Efficient Monocular Pose Estimation for Complex 3D Models

A. Rubio, M. Villamizar, L. Ferraz, A. Penate-Sanchez,
A. Ramisa, E. Simo-Serra, A. Sanfeliu and F. Moreno-Noguer
Institut de Robòtica i Informàtica Industrial, CSIC-UPC
Llorens Artigas 4-6, 08028 Barcelona, Spain

***Abstract*— We propose a robust and efficient method to estimate the pose of a camera with respect to complex 3D textured models of the environment that can potentially contain more than $100,000$ points. To tackle this problem we follow a top down approach where we combine high-level deep network classifiers with low level geometric approaches to come up with a solution that is fast, robust and accurate. Given an input image, we initially use a pre-trained deep network to compute a rough estimation of the camera pose. This initial estimate constrains the number of 3D model points that can be seen from the camera viewpoint. We then establish 3D-to-2D correspondences between these potentially visible points of the model and the 2D detected image features. Accurate pose estimation is finally obtained from the 2D-to-3D correspondences using a novel PnP algorithm that rejects outliers without the need to use a RANSAC strategy, and which is between 10 and 100 times faster than other methods that use it. Two real experiments dealing with very large and complex 3D models demonstrate the effectiveness of the approach.**

I. Introduction

Robust camera localization is a fundamental problem in a wide range of robotics applications, going from precise object manipulation to autonomous vehicle navigation. Despite being a topic researched for decades it is still an open challenge. There exist approaches based on infrared cameras and high-frequency systems such as Vicon [1], which have shown excellent results for localization and navigation of robots. However, these systems are limited to indoor environments where lighting conditions are controlled.

Another alternative to robustly localize the robot is to equip it with multiple sensors, such as lasers or stereo cameras, and then fusing the data from each of them. Although these systems increase the reliability of the robot for self-localization, they have a negative impact on the computational cost and payload. This is specially critical in some robotics tasks where small and low-cost robots (e.g. aerial robots) are frequently used.

In contrast to these multi-sensor approaches, in this work we propose an efficient and robust system based uniquely on a monocular camera and a known pre-computed 3D textured model of the environment, as shown in Fig. 1. Indeed, the proposed method is able to efficiently estimate the full pose (rotation and translation) of the camera within the 3D map, given solely one single input image. No temporal information is used about the previous poses that can constrain the region of the 3D model where the camera is pointing to. While this makes our problem significantly more complex, it makes the resulting pose estimation robust to issues such as drifting, occlusions of the model during short periods of time, or sudden camera motions.

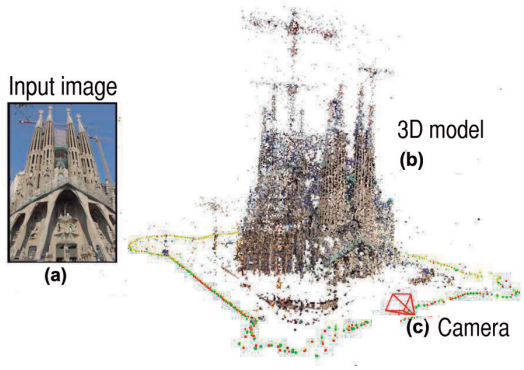


Fig. 1: Problem definition: Given an input image (a) and a known textured 3D model of the environment (b), the problem is to estimate the pose of the camera that captured the image with respect to the model (c). The main challenge addressed in this paper is to perform the correspondence of points efficiently and reliably for complex 3D that contain a large number of points. In this example, the model of the Sagrada Familia has over 100,000 points.

More precisely, let us assume our 3D model is made of n 3D points, each associated to a visual descriptor representing its appearance. Fig. 2 shows the overall scheme of the proposed method. In our case these visual descriptors correspond to SIFT features [12] obtained from a set of training images previously used, in an off-line step, to build the model (Fig. 2(g)). At runtime, the descriptors of the 3D model are compared against the m descriptors extracted from the input image (Fig. 2(d)) in order to determine a first set of 3D-to-2D match candidates, to then estimate the pose (Fig 1(c)). However, solving this correspondence problem has an $\mathcal{O}(n \cdot m)$ complexity, which is extremely costly if we consider that n can be very large (e.g. 100,000 points).

