Geometric-visual descriptor for improved image based localization

Achref Ouni, Eric Royer, Marc Chevaldonné, Michel Dhome

Geometric-visual descriptor for improved image based localization

Achref Ouni[1], Eric Royer[1], Marc Chevaldonné[1] and Michel Dhome[1]

***Abstract*— This paper addresses the problem of image based localization. The goal is to find quickly and accurately the relative pose from a query taken from a stereo camera and a map obtained using visual SLAM which contains poses and 3D points associated to descriptors. In this paper we introduce a new method that leverages the stereo vision by adding geometric information to visual descriptors. This method can be used when the vertical direction of the camera is known (for example on a wheeled robot). This new geometric visual descriptor can be used with several image based localization algorithms based on visual words. We test the approach with different datasets (indoor, outdoor) and we show experimentally that the new geometric-visual descriptor improves standard image based localization approaches.**

I. INTRODUCTION

Image based localization (IBL) is an important task for many applications such as augmented reality [9], autonomous navigation [7], real-time camera pose tracking [13]. However, despite many recent contributions to this problem [5], [10], [2], it's still a challenge to localize an image in large scale environments with illumination changes, etc. Given a query image, the problem consists in retrieving the position and orientation of the camera in a known environment.

More precisely we suppose we have a map coming from visual SLAM which contains poses and 3D points associated to descriptors. In this case the camera pose is traditionally computed from n matches between 3D points from the map and 2D features from an input query. A PnP (Perspective-n-Point) solver is used inside RANSAC (random sample and consensus) to recover the 6 DoF (Degrees of freedom) pose of the query. In this work we tackle the problem of matching features to features by adding geometric information to descriptors. To do this, we make some assumptions on the query image and the 3D map. Our work requires a stereo camera to get the query image and build the 3D map. We assume the vertical direction is known and the height of the cameras is fixed. Our assumptions are valid when the cameras are mounted on a vehicle or a wheeled mobile robot. The benefit of using stereo vision is to triangulate the features so that each point from the query is associated to 3D coordinates in the query camera reference frame. Among the 3 axis only the height (Z-axis) does not depend on the pose of the query. Each key-point is characterized by two elements: its local descriptors and its height. Hence we concatenate them to form a new descriptor combining geometric and visual information to be used in the matching process.

[1] Université Clermont Auvergne, CNRS, SIGMA Clermont, Institut Pascal, F-63000 CLERMONT-FERRAND, FRANCE achref.el.ouni@outlook.fr

978-1-7281-8068-7/20/$31.00

Our approach can be applied to either direct or indirect methods. In both situations, it can improve the state of the art methods as will be shown in the experimental part. This paper has two main contributions: (1) Exploiting the geometric descriptors to generate a robust vocabulary that will be used to build a geometric bag of visual words (G-BoVW), (2) Increasing the efficiency of the matching step (2D-3D). We test our approach on two different datasets. Indoor we test on a mobile robot in a museum and outdoor we test on the Oxford Robotcar Dataset. We show experimentally that the new geometric-visual descriptor improves standard image based localization approaches.

II. STATE OF THE ART

IBL can be addressed either by direct matching or indirect methods. Direct methods directly match the features descriptors between the query image and the 3D scene. Indirect methods are related to the content based image retrieval problem. The idea is to retrieve a set of key images that are similar to a query and match their descriptors in order to obtain n 2D-3D correspondences. Let's discuss first indirect methods. [17] ,proposes an efficient approach that selects the discriminative key-point from candidate images based on a voting system. Using BoVW (Bag of Visual Words) [18], [3] extract the most similar image in the dataset to find an approximate pose of the query. Alternatively the nearest neighbors can be computed using a CNN (Convolutional Neural Network) to generate a global descriptor for image representation. [14] adapt a CNN instead of BoVW in the stage of finding the similar images to an input query by extracting the feature vector from the feature layer (e.g $fc7$ in alexnet). A feature matching step is then used to establish 2D correspondences between the query and the closest images found previously.

On the other hand, direct methods proceed by matching descriptors between the input query and the 3D model built by a SfM or SLAM algorithm. [8]

Sattler et al [16] propose a direct matching method based on visual words to establish the correspondence between the query and the 3D scene. Because an image contains fewer primitives than the whole 3D model, Sattler et al [15] improve their framework by combining the 3Dto 2D and 2D to 3D matching strategies to increase the number of correspondences. In order to reduce the computational cost, [19] proposes a fast outlier rejection algorithm for large scale datasets. A similar work [20] exploits geometric visibility constraints to reject wrong matches with run-time $O(n)$.

