Visual Localization by Learning Objects-of-Interest Dense Match Regression

Philippe Weinzaepfel Gabriela Csurka Yohann Cabon Martin Humenberger
NAVER LABS Europe
firstname.lastname@naverlabs.com

Abstract

We introduce a novel CNN-based approach for visual localization from a single RGB image that relies on densely matching a set of Objects-of-Interest (OOIs). In this paper, we focus on planar objects which are highly descriptive in an environment, such as paintings in museums or logos and storefronts in malls or airports. For each OOI, we define a reference image for which 3D world coordinates are available. Given a query image, our CNN model detects the OOIs, segments them and finds a dense set of 2D-2D matches between each detected OOI and its corresponding reference image. Given these 2D-2D matches, together with the 3D world coordinates of each reference image, we obtain a set of 2D-3D matches from which solving a Perspective-n-Point problem gives a pose estimate. We show that 2D-3D matches for reference images, as well as OOI annotations can be obtained for all training images from a single instance annotation per OOI by leveraging Structure-from-Motion reconstruction. We introduce a novel synthetic dataset, VirtualGallery, which targets challenges such as varying lighting conditions and different occlusion levels. Our results show that our method achieves high precision and is robust to these challenges. We also experiment using the Baidu localization dataset captured in a shopping mall. Our approach is the first deep regression-based method to scale to such a larger environment.

1. Introduction

Visual localization consists in estimating the 6-DoF camera pose from a single RGB image within a given area, also referred to as map. This is particularly valuable if no other localization technique is available, *e.g.* in GPS-denied environments such as indoor locations. Interesting applications include robot navigation [8], self-driving cars and augmented reality (AR) [6, 22]. The main challenges include large viewpoint changes between query and train images, incomplete maps, regions without valuable information (*e.g.* textureless surfaces), symmetric and repetitive elements, varying lighting conditions, structural changes, dynamic objects (*e.g.* people), and scalability to large areas.

Traditional structure-based approaches [17, 18, 24, 26, 27, 34] use feature matching between query image and map, coping with many of the mentioned challenges. However, covering large areas from various viewpoints and under different conditions is not feasible in terms of processing time and memory consumption. To overcome this, image retrieval can be used to accelerate the matching for large-scale localization problems [10, 29, 35]. Recently, methods based on deep learning [2, 16] have shown promising results. PoseNet [16] and its improvements [4, 7, 15, 39] proceed by directly regressing the camera pose from input images. Even if a rough estimate can be obtained, learning a precise localization seems too difficult or would require a large amount of training data to cover diversities both in terms of locations and intrinsic camera parameters. More interestingly, Brachmann *et al*. [2, 3] learn dense 3D scene coordinate regression and solve a Perspective-n-Point (PnP) problem for accurate pose estimation. The CNN is trained end-to-end thanks to a differentiable approximate formulation of RANSAC, called DSAC. Scene coordinate regression-based methods obtain outstanding performance in static environments, *i.e.*, without changes in terms of objects, occlusions or lighting conditions. However, they are restricted in terms of scene scale.

The above mentioned challenges of visual localization become even more important in very large and dynamic environments. Essential assumptions, such as static and unchanged scenes are violated and the maps become outdated quickly while continuous retraining is challenging. This motivated us to design an algorithm inspired by advances on instance recognition [11] that relies on stable, predefined areas, and that can bridge the gap between precise localization and long-term stability in very vivid scenarios. We propose a novel deep learning-based visual localization method that proceeds by finding a set of dense matches between some Objects-of-Interest (OOIs) in query images and the corresponding reference images, *i.e.*, a canonical view of the object for which a set of 2D-3D correspondences is available. We define an Object-of-Interest as a discriminative area within the 3D map which can be reliably detected