In order to alleviate the computational load, we propose including a preliminary step based on a deep-learning network that yields an approximate initial pose, without the need to explicitly compute correspondences. This method provides the k most similar images (among the training images used to build the 3D model) to the input image (Fig. 2(e)). Then, the descriptors of these images are used to perform the 3D-to-2D matching (Fig. 2(f)). Therefore, the correspondence is done with a small part of the model but not with the complete model. And most importantly, this initial

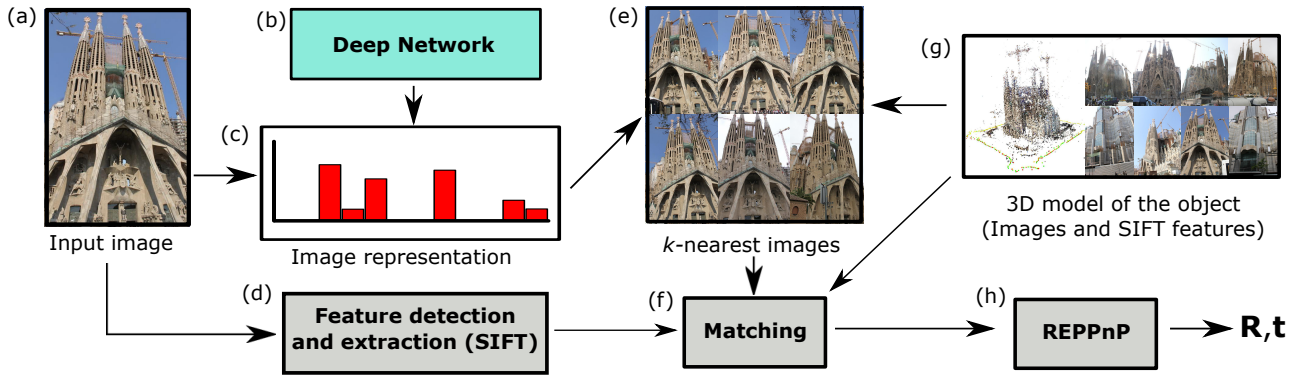


Fig. 2: Overall scheme of the proposed method for efficient camera pose estimation using highly complex models.

pre-selection of most similar images is done in a matter of milliseconds. Since these correspondences may still contain false matches, we get rid of them by using REPP*n*P [4], a novel RANSAC-less algorithm for rejecting outliers. The experiments have shown that the proposed method not only reduces the computation cost but that it also achieves high accuracy rates in highly complex models.

The rest of the paper describes each components of the proposed method. Sec. III explains the construction of the 3D model. Sec. IV describes the initial pose estimation using the deep-learning network and the REPP*n*P algorithm used for the subsequent refinement. In Sec. V the method is extensively evaluated over two different models. Finally, Sec. VI summarizes the main contributions and future work.

II. Related Work

Methods for 3D pose estimation from monocular images can be roughly split into two categories: *Geometric approaches* that rely on local image features (e.g., points) and use geometric relations to compute the pose; and *Appearance-based methods* that compute global descriptors of the image, and then use machine learning approaches to estimate which image within the training set is the closest one to a given input image.

Geometric approaches use local descriptors to estimate 2D-to-3D matches between one input image and one or several reference images registered to a 3D model. PnP algorithms such as EPnP [10], [13] are then used to enforce geometric constraints and solve for the pose parameters. On top of that, robust RANSAC-based strategies [2], [14] can be used both to speed up the matching process and to filter outlier correspondences. Yet, while these methods provide very accurate results, they require both the reference and input images to be of high quality, such that local features can be reliably and repetitively extracted. Additionally, if the number of points is very large, the outlier rejection scheme can become extremely slow. Recently, [23] has shown that priors about the orientation of the camera relative to the ground plane can speed up this process. We do not consider these kind of priors for our method, though.

On the other hand, approaches relying on global descriptions of the image are less sensitive to a precise localization of individual features. These methods typically use a set of training images acquired from different viewpoints to statistically model the spatial relationship of the local features, either using one single detector for all poses [7], [11], [22] or a combination of various pose-specific detectors [15], [16], [24], [26], [27]. Another alternative is to bind image features with poses during training and have them vote in the pose space [6]. The limitation of these approaches is that the estimated pose tends to be innacurate, and highly depends on the spatial resolution at which the training images have been acquired. The highest granularity of the training set, the more precise can be the estimated pose, although the resulting detector is more prone to give false positives.