In either case PnP is used to retrieve the camera pose, three correspondences are sufficient to recover the pose if we have the intrinsic parameters. This is usually done with

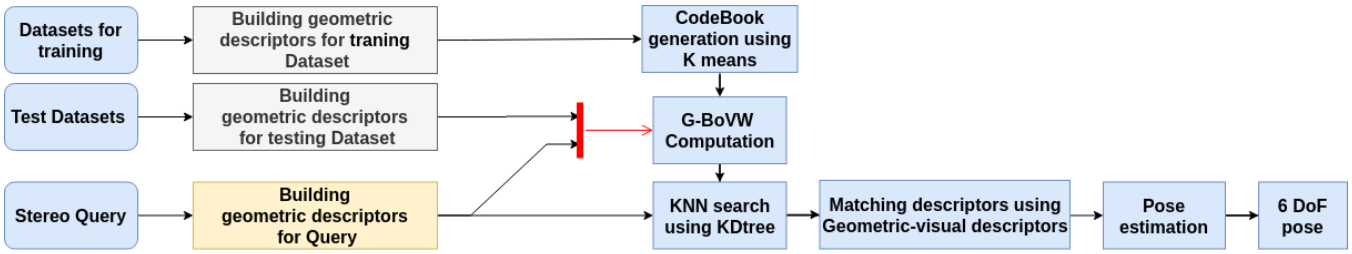

Fig. 1. Global Framework

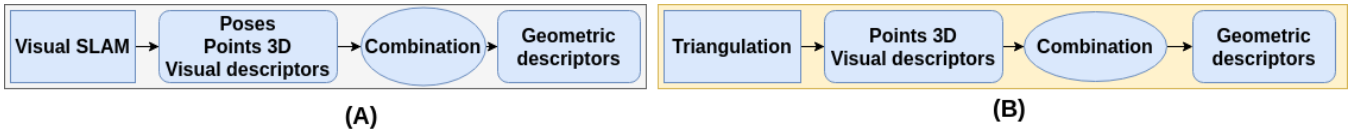

Fig. 2. (A) Building geometric descriptors for Datasets (B) Building geometric descriptors for Query

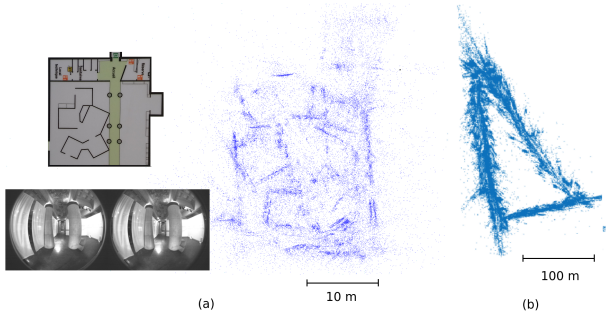

Fig. 3. (a) Plan and 3D map of museum (b) 3D map of Oxford Robotcar

RANSAC to eliminate outliers. Wrong correspondences can occur, especially with repetitive structures in the environment. Lowe [11] use the ratio-test to limit the number of false correspondences. As proposed by [6] to evaluate the results, we consider a query as well registered in the model only if we obtain more than 12 inliers after RANSAC. Our proposed geometric descriptor can be used with most of the state of the art methods and in the experimental part we show results obtained with three different methods: [16], [17] and our own indirect method.

III. The Proposed Method

In this section, we present an indirect method to compute the pose of a query stereo pair. Here, it is necessary to exploit geometric information to handle the large quantity of wrong correspondences and to successfully recover the 6 DoF pose. We tackle the weakness on the matching step by combining the height and the visual descriptors. We make two assumptions: the vertical direction is known and the height of the cameras is fixed. This is the case for example if the camera is mounted on a vehicle or a wheeled robot. In the sequel, the Z axis is vertical.

Figures 1 and 2 present all the steps in our approach. We start by building the 3D map using visual SLAM. Once we extract the visual features from the images (keys and query), we build a geometric visual descriptor. Then, we select the closest key images to the input query using bags of visual words integrating geometric information (G-BoVW) explained in the end of this section. Finally, using the matches between the modified descriptors we compute the relative pose. These steps are detailed in the following and summarized in Algorithm 1.