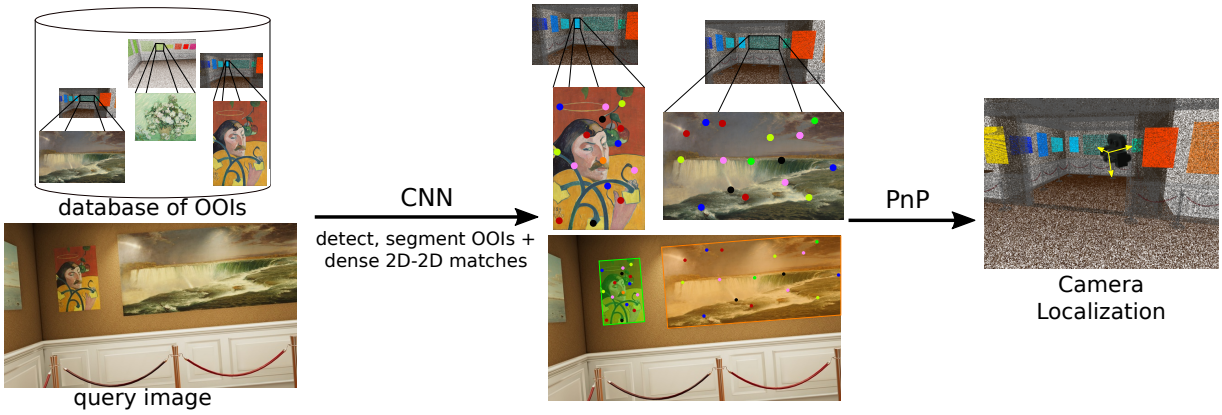

Figure 1. Overview of our pipeline. Given a query image, we use a CNN to first, detect and segment a predefined list of Objects-of-Interest, and second, to regress dense matches to their reference images (dots in color). These reference images contain a dense 2D pixels to 3D coordinates mapping. We thus obtain 2D-3D matches by transitivity and solve a PnP problem to obtain the camera localization.

from multiple viewpoints, partly occluded, and under various lighting conditions. Figure 1 shows an overview of our pipeline. Our model relies on DensePose [9], a recent extension of Mask R-CNN [11] that not only detects and segments humans, but also regresses dense matches between pixels in the image and a mesh surface. In our case, we use it (i) to detect and segment OOIs, and (ii) to obtain a dense set of 2D-2D matches between the detected OOIs and their reference images. Given these matches, together with the 2D-3D correspondences of the reference images, we obtain a set of 2D-3D matches from which camera localization is obtained by solving a PnP problem using RANSAC.

Our method is carefully designed to tackle open challenges of visual localization, it has several advantages, and few limitations with respect to the state of the art. First, reasoning in 2D allows to train the model from few training data: we can artificially generate a rich set of viewpoints for each object with homography data augmentation and we can achieve robustness to lighting changes with color jittering. Second, our method can handle dynamic scenes as long as the OOIs remain static. For instance, one can accurately estimate the pose in a museum with visitors present, even if training data does not contain any humans. Third, if some OOIs are moved, we do not need to retrain the whole network as pose and scene regression-based approaches would require, but we only need to update the 2D-3D mapping of the reference images. Fourth, our method focuses on discriminative objects and thus avoids ambiguous textureless areas. Fifth, our method can scale up to large areas and high numbers of OOIs as object detectors can segment thousands of categories [13]. One clear limitation of our method is that query images without any OOI cannot be localized. However, in many applications such as AR navigation, OOIs are present most of the time and local pose tracking (*e.g.* visual-inertial odometry [1]) can be used in-between. OOI detection is interesting by itself in such an application, *e.g.* to display metadata on paintings in a museum or on shops in malls and airports. Furthermore, in a complex real-world application, OOIs can be used to more easily guide the user to localize successfully. Commands such as 'Take a picture of the closest painting' might be easier to understand than 'Take a picture with sufficient visual information'.

In this paper, we restrict OOIs to planar objects which are common in real-world applications: paintings, posters, store fronts or logos are frequent in environments where localization is challenging such as shopping malls, airports or museums. While the method can be generalized to non-planar objects, considering planar OOIs has several advantages, in addition to homography data augmentation. First, the transformation between any instance of the OOI in a training image and its reference image is a homography, thus allowing to easily propagate dense sets of matches using a few correspondences only. Second, the mapping between 2D pixels in the reference image and 3D world coordinates can be built from a small set of images since planes can be robustly reconstructed in 3D. Finally, planar OOIs allow us to reason in 2D, removing one degree of freedom compared to 3D coordinate regression. Additionally, we show that our method can be used with a minimal amount of manual annotations, being one instance segmentation for each (planar) OOI in any training image.

