous coordinates), $\mathbf{P}$ is the 3D point in the world reference system and $\mathtt{K}, \mathtt{R}$ and $\mathbf{t}$ are respectively the intrisic parameters, rotarion and translation matrices. Since the presence

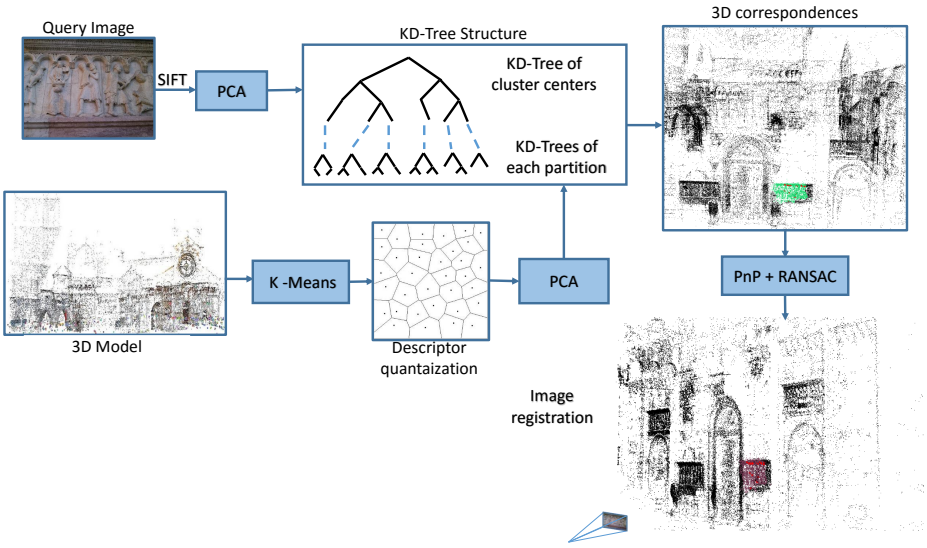

Fig. 2. Summarization of the proposed approach.

of possible outliers in the correspondences can compromise the results of the PnP algorithm, its execution is often enclosed in a RANSAC loop formulating transformation hypotheses and minimizing the reprojection error that such hypotheses result in.

Matching the 2D descriptors of a query that potentially features several thousands keypoints to a 3D point cloud composed of hundred of thousands of points requires some strategy to reduce the search space in order to be computationally feasible. In fact, our experiments showed that, despite the recent advancements in hardware quality, approaching the problem in terms of multi (CPU) or many-core (GPU) parallelization is not sufficient to satisfy the requirements of an interactive application (see Section 4).

To achieve fast 2D-3D correspondences, we follow the strategy adopted in [18] that builds a vocabulary of visual words quantizing SIFT descriptors. First of all, a visual vocabulary is obtained through vector quantization of the local SIFT features of the 3D point cloud. The visual vocabulary is generated by clustering the keypoints in the feature space using the k-means algorithm and Euclidean distance as the clustering metric (we empirically fix the number of visual words to 1000).

Given a query image, the correspondence search can be performed in two steps: first, a similarity search assigning the descriptor to a centroid in the cluster, followed by the retrieval of the best correspondence among the descriptors assigned to the selected visual word. The process of matching a descriptor with both the database of visual words and the subsequent search of the nearest de-

scriptor can be performed through the use of a KD-Tree to improve the search speed.

In fact, the use of tree structures such as KD-Trees is a popular choice when trying to reduce the workload of this phase [3]. KD-Trees are space-partitioning binary trees designed to organize k-dimensional data points. At each non-leaf node, a splitting hyperplane is generated dividing the search space in two half-spaces. Since every node in the tree is associated with one of the k dimensions, at each level an hyperplane perpendicular to that dimension is chosen. Hence, to build a balanced tree, the number of required data points n increases exponentially with the dimension k of such points. If the property $n >> k$ is not satisfied, the search based on a KD-Tree degenerates to an $O(n)$ linear search instead of the $O(\log n)$ property of a binary tree.

To overcome this issue of KD-Trees, where in practice given the dimension of the 3D point cloud of a structure the theoretical average search cost is never reached, we propose to reduce the dimensionality of the quantized descriptors via Principal Component Analysis (PCA) [10]. PCA is a statistical approach that, through orthogonal transformations maps a set of correlated data points into a lower dimensional space where the resulting variables are linearly independent. Recent literature demonstrated that the use of PCA to reduce the dimensionality of local SIFT descriptors allows to obtain compact and informative feature vectors [11, 16].

Applying dimensionality reduction to both the first level KD-Tree (cluster centroids) and the second level KD-Trees (one for each visual word) allows to exploit the advantages of quantizing the descriptors while jointly reducing the dimensionality of the data points and improve the search performance. Differently from [18], the usage of dimensionality reduction techniques has a major impact on the computational time while still resulting in a similar number of valid matches. In fact, given the size of the 3D model employed in our experiments, the high dimensionality of the SIFT descriptor is not ideal and the improvements tied to the use of a KD-Tree with respect to the Brute Force approach are otherwise negligible. Figure 2 summarizes the proposed solution.