In this paper we combine the best of both worlds. On the one hand, we will use a global descriptor based on a deep convolutional network to get a first estimation of the pose. Deep networks have recently shown impressive results in image classification tasks [9]. This initial estimate will in its turn reduce the number of potential 2D-to-3D matches, and make geometric approaches applicable. On the geometric side, we will make use of a very recent PnP approach, which inherently incorporates an outlier rejection scheme without the need to run RANSAC [5]. The combinations of both ingredients will result in a powerful pose estimation algorithm capable of dealing with very large models.

III. Building the 3D Model

Our approach assumes a 3D textured model of the scene to be available. To build these models we used Bundler [20], a structure-from-motion system for unordered image collections. Given a set of N images of the scene we seek to model, this package initially extracts SIFT descriptors of all images, and then simultaneously estimates the N camera poses and 3D structure using bundle adjustment. This is usually a time consuming process that can take a few hours.

For each 3D point of the model, Bundler provides its SIFT descriptor, its 3D position, its RGB color and the number of images where the point appears. Also, for each image it gives the calibration matrix, the rotation matrix, $\mathbf{R}$, and the translation vector, $\mathbf{t}$. This data will be used as "ground truth" when evaluating the precision of the algorithm.

For this paper, we used two 3D models: The *Sagrada Familia* dataset (from [17]), composed of 478 images of this church in Barcelona. The resulting 3D model contains

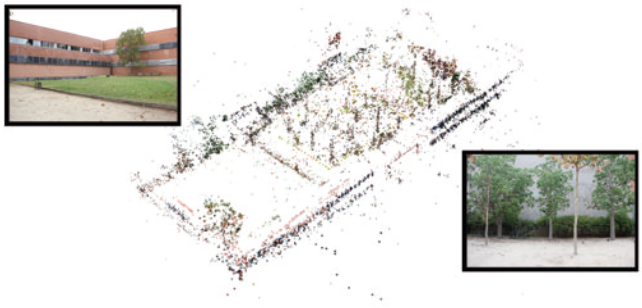


Fig. 3: Courtyard 3D model built and two sample images.

$100,532$ 3D points. Fig. 1(b) shows the model, and plots in yellow the retrieved camera pose for each one of the N training set images. The *Courtyard* of the mathematics school dataset is composed of 265 images, which resulted in a 3D model with $30,196$ points. The model and a two sample images are shown in Fig. 3. Note that while the amount of points in the first model is larger, its images present less difference in terms of visual appearance.

IV. Merging Appearance and Geometric Methods

We next describe how appearance and geometric methods are combined in a top-down coarse-to-fine approach, to tackle the problem of pose estimation from very large models. Let us first formulate our problem:

Assume we are given a 3D model with n points $\{\mathbf{p}_1, \ldots, \mathbf{p}_n\}$ and N training images $\{T_1, \ldots, T_N\}$ which have been used to compute the 3D model. Each of these images has an associated pose $\{\mathbf{R}_i, \mathbf{t}_i\}$. Each point $\mathbf{p}_i$ has an associated SIFT descriptor and a visibility list $\mathbf{v}_i$ with the indexes of the training images from where it is seen. Given an input image I, our goal is to accurately compute the pose from where it was acquired.

One straightforward solution would be to extract m 2D features $\{\mathbf{u}_1, \ldots, \mathbf{u}_m\}$ from the input image I and compute 2D-to-3D correspondences $\{\mathbf{u}_i \leftrightarrow \mathbf{p}_i\}$ by just comparing SIFT descriptors. This set of correspondences could then be filtered using a RANSAC+PnP scheme to get rid of outliers and accurately estimate the pose. Nonetheless, since we are considering cases where $n >> m$ those correspondences are prone to contain a very large percentage of outliers, which might dramatically slow down the process.

In this work we propose a two stage strategy that combines the so-called appearance and geometric methods. The former will compute the subset of the training images which is more similar to our input image. This will constrain the set of candidate poses. The latter will use this subset of poses to limit the number of 3D points of the model that are potentially visible, and refine the pose using a geometric approach. These two steps are next discussed.

A. Coarse Pose Estimation

Given our input image I and training images $\{T_1, \ldots, T_N\}$ we seek to design a fast and robust strategy to get the k training images which are more similar to I. For obtaining this subset we could represent the images using any global image descriptor, e.g, bag of words, GIST, and simply compare such descriptors.