We demonstrate the strength of our approach on two challenging datasets. The first one is a newly introduced synthetic dataset, VirtualGallery, which represents an art gallery. The second one is the Baidu localization dataset [33] captured in a shopping mall with only a few viewpoints at training and some changes in the scenes at testing, showing the applicability of our approach in complex real-world environments. While our method is accurate on VirtualGallery (even with different lighting conditions and occlusions) achieving a median error of less than 3cm and 0.5°, it also scales up to larger environment such as the Baidu localization dataset. In contrast, deep state-of-the-art regression-based approaches fail in such scenarios.

2. Related Work

Most methods for visual localization can be categorized into four types of approaches: structure-based methods, image retrieval-based methods, pose regression-based methods and coordinate regression-based methods.

Structure-based methods [17, 18, 24, 26, 27, 34] use descriptor matching (*e.g*. SIFT [21]) between 3D points of the map associated with local descriptors and keypoint descriptors in the query image. However, these point features are not able to create a representation which is sufficiently robust to challenging real-world scenarios such as different weather, lighting or environmental conditions. Additionally, they lack the ability to capture global context and require robust aggregation of hundreds of points in order to form a consensus to predict a pose [41].

Image retrieval-based methods [10, 29, 35, 36, 37] match the query image with the images of the map using global descriptors or visual words to obtain the image location from the top retrieved images. The retrieved locations can further be used to either limit the search range within large maps of structure-based approaches [5, 28], or to directly compute the pose between retrieved and query images [42]. These methods allow to speed-up search in large environments, but share similar drawbacks when using structure-based methods for accurate pose computation. InLoc [35] shows recent advances in image retrieval-based methods leveraging dense information. It uses deep features to first retrieve the most similar images, and then to estimate the camera pose within the map. A drawback is the heavy processing load and the need of accurate dense 3D models.

Pose regression-based methods [4, 16, 39] were the first deep learning approaches trained end-to-end for visual localization. They proceed by directly regressing the 6-DoF camera pose from the query image using a CNN, following the seminal PoseNet approach [16]. The method has been extended in several ways, by leveraging video information [7], recurrent neural networks [39], hourglass architecture [23] or a Bayesian CNN to determine the uncertainty of the localization [14]. More recently, Kendall *et al*. [15] replace the naive L2 loss function by a novel loss that relies on scene geometry and reprojection error. Brahmbhatt *et al*. [4] additionally leverage the relative pose between pairs of images at training. Overall, pose regression-based methods have shown robustness to many challenges but remain limited both in accuracy and scale.

Scene coordinate regression-based methods [2, 32, 38] proceed by regressing dense 3D coordinates and estimating the pose using a PnP solver with RANSAC. While random forests were used in the past [32, 38], Brachmann *et al*. [2] recently obtain an extremely accurate pose estimation by training a CNN to densely regress 3D coordinates. They additionally introduce a differentiable approximation of RANSAC, called DSAC, allowing end-to-end training for visual localization at the cost of multi-step training, with depth data required for the first step. DSAC++ [3] is an improvement of the method where real depth data is not mandatory and can be replaced by a depth prior. The network is still trained in multiple steps: first based on depth data or the prior; second, based on minimization of the reprojection error; and finally based on camera localization error using the DSAC module. DSAC++ obtains outstanding performance on datasets of relatively small scale with constant lighting conditions and little dynamics. However, the method does not converge for larger scenes. In contrast, our method is built upon an object detection pipeline, which scales to large environments. Compared to DSAC++, since we consider planar objects, we can regress 2D matches instead of 3D coordinates, which removes one degree of freedom, and allows homography data augmentation.

3. Visual Localization Pipeline

In this section, we first present an overview of our approach in Section 3.1. Next, we introduce details about our CNN model for segmenting OOIs and dense matching (Section 3.2). Finally, Section 3.3 explains how SfM maps can be leveraged to train our approach from weak supervision.