Algorithm 1 Indirect IBL

Require: Database D, Query Stereo Pair Q, Geometric Visual words GVW, Threshold ϵ
3D-Map, DescriptorsDB $\leftarrow$ Visual-SLAM(D)
XYZ, DescriptorsQuery $\leftarrow$ Triangulation (Q)
$Z_q \leftarrow$ AdaptHeight (XYZ)
$Z_m \leftarrow$ AdaptHeight (3D-map)
GeoDesc$_q \leftarrow$ combine ($Z_q, DescriptorsQuery$)
ListKNN $\leftarrow$ GetNN (GeoDesc$_q, GeoDesc_m, GVW$)
$M = \emptyset$
for `i=1 to ListKNN` **do**
  GeoDesc$_m \leftarrow$ combine ($Z_m, DescriptorsDB_{NNi}$)
  Matches =matchFeatures (GeoDesc$_q, GeoDesc_m, \epsilon$)
  $M = M \cup$ Matches
end for
RANSAC-Inliers=RANSAC-P3P (M)
if RANSAC Inliers ≥ 12 **then**
  return Camera Pose
else
  Pose not found
end if

The benefit of using a stereo pair as a query instead of a single image is to be able to triangulate 3D points in order to obtain more useful information. Each feature from the query stereo pair is then characterized by two elements: its local descriptor (Kaze [1] in our case) and its 3D point in the camera coordinate system. Among the 3 axes only the height (z-axis) does not depend on the pose of the query. So for each 3D point we extract the Z coordinate. Then we concatenate

the Kaze visual descriptor (K_p) with this invariant geometric information in order to obtain a combined descriptor [K_p, Z_n]. Z_n is not the raw Z coordinate but it's computed from Z with a normalization function. The goal is to be more discriminant with features with similar appearances but at different height in the scene.

It's necessary to normalize Z for two reasons: first to minimize the impact of outliers and the second to balance the weight of the visual descriptors and the height. The stereo triangulation produces some outliers with very high or very low height. To fix this problem, we define a grid in the horizontal plane of the slam map and we remove the 10% highest and 10% lowest in each cell in the grid. After the noise elimination step, we need to make the weight of the Z coordinate equal to the weight of the visual descriptor with equation (1) which depends on two principal parameters:

- H_c: The height of the camera above the ground (only required if the height of the cameras is changed between the mapping and the localization step).
- The amplitude of Z values in the 3D map, that is the difference between Z_{max} and Z_{min}, after the noise elimination step.

$$Z_n = \frac{Z + H_c}{Z_{max} - Z_{min}} DimDesc \tag{1}$$

where $DimDesc$ is the size of descriptors (64 for Kaze).

TABLE I
COMPARISON OF OUR APPROACH WITH THE RE-IMPLEMENTED METHODS FROM THE STATE OF THE ART

Indoor dataset(Museum)		
Descriptor	Visual descriptor	Geometric visual descriptor
Fast search [16]	271	312
Active search [17]	281	326
Indirect(Algorithm 1)	191	256
Direct(F2P)	205	257
Outdoor dataset(Oxford)		
Descriptor	Visual descriptor	Geometric visual descriptor
Fast search [16]	626	661
Active search [17]	631	671
Indirect(Algorithm 1)	550	639
Direct(F2P)	607	660

Geometric Bag Of Visual Words(G-BoVW): In the first step, we generate the visual vocabulary from a training dataset. The training dataset is composed of video sequences recorded in the same environment of the MAP. So, we detect and extract the feature from the training dataset, then we apply the visual SLAM algorithm with the aim to assign to the features their corresponding height. After normalizing the height, we collect all the geometric features and we generate a codebook (visual vocabulary) by applying the Kmeans clustering algorithm. Visual words are then defined as the centers of the learned clusters. Given a query image, we triangulate the stereo image for obtaining the corresponding height and descriptors.

For each image in the dataset, we obtain the height of the point from the SLAM Map. Similar to BoVW algorithm, we associate for each feature the nearest visual words using L2 metric then we create the histogram containing the frequency of the words for each image. Finally, the similarity between the query and candidates is measured by the distance between visual words vectors. The G-BoVW are used to compute a list of K images which are the nearest neighbors of the query.
Matching descriptors: We match the descriptors between the query image and the nearest neighbors based on the geometric visual descriptors. We accept a match if the ratio test is true:

$$\frac{||[K_p, Z_n] - [K_{p1}, Z_{n1}]||_2}{||[K_p, Z_n] - [K_{p2}, Z_{n2}]||_2} < \epsilon \tag{2}$$

Where $[K_p, Z_n]$ is the descriptor in the query, $[K_{p1}, Z_{n1}]$ $[K_{p2}, Z_{n2}]$ are the descriptors of the point in the candidate image with respectively the lowest and second lowest euclidean distance. ϵ is the threshold ratio. The found matches are used to compute the relative pose by applying P3P+RANSAC.