In this paper, though, we have used the recent generic image-level features obtained from a deep network for image representation. In particular, we use the 4,096-dimensional second to last layer of a Convolutional Neural Network (CNN) as our high-level image representation. The full network has 5 convolutional layers followed by 3 fully connected layers, and obtained the best performance in the ILSVRC-2012 challenge. The network is trained on a subset of ImageNet [3] to classify 1,000 different classes. We use the publicly available implementation and pre-trained model provided by [8]. The features obtained with this procedure have been shown to generalize well and outperform traditional hand-crafted features, thus they are already being used in a wide diversity of tasks [21].

Comparing the resulting vectors of this representation we obtain, for each input image I, a subset of the k most "similar" training images $\{T_1, \ldots, T_k\}$. These images may or may not be clustered in a similar region of the space. Indeed, the value of k is chosen sufficiently large to ensure that the subset of "similar" images contains at least one image which is actually close to I, i.e, the set of poses associated to these images represents just a very rough estimation of the ground truth pose. We next explore them and refine the pose using a geometric approach.

B. Fine Pose Estimation

The k closest training images provide a coarse estimate of the camera pose. To increase the accuracy we adapted the REPPnP method [4], which simultaneously allows to discard outlier correspondences while estimates the camera pose.

Standard PnP approaches assume the 2D-to-3D correspondences to be free of outliers. Therefore, when dealing with real images these methods need an outlier rejection preprocessing step (e.g. RANSAC + P3P), which may significantly reduce the overall computational efficiency.

Given c 2D-to-3D correspondences $\{\mathbf{u}_i \leftrightarrow \mathbf{p}_i\}$ and the camera internal calibration matrix $\mathbf{A}$, PnP methods build upon the perspective constraint for each 2D feature point i,

$$d_i \begin{bmatrix} \mathbf{u}_i \\ 1 \end{bmatrix} = \mathbf{A} \, [\mathbf{R}|\mathbf{t}] \begin{bmatrix} \mathbf{p}_i \\ 1 \end{bmatrix} , \tag{1}$$

where d_i is the depth of the feature point.

REPPnP reformulates the previous equations as a low-rank homogeneous set of equations $\mathbf{M}\mathbf{x} = 0$, where $\mathbf{M}$ is a $2c \times 12$ matrix representing the perspective constraints for all c correspondences. In [4] is shown that the solutions $\mathbf{x}$ can be estimated assuming that the rank of the null-space of $\mathbf{M}$ is equal to 1. Once $\mathbf{x}$ is estimated, the camera pose $[\mathbf{R}|\mathbf{t}]$ is solved using a generalization of the Orthogonal Procrustes problem [18], combined with a projected gradients optimization. In REPPnP, the outlier rejection is done by iterating the estimation of $\mathbf{x}$. At each iteration, those equations in $\mathbf{M}$ that after being projected onto $\mathbf{x}$ provide the lowest algebraic errors are chosen. This procedure shows

k	Time (s)	# 3D points	# Matches	e_{rot}	e_{trans}
		Sagrada Familia model			
All	45.6694	100532	470	0.0124	0.0231
6	12.3344	24855	424	0.0178	0.0305
8	14.1634	29046	436	0.0176	0.0306
12	16.9470	35422	447	0.0177	0.0303
16	20.2644	43005	452	0.0178	0.0306
		Courtyard model			
All	50.2912	30196	379	0.0279	0.0449
3	15.5296	6189	296	0.0255	0.0441
6	21.2358	10038	322	0.0267	0.0476
8	24.3889	12134	328	0.0212	0.0343
12	28.9528	15152	334	0.0206	0.0360
16	33.5211	18145	337	0.0303	0.0538

TABLE I: Performance of the proposed approach in the two models for different values of the parameter k. We also consider the case when *all* the model points are considered.

convergence with up to 50% of outliers and requiring up to two orders of magnitude less computational time than standard P3P + RANSAC + PnP algorithms.