3.1. Visual localization from detected OOIs

Let $\mathcal{O}$ be the set of OOIs, and $|\mathcal{O}|$ the number of OOI classes. Our method relies on reference images: each OOI $o \in \mathcal{O}$ is associated with a canonical view, *i.e*., an image $\mathcal{I}_o$ where o is fully visible. We assume for now that each OOI is unique in the environment and that the mapping M_o between 2D pixels $\mathbf{p}'$ in the reference image and the corresponding 3D points in the world $M_o(\mathbf{p}')$ is known.

Given a query image, our CNN outputs a list of detections. Each detection consists of (a) a bounding box with a class label o, *i.e*., the id of the detected OOI, and a confidence score, (b) a segmentation mask, and (c) a set of 2D-2D matches $\{\mathbf{q} \rightarrow \mathbf{q}'\}$ between pixels $\mathbf{q}$ in the query image and pixels $\mathbf{q}'$ in the reference image $\mathcal{I}_o$ of the object-of-interest o, see Figure 1. By transitivity, we apply the mapping M_o to the matched pixels in the reference image and obtain for each detection a list of matches between 2D pixels in the query image and 3D points in the world coordinates: $\{\mathbf{q} \rightarrow \mathcal{M}_o(\mathbf{q}')\}$.

Given the list of 2D-3D matches for all detections in the query image and the intrinsic camera parameters, we estimate the 6-DoF camera pose by solving a Perspective-n-Point problem using RANSAC.

Note that if there is no detection in a query image, we do not have matches and, thus, we are not able to perform localization. However, in venues such as museums or airports, OOIs can be found in most of the images. Moreover, in real-world applications, localization is used in conjunc-

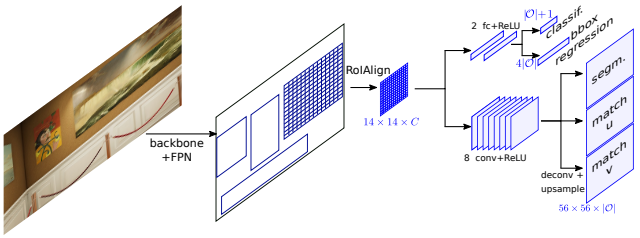

Figure 2. Overview of the network architecture to detect and segment OOIs as well as to obtain dense matches with respect to the reference images.

tion with local pose tracking and, thus, precision of the system is more important than coverage.

In summary, the only components learned in our approach are detection and dense matching between a query image and the reference images of the OOIs.

Handling non-unique OOIs. So far, we assumed that each OOI is unique, *i.e.*, it appears only once in the environment. While most OOIs are highly discriminative, some of them can have multiple instances in an environment, *e.g.* logos in a shopping mall. In this case, as a detector cannot differentiate them, we aggregate the same OOIs in a single class and train the model with a common reference image. The mapping M_o is not unique anymore since there exist multiple 3D coordinate candidates for the same reference image, *i.e.*, one per instance present in the environment. Given that in practice we have a limited number of duplicates per OOI and a limited number of detections per image, we solve the PnP problem for the best combination of possible coordinates using geometric plausibility checks. In detail, we minimize the sum of reprojection errors of the OOI 3D center points in the query image. Ideally, the 3D center point lies in the middle of the segmented detection. We ignore noise due to incomplete detections of OOIs.

3.2. Detecting OOIs and matching to references

We follow DensePose [9], an extension of Mask R-CNN [11], designed for finding dense correspondences between any point on a human body and the corresponding point on the surface mesh. For each box generated by the region proposal networks (RPN) [25], C-dimensional convolutional features are estimated with a fixed resolution of 14×14 using the RoIAlign layer. The Feature Pyramid Networks (FPN) improvement [19] is used to better handle small objects. Box features are fed to two branches. One of them is designed to predict the class score (human *vs*. non-human in their case) and to perform class-specific box coordinate regression, following Faster R-CNN design [25]. The other branch, which is fully-convolutional, predicts a per-pixel human body part label and per-pixel correspondences (*i.e.*, two coordinates) with the corresponding mesh surface. In practice, the CNN predicts segmentation and correspondences on a dense grid of size 56×56 in each box which is then interpolated to obtain a per-pixel prediction.