4 Experimental Evaluation

To evaluate the impact on both performance and matching quality of our search strategy, we adopt the publicly available dataset presented in [1]. This dataset, which revolves around the romanic Cathedral of Modena, features 743 images captured from different points of view and in unconstrained lighting and weather conditions. These images have been employed, using the structure from motion tool presented in [25], to build a sparse 3D model of the structure. The 3D point cloud contains more that 220.000 points obtained from more than 1.000.000 SIFT correspondences and it is in line with models built in recent state of the art techniques [23, 12]. All the presented results were performed on an Intel Core2 Quad q9550 2.83GHz CPU and a Nvidia GeForce GTX660 2GB GDDR5 GPU.

Table 1. Execution times of retrieving the 2D-3D matches of a query image with 2000 SIFT keypoints. The number of 3D Points have been subsampled from the complete point cloud and different sampling sizes are reported. The methods reported are: Brute Force (BF) on CPU (multi-core), BF on GPU, KD-Tree on CPU, KD-Tree on GPU.

Method	3D Points	Time(s)
BF	10^3	0.083
	10^4	0.770
	10^5	7.472
KD-Tree	10^3	0.911
	10^4	8.467
	10^5	83.885
BF-GPU	10^3	0.532
	10^4	6.493
	10^5	76.896
KD-GPU	10^3	0.699
	10^4	8.944
	10^5	104.854

Given the recent improvements achieved by hardware producers, the first option evaluated to improve the performance of the 2D-3D matching is to exploit multi-core parallelization in the algorithms. In particular, we present an evaluation of the parallelization of the two main exhaustive strategies in 2D-3D matching: the brute-force approach and the nearest neighbor search based on KD-Trees. Table 1 reports the results exploiting both CPU and GPU parallelization. It can be noticed that due to the high dimensionality of SIFT descriptors, the adoption of a KD-Tree to speed-up the search for a match does not provide the expected improvement. Moreover, given the nature of the problem, where both the query data and the KD-Tree must be transferred from the local memory to the GPU, adopting a GPU does not provide any improvement with respect to the brute force strategy. On the contrary, since due to the memory requirements of the search it is not possible to transfer all the required data together to the GPU memory, it can be seen how the communication time dominates the actual computation and results in an overall execution time worst than running on CPU. Based on these results, it clearly appears how tackling the problem from an hardware point of view by parallelizing existing exhaustive search algorithms is not sufficient to achieve the required performance for interactive applications.

To evaluate the impact in performance and accuracy of our solution we compare it with a similar approach proposed in [18] that performs an approximative 2D-3D matching strategy. To perform a fair comparison for both methods we use the same experimental setting (same query images, local features, clustering procedure and 3D model). The results are reported considering queries with an average of 2500 SIFT keypoints and the full 3D point cloud. Table 2 shows the results in terms of required time and matching performance. In particular, we evaluate the impact of the reduction of the SIFT vector dimensionality through PCA. The objective of this dimensionality reduction is to scale the SIFT feature

Table 2. Comparison between the approach proposed by Sattler et al. [18] and the presented solution with different dimensionality reduction.

Method	Dimensionality	Time (s)	Valid Matches
Sattler et al. [18]	128	1.2538	22.6
Our approach	64	0.6690	22.2
	32	0.3996	19.2
	16	0.2634	7.8
	8	0.1766	6.4

vector to a dimensionality that enables the use of balanced KD-Trees. In fact, given the amount of 3D points in our dataset, obtaining a balanced KD-Tree from 128-d descriptor is not possible since it would theoretically require 2^{128} points. It can be noticed the lower the number of considered principal components is, the fastest the method gets at the price of a lower matching performance. In particular, the results show that depending on the desired accuracy and the particular interaction requirements the dimensionality of the resulting vector can be adjusted to either speed up the method or increase its accuracy. For example, reducing the descriptor dimensionality from 128 to 64 halves the required time to perform the search using the KD-Tree while keeping the number of resulting matches substantially unchanged.

Since, in general, the minimum number of required 2D-3D correspondences needed by PnP methods is 4, it can be noticed how dividing the descriptor dimensionality by a factor of 4 still produces sufficient correspondences to robustly (i.e., in a RANSAC loop) perform the query registration. Thanks to the reduction to 32-d vectors, the matching process takes in average less than 0.4 seconds which is an acceptable value for an interactive application. Dimensionality reduction below 32, while still providing significant speed-up and achieving an average of more than 4 matches, is not guaranteed to enable the execution of multiple RANSAC iterations and should therefore be discarded.