In order to adapt REPPnP to the case of this paper with large scenarios, we propose building the $\mathbf{M}$ matrix as k different parts, each coming from one of the k similar images retrieved in the previous stage. By doing this we can reduce drastically the number of outliers, to rates below the 50%, for which the REPPnP is shown to succeed. Concretely, we propose to solve,

$$\left[\mathbf{M}^{1^\top}\mathbf{M}^{2^\top}...\,\mathbf{M}^{k^\top}\right]^\top \mathbf{x} = 0 \qquad (2)$$

where $\mathbf{M}^j$ for $j = [1, 2, ..., k]$ represent the homogeneous equations obtained as in [4] by matching the 2D points $\mathbf{u}_i$ with the 3D points $\mathbf{p}_i^j$ which are visible (according to $\mathbf{v}_, \ldots, \mathbf{v}_k$) in each of the k nearest images.

With the proposed method, taking $k = 6$, we decrease the number of 3D points to be matched to one quarter of the total in the Sagrada Familia scenario and to one third in the Courtyard scenario, increasing the number of inlier correspondences. Table I shows the performance of our approach for both models and different values of k.

V. Results

In this section, the efficiency and accuracy of the proposed method will be evaluated using the two models presented in Sec.III. Previously, we will show how our image retrieval algorithm performs on the test datasets.

For the experiments, both datasets have been evenly split into two disjoint sets. One of them is used as training set for the construction of the model, and the other one as testing set for the evaluation of the method.

A. Image retrieval for Coarse Pose Estimation

In order to evaluate the quality of the image retrieval using the deep network descriptors, we have plotted in Fig. 4 the Euclidean distance between the descriptors for each pair of test/train images in the Sagrada Familia dataset. For each test image (row) the training images are ordered by angular distance of the rotation matrix. The accumulation of bluish regions on the left hand side of the graph indicates that the distance between descriptors increases as we move away from the original viewpoint, and thus confirms a high confidence in the image retrieval process we use.

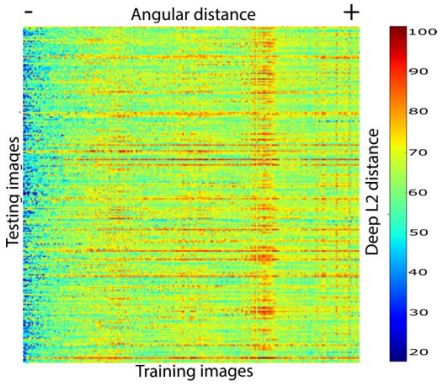


Fig. 4: Comparison of the L2 distances between distance deep network features and the known angular distance between the test images and training images. Each row is sorted according to the angular distance. We can see a strong correlation between images that are close to each other and their descriptors obtained from the deep network.

As illustrative examples, in Fig. 5 we show one sample query for each dataset. We plot both the input image and the closest images in the training set according to the distances between the deep network descriptors. In both cases nearly all selected images present similar poses to the test image.

B. Efficiency assessment

In order to have a clear picture of the cost of our method, we have measured the computational time for each of the different stages: SIFT feature extraction on the input image, image retrieval through our deep learning approach, selection of the corresponding SIFT descriptors from the model, descriptor matching and outlier removal along with pose estimation done by the REPPnP algorithm. To compare the computation time of the approach against a typical baseline, we have replaced REPPnP in our pipeline, by RANSAC combined with OPnP [28], one of the fastest and most accurate approaches in the state-of-the-art.

Average times over the complete test set are given in Fig. 6. As can be observed, the coarse filtering of model images reduces significantly the computation time of the geometrical estimation of the camera pose.

It can also be seen in the figure that the most time consuming step of the whole pipeline is by far the SIFT descriptor matching. This step is performed using the MATLAB implementation of the VLFeat open source library [25].

It is, therefore, critical to obtain a good set of neighboring images: the better it is, the less model descriptors will be required in the geometrical estimation step. In our experiments, this coarse to fine approach led to 75% reduction of the computational time, but it could be even more significant on larger models.