In our case, given the box features after RoIAlign, we use a similar CNN model as DensePose, see Figure 2. One branch predicts OOI scores and regresses bounding boxes, the only difference being that we have $|\mathcal{O}| + 1$ classes (including the background class) instead of 2. The second branch predicts different tasks: (a) binary segmentation for each OOI, (b) OOI-specific u and v reference image coordinate regression. At training time several losses are combined. In addition to the FPN loss for box proposals, we use the cross-entropy loss for box classification. For the ground-truth class we use a smooth-L1 loss on its box regressor, the cross-entropy loss for the 56×56 mask predictor, and smooth-L1 losses for the $u-$ and $v-$regressors. Training requires a ground-truth mask and matches for every pixel. In Section 3.3, we explain how such annotations can be automatically obtained from minimal annotation. At test time, we keep the box detections with classification score above 0.5 and keep the matches for the points within the segmentation mask.

Implementation details. We use FPN [19] with both ResNet50 [12] and ResNeXt101-32x8d [40] backbones. The branch that predicts segmentation and match regression follows the Mask R-CNN architecture: it consists of 8 convolutions and ReLU layers before a final convolutional layer of each task. We train the network for 500k iterations, starting with a learning rate of 0.00125 on 1 GPU and dividing it by 10 after 300k and 420k iterations. We use SGD as optimizer with a momentum of 0.9 and a weight decay of 0.0001. To make all regressors proceeding at the same scale, we normalize the reference coordinates in $[0, 1]$.

Data augmentation. As our CNN regresses matches only in 2D, we apply homography data augmentation on all input images. To generate plausible viewpoints, we compute a random displacement limited by 33% of the image size for each of the 4 corners and fit the corresponding homography. We do not use flip data augmentation as OOIs (logos, paintings, posters) are left-right ordered. We study the impact of using color jittering (brightness, contrast, saturation) for robustness against changing lighting conditions in our experiments (Section 5).

3.3. Weakly-supervised OOI annotation

Key of our approach is the minimal amount of manual annotations required, thanks to a propagation algorithm that leverages a SfM reconstruction obtained with COLMAP [30]. The only annotation to provide is one segmentation mask for each planar OOI, see the blue mask in the middle of Figure 3. The reference image for this OOI is defined by the annotated mask.

Using the set of 2D to 3D matches from SfM, we label 3D points that match with 2D pixels in the annotated OOI segmentation, see blue lines and dots in Figure 3. We

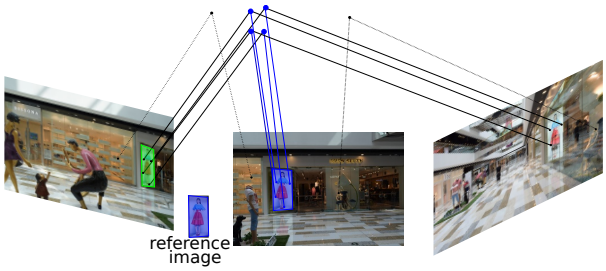

Figure 3. The bounding box of the manual mask annotation in blue (middle frame) defines the reference image for the OOI. We propagate the OOI to the other training images (green mask in the left image) using the 3D positions of keypoints within the annotation (blue mask). Propagation may fail if not enough matches are present (right image).

propagate the label to all training images which contain observations of these 3D points. If there are at least 4 matches in an image, we can fit a homography given that the OOI o is planar. For more robustness to noise, we only consider regions with a minimum of 7 matches and use RANSAC-based homography estimation. This homography is used to propagate the mask annotation as well as the dense 2D-2D matches, see the green mask on the left image of Figure 3.

Either due to a low number of matches leading to missing propagation, see right image in Figure 3, or because of noisy matches in the SfM model, the propagation is not always successful. Fortunately, the CNN model is, to some extent, robust against noisy or missing labels.

Handling non-unique OOIs. For non-unique OOIs, such as a logo appearing multiple times in a shopping mall, we annotate one segmentation mask for each instance of the OOI and we apply our propagation method on each instance independently. As mentioned above, it would be impossible for any detector to differentiate between the different instances. Thus, we merge them into a single OOI class and compute a homography between the reference images of the different instances and the dedicated main reference image using SIFT descriptor [21] matches. As main reference for the class (OOI), we select the reference image with the highest number of 3D matches. Since the regressed 2D-2D matches correspond to the main class, we additionally apply a perspective transform using the computed intra-class homographies, see Section 3.1.