IV. EXPERIMENTAL SETUP

Benchmark datasets. We test against two different databases: indoor (museum, Figure 4.a), outdoor (Oxford Robotcar dataset [12], Figure 4.b)[1]. Museum is a dataset composed of 443 query images with a slam point cloud built using 859 key images which generate 61192 3D points and 1.6 millions of descriptors. Oxford Robotcar dataset [12]) is an outdoor dataset with 1000 query images and the slam point cloud is built using 1100 key images which generate 159959 of 3D points and 2.5 millions of descriptors. While Museum has fewer queries than Oxford, it is more challenging due to strongly identical appearances and repeated structures. The evaluation of our approach depends on the number of poses that are successfully computed. We apply RANSAC in combination with P3P on the set of 2D-3D matches found with a reprojection error $\varepsilon \leq 3$ pixels. As proposed in [6] we consider a query as well registered in the model only if we obtain more than 12 inliers after RANSAC.
Implementation details In all experiments we used KAZE [1] descriptors for both the query images and key images. We test the performance of our approach using the list of KNN obtained by content based image retrieval algorithm [4]. In our experiments we set the threshold ratio $\epsilon = 0.7$ in equation (2). The Addition of geometric information (Z) increases the number of registered images. Even if a pose is successfully computed without geometric information, we have more inliers when we use Z. Getting more inliers positively influences the accuracy of the poses. Figures 4,5 present the number of correctly recovered poses when using two lists of Knn : the first list of Knn obtained from BoVW (yellow and orange curves) and the second list

[1] https://robotcar-dataset.robots.ox.ac.uk/datasets/2014-08-11-10-22-21/?fbclid=IwAR2rgyv58BnS6QLiuIayMKsC64mCc3V4TDk1S8RaZV2_o6pcRvqOlvTHmLc

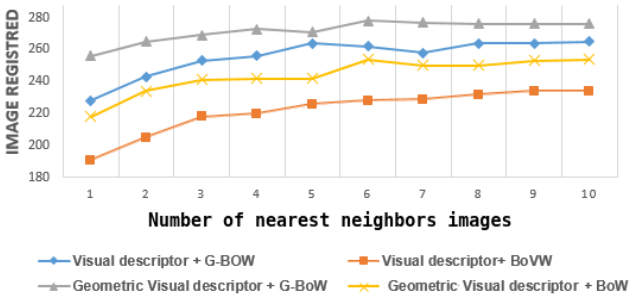

Fig. 4. Quantitative results of our method in a museum (Indoors)

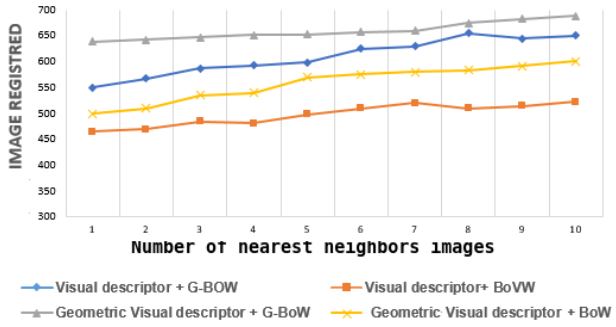

Fig. 5. Quantitative results of our method in oxford datasets(Outdoors)

of Knn obtained from G-BoVW (gray and blue curves). In each case, we repeat the test by changing the size of the nearest neighbors list from 1 to 10. In addition, using the geometric visual descriptors in all cases (Indoors/Outdoors) we have a higher number of poses.

Comparisons with the state of the art In IBL almost all proposed methods are intended for mono camera and evaluated on databases such as Dubrovnik 6k, Rome ... . In our case the query is given by a stereo camera so it's impossible to test on the classical datasets. Therefore, we select from the state of the art two methods who use visual vocabulary to estimate the pose ([16] , [15]) and we test them on the databases presented on the figure 3. For the query images we keep only the features which are triangulated after stereo matching. We compare in Table 1 the number of poses successfully localized. The geometric visual descriptor clearly shows its effectiveness on all the re-implementation and indirect methods.