Finally, in Table II we show the average time required to compute the deep network descriptors and the bag of visual words from a new image (left), and the REPPnP

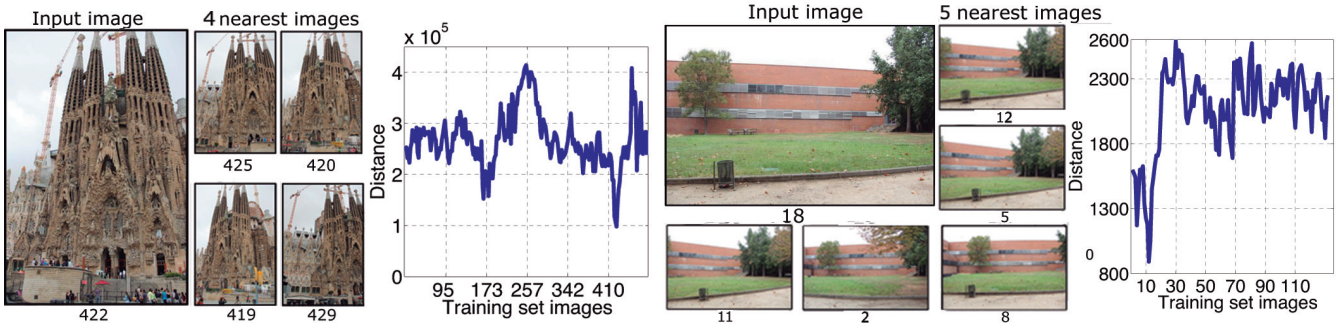


Fig. 5: Input image 422 of the Sagrada Familia model (top) and 18 of the Courtyard model (bottom), with similar images selected by our image retrieval algorithm ($k = 4$ and $k = 5$) and a plot showing the distance to the training images.

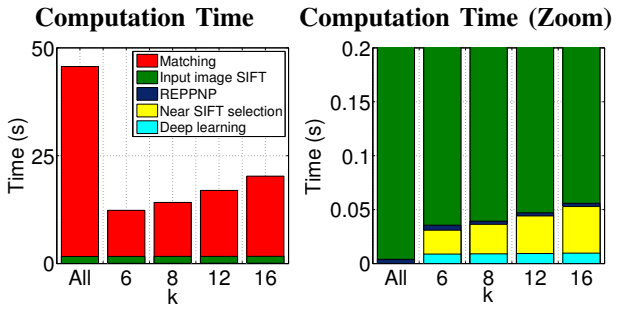


Fig. 6: Analysis of computation time. Left: Computation time of the approaches with different values of k. The case *All* is equivalent to not doing any appearance based pre-computation over the 3D model. Right: Zoom of the computation times.

Model	Bag of Words	CNN Features	Gain (%)
Sag. Familia	1.63	0.0208	98.72
Courtyard	5.48	0.0207	99.62
k	**RANSAC+OPnP**	**REPPnP**	**Gain (%)**
6	0.0217	0.0047	78.34
8	0.0193	0.0030	84.46
12	0.0188	0.0030	84.04
16	0.0187	0.0030	83.96

TABLE II: Time values in seconds for the different methods evaluated. The upper part of the table compares the two methods evaluated for the coarse estimation of the pose (Bag of words vs. Deep network features). The lower part reports the computation time of the methods used for fine pose estimation (RANSAC + OPnP vs. REPPnP).

and RANSAC (right). While a deep network descriptor can be extracted directly from an input image in a matter of milliseconds, the bag of visual words requires first computing SIFT descriptors of that image (on the order of seconds, depending on the size of the image), and then finding the corresponding visual word for each descriptor with a pre-computed dictionary. Regarding the geometrical estimation part, by using REPPnP we are able to reduce the time required by a factor of 5.

C. Accuracy

The accuracy is computed as the rotation and the translation errors in the calculated pose. The rotation error is estimated using quaternions as $e_{\text{rot}} = \|quat(\mathbf{R}) - quat(\mathbf{R}_{\text{true}})\|/\|quat(\mathbf{R}_{\text{true}})\|$, and the translation error as $e_{\text{trans}} = \|\mathbf{t} - \mathbf{t}_{\text{true}}\|/\|\mathbf{t}_{\text{true}}\|$. The estimated pose is $\{\mathbf{R}, \mathbf{t}\}$, and $\{\mathbf{R}_{\text{true}}, \mathbf{t}_{\text{true}}\}$, corresponds to the ground truth given by the Bundler algorithm, as mentioned in Sec. III. As observed in Fig. 7, the errors found using the coarse to fine approach are comparable to the ones obtained with the complete model. In all experiments, we obtain a significant reduction of the error when compared to RANSAC+OPnP.

Fig. 8 shows a qualitative comparison of the reprojection of the SIFT descriptors in the input image obtained with the $\mathbf{R}$ and $\mathbf{t}$ given by REPPnP and RANSAC+OPnP, along with the ground truth reprojection (given by Bundler). By using REPPnP we obtain good results over all the datasets while in contrast, RANSAC+OPnP, does not always provide a good estimation in all images.