4. Datasets

The VirtualGallery dataset. We introduce a new synthetic dataset to study the applicability of our approach and to furthermore measure the impact of varying lighting conditions and occlusions on different localization methods. It consists of a scene containing 3-4 rooms, see Figure 4 (left), in which 42 free-of-use famous paintings[1] are placed on the

[1] https://images.nga.gov/

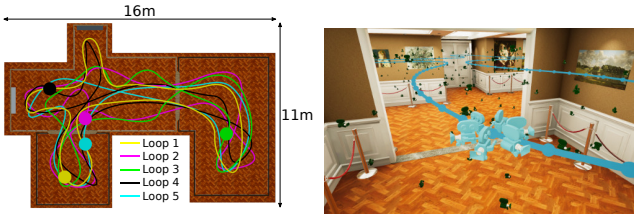

Figure 4. *Left:* floorplan of the art gallery with the different training loops. *Right:* a training loop with the 6 cameras at a fixed height (cyan) and test cameras at different plausible places.

Figure 5. Test samples from VirtualGallery. *First row:* test images with various viewpoints. *Second and third rows:* different lighting conditions. *Fourth row:* different human densities (occlusions).

walls. The scene was created with the Unity software, allowing to extract ground-truth information such as depth, semantic and instance segmentations, 2D-2D and 2D-3D correspondences, together with the rendered images.

We consider a realistic scenario that simulates the scene captured by a robot for training and photos taken by visitors for testing. The camera setup consists of 6 cameras in a 360° configuration at a fixed height of 165cm, see Figure 4 (right). The robot follows 5 different loops inside the gallery with pictures taken roughly every 20cm, resulting in about 250 images for each camera, see Figure 4 (left).

At test time, we sample random positions, orientations and focal lengths, ensuring that viewpoints (a) are plausible and realistic (in terms of orientation, height and distance to the wall), and (b) span the entire scene. This covers the additional challenges of viewpoint changes and varying intrinsic camera parameters between training and test images.

To study robustness to lighting conditions, we generate the scene using 6 different lighting configurations with significant variations between them, both at training and test time, see second and third rows of Figure 5. To evaluate robustness to occluders such as visitors, we generate test images which contain randomly placed human body models, see last row of Figure 5. The test set con-

Figure 6. Examples of training images from the Baidu localization dataset with propagated mask annotations.

sists of 496 images that are rendered for each of the 6 lighting conditions and for 4 different densities of humans present in the scene (including the empty case). The dataset can be found at http://www.europe.naverlabs.com/Research/3D-Vision/Virtual-Gallery-Dataset.

Objects-of-Interest. We use each painting as an object-of-interest, each one being unique in the scene. We use the original image downloaded from the website as reference image. We obtain ground-truth segmentation masks and 2D-2D correspondences for each image using Unity. We also get the position and size where the painting is placed in the scene, thus providing the 2D-3D mapping functions. Note that, among the test images, 5 out of 496 do not contain any OOI, and 23 (respectively 48) additional ones have no OOI visible by more than 50% (respectively 80%).

The Baidu localization dataset [33]. It consists of images captured in a Chinese shopping mall covering many challenges for visual localization such as reflective and transparent surfaces, moving people and repetitive structures. It contains 689 images captured with DSLR cameras as training set and over 2000 cell phone photos taken by different users a few months later as test set. The test images contain significant viewpoint changes compared to the training images that are all taken parallel or perpendicular with respect to the main corridor. All images were semi-automatically registered to the coordinate system defined by a LIDAR scanner. We use the provided camera poses, both for training (3D reconstruction of the OOIs and annotation propagation) and testing (ground-truth evaluation of the results). We did not use the LIDAR data, even if it would potentially improve the 3D reconstruction quality of our OOIs.

Objects-of-Interest. We manually annotated one segmentation mask for 220 instances from 164 classes representing different types of planar objects such as logos or posters on storefronts. We then propagated these annotations to all training images, see Section 3.3. This real-world dataset is more challenging than VirtualGallery, thus, some OOI propagations are noisy. Figure 6 shows examples of training images with masks around the OOIs after propagation.