5 Conclusions

In this paper we presented an analysis of the standard image registration pipeline, focused on finding the bottlenecks of the procedure and provide a solution that enables the use of registration techniques when requiring interactive response times. In particular, we experimentally showed that exploiting hardware parallelism does not result into improvements. We hence proposed to cluster the descriptors in the 3D point cloud to reduce the search space approximating the results. Through the use of KD-Trees, both the search for the nearest visual word and the retrieval of the closest descriptor belonging to the given visual word are speeded up. To overcome the most critical issue of KD-Trees, i.e. their inability to effectively deal with high dimensional data points, we performed principal component analysis and reduced the resulting descriptor to a dimensionality more suitable. The experimental evaluation on a public benchmark dataset showed

that our solution is capable of obtaining a significant speed-up compared to a state of the art approach while maintaining similar levels of matching quality, a key step in the design of interactive augmented reality applications.

References

1. Alletto, S., Abati, D., Serra, G., Cucchiara, R.: Exploring architectural details through awearable egocentric vision device. Sensors 16(2) (2016)
2. Arth, C., Wagner, D., Klopschitz, M., Irschara, A., Schmalstieg, D.: Wide area localization on mobile phones. In: Proc. IEEE International Symposium on Mixed and Augmented Reality (2009)
3. Bentley, J.L.: Multidimensional binary search trees used for associative searching. Communications of the ACM 18(9), 509–517 (1975)
4. Brown, M., Lowe, D.G.: Unsupervised 3d object recognition and reconstruction in unordered datasets. In: Proc. of International Conference on 3-D Digital Imaging and Modeling (2005)
5. Castle, R., Klein, G., Murray, D.W.: Video-rate localization in multiple maps for wearable augmented reality. In: Proc. of IEEE International Symposium on Wearable Computers (2008)
6. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press, ISBN: 0521540518, second edn. (2004)
7. Hauagge, D., Wehrwein, S., Upchurch, P., Bala, K., Snavely, N.: Reasoning about photo collections using models of outdoor illumination. In: Proc. of British Machine Vision Conference (2014)
8. Hays, J., Efros, A.: Im2gps: estimating geographic information from a single image. In: Proc. of CVPR (2008)
9. Irschara, A., Zach, C., Frahm, J., Bischof, H.: From structure-from-motion point clouds to fast location recognition. In: Proc. of IEEE International Conference on Computer Vision and Pattern Recognition (2009)
10. Jolliffe, I.: Principal component analysis. Wiley Online Library (2002)
11. Ke, Y., Sukthankar, R.: Pca-sift: A more distinctive representation for local image descriptors. In: Proc. of IEEE International Conference on Computer Vision and Pattern Recognition (2004)
12. Kroeger, T., Van Gool, L.: Video registration to sfm models. In: Proc. of IEEE European Conference on Computer Vision (2014)
13. Li, Y., Snavely, N., Huttenlocher, D., Fua, P.: Worldwide pose estimation using 3d point clouds. In: Proc. of European Conference on Computer Vision (2012)
14. Li, Y., Snavely, N., Huttenlocher, D., Fua, P.: Worldwide pose estimation using 3d point clouds. In: Proc. of IEEE European Conference on Computer Vision (2012)
15. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60(2), 91–110 (2004)
16. Perronnin, F., Sánchez, J., Mensink, T.: Improving the fisher kernel for large-scale image classification. In: Proc. of IEEE European Conference on Computer Vision (2010)
17. Pollefeys, M., Van Gool, L., Vergauwen, M., Verbiest, F., Cornelis, K., Tops, J., Koch, R.: Visual modeling with a hand-held camera. International Journal of Computer Vision 59(3), 207–232 (2004)
18. Sattler, T., Leibe, B., Kobbelt, L.: Fast image-based localization using direct 2d-to-3d matching. In: Proc. of IEEE International Conference on Computer Vision (2011)

19. Sattler, T., Weyand, T., Leibe, B., Kobbelt, L.: Image retrieval for image-based localization revisited. In: Proc. of British Machine Vision Conference (2012)
20. Schindler, G., Brown, M., Szeliski, R.: City-scale location recognition. In: Proc. of IEEE Conference on Computer Vision and Pattern Recognition (2007)
21. Schops, T., Engel, J., Cremers, D.: Semi-dense visual odometry for ar on a smartphone. In: Proc. of IEEE International Symposium on Mixed and Augmented Reality (2014)
22. Snavely, N., Seitz, S.M., Szeliski, R.: Photo tourism: exploring photo collections in 3d. In: ACM Transactions on Graphics. vol. 25, pp. 835–846. ACM (2006)
23. Snavely, N., Seitz, S.M., Szeliski, R.: Modeling the world from internet photo collections. International Journal of Computer Vision 80(2), 189–210 (2008)
24. Torii, A., Arandjelovic, R., Sivic, J., Okutomi, M., Pajdla, T.: 24/7 place recognition by view synthesis. In: Proc. of IEEE Conference on Computer Vision and Pattern Recognition (2015)
25. Wu, C., Agarwal, S., Curless, B., Seitz, S.: Multicore bundle adjustment. In: Proc. of IEEE Conference on Computer Vision and Pattern Recognition (2011)