VI. Conclusions

The time required to estimate pose on a large scale 3D models has been significantly reduced using a method that combines a purely appearance based technique with a geometrical approach. By using first global image appearance we reduce the number of matches to test but at the same time by performing a PnP match over the candidate images the error in our pose estimation becomes nearly negligible.

In this work we have shown how it is possible to leverage deep learning techniques to improve appearance based approaches for the robotic community. It is remarkable that we are able to perform accurate pose estimation over hundreds of thousands of points in a few seconds per image, especially considering that we are using a MATLAB implementation. As future work, we will consider the possibility of further exploiting deep networks by replacing the ubiquitous SIFT descriptor by descriptors learnt with CNNs [19].

VII. Acknowledgments

This work has been partially funded by the Spanish Ministry of Economy and Competitiveness under projects ERA-Net Chistera project ViSen PCIN-2013-047, PAU+ DPI2011-27510 and ROBOT-INT-COOP DPI2013-42458-P, and by the EU project ARCAS FP7-ICT-2011-28761.

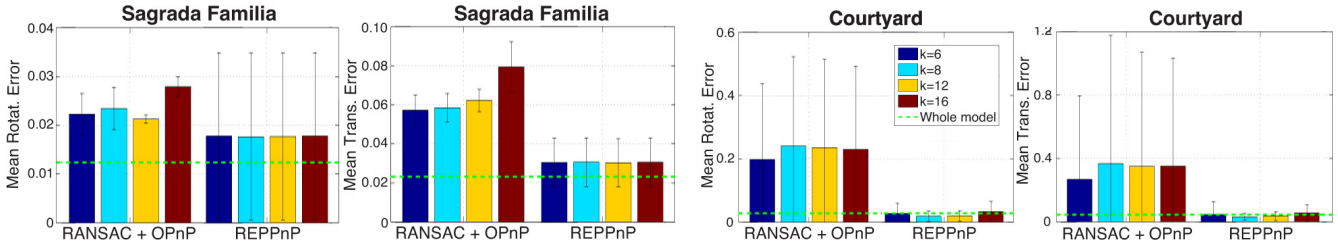
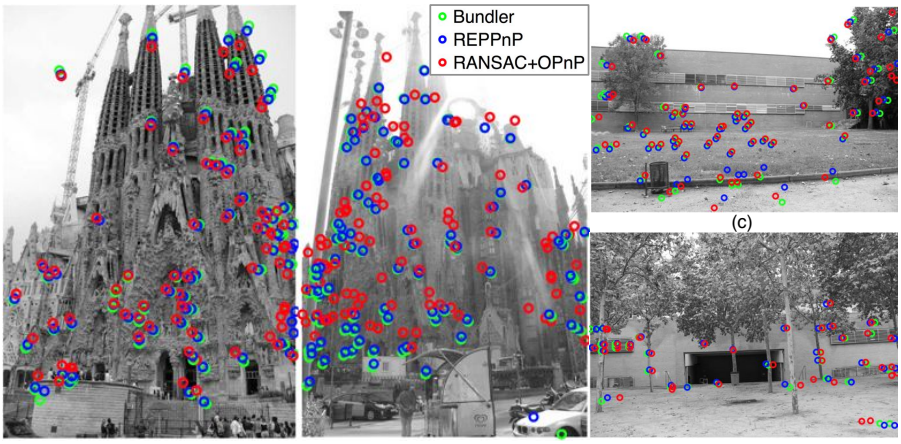


Fig. 7: Rotation and translation errors for different values of k nearest images for both models. All charts show, as a reference, the corresponding error when estimating the pose with the whole model.



Rotation errors		
	REPP*n*P	**RANSAC+OP*n*P**
(a)	0.0068	0.0007
(b)	0.0042	0.0812
(c)	0.0033	0.0024
(d)	0.0012	0.0079

Translation errors		
	REPP*n*P	**RANSAC+OP*n*P**
(a)	0.0306	0.0383
(b)	0.0109	0.1497
(c)	0.0091	0.0091
(d)	0.0109	0.0267

Fig. 8: Examples of pose estimation results for the Sagrada Familia (left) and Courtyard (right) models. For each image we show the reprojected 3D coordinates of the SIFT points using the ground truth $\mathbf{R}$ and $\mathbf{t}$ (from Bundler) and the ones estimated by both REPP*n*P and RANSAC+OP*n*P with $k = 6$.