5. Experimental results

We evaluate our approach on VirtualGallery (Section 5.1) and on the Baidu localization dataset (Section 5.2).

5.1. Experiments on VirtualGallery

We use different train/test scenarios to evaluate some variants of our method and to study the robustness of state-of-the-art approaches to lighting conditions and occlusions. In the experiments below, we use the ground-truth correspondences obtained from Unity. We experimented with manual annotations and obtained similar performance.

Impact of data augmentation. We first study the impact of homography data augmentation at training. We train models with ResNet50 (resp. ResNext101) backbone on the first loop of the standard lighting condition, and report the percentages of successfully localized images with the standard lighting condition and no humans in Figure 7 (plain blue and dotted blue, resp. black, curves). Homography data augmentation significantly improves the performance, specifically for highly accurate localization: the ratio of localized images within 5cm and 5° increases from 25% to 69% with ResNet50 backbone. The impact is less significant at higher error thresholds with an improvement from 72% to 88% at 25cm and 5°. Homography data augmentation at training allows to generate more viewpoints of the OOIs and, thus, to better detect and match OOIs captured from unknown viewpoints at test time.

Impact of regressing dense 2D-2D matches. We now compare our approach that relies on 2D-2D dense correspondences to a few variants. First, we directly regress the 8-DoF homography parameters (OOIs-homog) between the detected OOI and its reference image and generate the dense set of 2D-2D matches afterwards. Second, we directly regress 3D world coordinates for each OOI instance (OOIs-2D-3D) without using the reference image. Performances with the same train/test protocol as above are reported in Figure 7. Regressing the 8 parameters of the homography leads to a drop of performance, only 70% of the images could be successfully localized within 25cm and 5°. CNNs have difficulties regressing parameters of transformations where small differences can lead to significant changes of the result. The 3D variant performs reasonably well, with roughly 70% of images localized within 10cm and 5°, and 81% within 25cm and 5°. However, our method with 2D reference images outperforms the 3D variant, specifically for low position error thresholds. Our 2D reference images ensure that all 3D points are on the planar object, while the 3D variant adds one extra and unnecessary degree of freedom. We finally replace the ResNet50 backbone by ResNeXt101-32x8d and obtain a more precise localization at low thresholds (8% additional images are localized within 5cm and 5°); the difference becomes marginal at higher thresholds.

Comparison to the state of the art. We now compare our approach to a SfM method (COLMAP [30, 31]), PoseNet with geometric loss [15], and DSAC++ trained with a 3D model [3]. All methods are trained on all loops from all

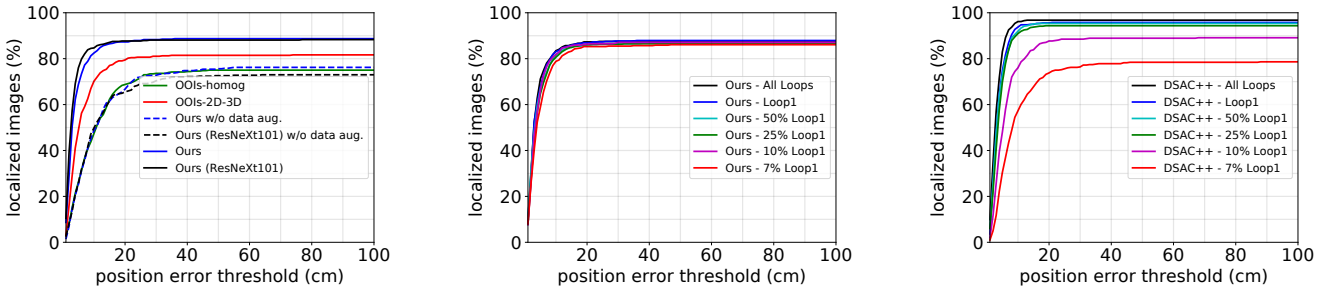

Figure 7. Percentages of localized images on the VirtualGallery test set with the standard lighting condition without humans for varying position error thresholds and a fixed orientation error threshold of 5°. *Left:* comparison between variants of our approach trained on the first loop of the standard lighting condition. *Middle and Right:* robustness to a lower amount of training data (from the standard lighting condition) for our approach (*Middle*) and DSAC++ (*Right*).