V. CONCLUSION

We have presented an efficient pose estimation method based on a new geometric visual descriptor: we extract the height of visual features using triangulation on stereo images and we use it to construct the new descriptor by concatenating the height with the visual features. We have presented two main uses of the descriptors in this paper: (i) to improve the BoVW performance (ii) to ameliorate the 2D-3D correspondence search. We have studied the influence of the height Z compared to the Kaze descriptor without Z. The experiments show the advantages and the performance of our new combined descriptor when used in combination with state of the art methods. This approach is very useful for many applications when the cameras are mounted on a vehicle or a mobile robot.

REFERENCES

[1] Pablo Fernández Alcantarilla, Adrien Bartoli, and Andrew J Davison. Kaze features. In *European Conference on Computer Vision*, pages 214–227, 2012.

[2] Ronald Clark, Sen Wang, Andrew Markham, Niki Trigoni, and Hongkai Wen. Vidloc: A deep spatio-temporal model for 6-dof video-clip relocalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 3, 2017.

[3] Mark Cummins and Paul Newman. Appearance-only slam at large scale with fab-map 2.0. *The International Journal of Robotics Research*, 30(9):1100–1123, 2011.

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. Ieee, 2009.

[5] Raúl Dıaz and Charless C Fowlkes. Cluster-wise ratio tests for fast camera localization. 2016.

[6] Arnold Irschara, Christopher Zach, Jan-Michael Frahm, and Horst Bischof. From structure-from-motion point clouds to fast location recognition. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2599–2606. IEEE, 2009.

[7] Johannes Kopf, Boris Neubert, Billy Chen, Michael Cohen, Daniel Cohen-Or, Oliver Deussen, Matt Uyttendaele, and Dani Lischinski. *Deep photo: Model-based photograph enhancement and viewing*, volume 27. ACM, 2008.

[8] Yunpeng Li, Noah Snavely, and Daniel P Huttenlocher. Location recognition using prioritized feature matching. In *European conference on computer vision*, pages 791–804. Springer, 2010.

[9] Haomin Liu, Guofeng Zhang, and Hujun Bao. Robust keyframe-based monocular slam for augmented reality. In *International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 1–10, 2016.

[10] Liu Liu, Hongdong Li, and Yuchao Dai. Efficient global 2d-3d matching for camera localization in a large-scale 3d map. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, pages 2391–2400. IEEE, 2017.

[11] David Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[12] Will Maddern, Geoffrey Pascoe, Chris Linegar, and Paul Newman. 1 year, 1000 km: The oxford robotcar dataset. *The International Journal of Robotics Research*, 36(1):3–15, 2017.

[13] Richard A Newcombe, Steven J Lovegrove, and Andrew J Davison. Dtam: Dense tracking and mapping in real-time. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2320–2327. IEEE, 2011.

[14] Filip Radenović, Giorgos Tolias, and Ondřej Chum. Cnn image retrieval learns from bow: Unsupervised fine-tuning with hard examples. In *European conference on computer vision*, pages 3–20, 2016.

[15] Torsten Sattler, Bastian Leibe, and Leif Kobbelt. Improving image-based localization by active correspondence search. In *European conference on computer vision*, pages 752–765, 2012.

[16] Torsten Sattler, Bastian Leibe, and Leif Kobbelt. Towards fast image-based localization on a city-scale. In *Outdoor and Large-Scale Real-World Scene Analysis*, pages 191–211. Springer, 2012.

[17] Torsten Sattler, Tobias Weyand, Bastian Leibe, and Leif Kobbelt. Image retrieval for image-based localization revisited. In *BMVC*, volume 1, page 4, 2012.

[18] Yafei Song, Xiaowu Chen, Xiaogang Wang, Yu Zhang, and Jia Li. Fast estimation of relative poses for 6-dof image localization. In *2015 IEEE International Conference on Multimedia Big Data (BigMM)*, pages 156–163. IEEE, 2015.

[19] Linus Svarm, Olof Enqvist, Magnus Oskarsson, and Fredrik Kahl. Accurate localization and pose estimation for large 3d models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 532–539, 2014.

[20] Bernhard Zeisl, Torsten Sattler, and Marc Pollefeys. Camera pose voting for large-scale image-based localization. In *International Conference on Computer Vision*, pages 2704–2712, 2015.