REFERENCES

[1] Vicon. www.vicon.com.

[2] O. Chum and J. Matas. Matching with PROSAC-progressive sample consensus. In *CVPR*, 2005.

[3] J. Deng, W. Dong, R Socher, L.-J. Li, K. Li, and L Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[4] L. Ferraz, X. Binefa, and F. Moreno-Noguer. Very fast solution to the PnP problem with algebraic outlier rejection. In *CVPR*, 2014.

[5] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. In *Communications ACM*, 1981.

[6] D. Glasner, M. Galun, S. Alpert, R. Basri, and G. Shakhnarovich. Viewpoint-aware object detection and pose estimation. In *ICCV*, 2011.

[7] W. Hu and S.-C. Zhu. Learning a probabilistic model mixing 3D and 2D primitives for view invariant object recognition. In *CVPR*, 2010.

[8] Y. Jia. Caffe: An open source convolutional architecture for fast feature embedding. In *http://caffe.berkeleyvision.org/*, 2013.

[9] A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[10] V. Lepetit, F. Moreno-Noguer, and P. Fua. EPnP: An accurate O(n) solution to the PnP problem. *IJCV*, 81(2):155–166, 2009.

[11] J. Liebelt and C. Schmid. Multi-view object class detection with a 3d geometric model. In *CVPR*, 2010.

[12] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[13] F. Moreno-Noguer, V. Lepetit, and P. Fua. Accurate noniterative O(n) solution to the PnP problem. In *ICCV*, 2007.

[14] F. Moreno-Noguer, V. Lepetit, and P. Fua. Pose priors for simultaneously solving alignment and correspondence. In *ECCV*, 2008.

[15] M. Ozuysal, V. Lepetit, and P. Fua. Pose estimation for category specific multiview object localization. In *CVPR*, 2009.

[16] N. Payet and S. Todorovic. From contours to 3D object detection and pose estimation. In *ICCV*, 2011.

[17] A. Penate-Sanchez, F. Moreno-Noguer, J. Andrade-Cetto, and F. Fleuret. LETHA: Learning from high quality inputs for 3D pose estimation in low quality images. In *3DV*, 2014.

[18] P.H. Schönemann and R.M. Carroll. Fitting one matrix to another under choice of a central dilation and a rigid motion. *Psychometrika*, 35(2):245–255, 1970.

[19] Edgar Simo-Serra, Eduard Trulls, Luis Ferraz, Iasonas Kokkinos, and Francesc Moreno Noguer. Fracking Deep Convolutional Image Descriptors. *CoRR*, abs/1412.6537, 2014.

[20] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: Exploring image collections in 3d. In *SIGGRAPH*, 2006.

[21] R. Socher, A. Karpathy, Q.V. Le, C.D. Manning, and A.Y. Ng. Grounded compositional semantics for finding and describing images with sentences. In *TACL*, 2014.

[22] H. Su, M. Sun, L. Fei-Fei, and S. Savarese. Learning a dense multi-view representation for detection, viewpoint classification and synthesis of object categories. In *ICCV*, 2009.

[23] L. Svarm, O. Enqvist, M. Oskarsson, and F. Kahl. Accurate localization and pose estimation for large 3d models. In *CVPR*, June 2014.

[24] A. Thomas, V. Ferrari, B. Leibe, T. Tuytelaars, B. Schiel, and L. Van Gool. Towards multi-view object class detection. In *CVPR*, 2006.

[25] A. Vedaldi and B. Fulkerson. *VLFeat*: An open and portable library of computer vision algorithms. 2008.

[26] M. Villamizar, A. Garrell, A. Sanfeliu, and F. Moreno-Noguer. Online human-assisted learning using random ferns. In *ICPR*, 2012.

[27] M. Villamizar, A. Sanfeliu, and F. Moreno-Noguer. Fast online learning and detection of natural landmarks for autonomous aerial robots. In *ICRA*, 2014.

[28] Y. Zheng, Y. Kuang, S. Sugimoto, K. Aström, and M. Okutomi. Revisiting the pnp problem: A fast, general and optimal solution. In *ICCV*, 2013.