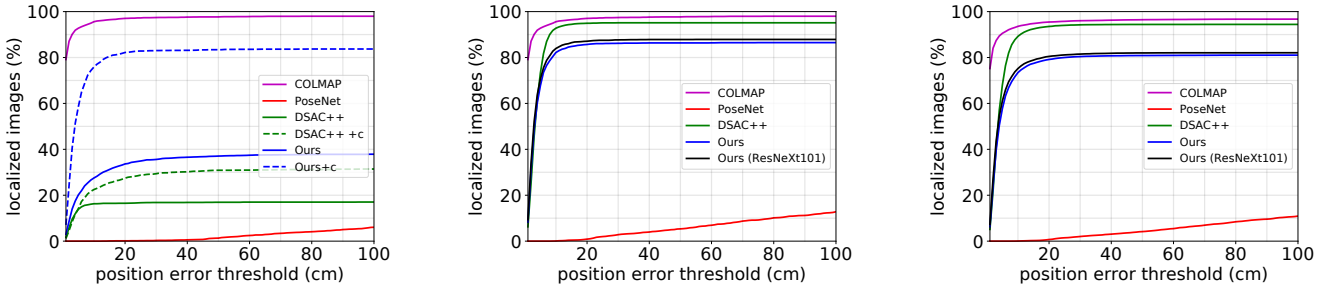

Figure 8. Robustness to lighting conditions and occlusions with the percentages of localized images on VirtualGallery test set for varying position error threshold and a fixed 5° orientation error threshold. *Left:* training on the standard lighting condition, testing on all lighting conditions without human (+c indicates training with color jittering). *Middle:* training on all lighting conditions (except COLMAP), testing on all lighting conditions without humans. *Right:* training on all lighting conditions (except COLMAP), testing on all lighting conditions with all human occlusion levels.

lighting conditions and tested on all lighting conditions without human occlusion, except COLMAP which is only trained using the standard lighting. The percentages of successfully localized images at a given error threshold are reported in Figure 8 (middle). The performance of PoseNet with geometric loss [15] is quite low because the training data does not contain enough variation: all images are captured at the same height, with 0° roll and pitch. Consequently, it learns this training data bias which is not valid anymore on the test data. COLMAP performs best with about 95% of the images localized within 10cm and 5°. DSAC++ localizes 75% of images within 5cm and 5°. Our approach performs similarly at low thresholds achieving about the same percentage of images successfully localized within 5cm and 5°. At higher thresholds, our method saturates earlier (around 88% with ResNeXt101 backbone) than DSAC++. We find that our approach fails to detect OOIs in cases where they are poorly visible (see top right example of Figure 5), thus we cannot localize such images. In contrast, DSAC++ can better handle this case as it relies not only on the OOIs but on the entire image. Overall, our method still computes highly accurate localization in standard cases where at least one OOI is present. We achieve a median error of less than 3cm, which is slightly lower than DSAC++.

Impact of the quantity of training data. Figure 7 (middle) shows the performance of our method when reducing the amount of training data. The percentages of localized images is roughly constant, even when training on only 1 image out of 15 (7%) of the first loop. In contrast, DSAC++, see Figure 7 (right), shows a larger drop of performance when training on a few images, highlighting the robustness of our approach to a small amount of training data. We did not observe significant difference with COLMAP, as two views of each point are sufficient to build the map of this comparably easy dataset for structure-based methods.

Robustness to lighting conditions. To study the robustness to different lighting conditions, we compare the average performance over the test sets with all lighting conditions without human, when training on all loops of the standard lighting condition only, see Figure 8 (left) *vs*. training on all loops and all lighting conditions, see Figure 8 (middle).

The performance of PoseNet drops by about 10% at 1m and 5° when training on only one lighting condition, despite color jittering at training. The performance of COLMAP stays constant even if we train using the standard lighting condition only (SfM does not work well with multiple images from identical poses, as in our training data with different lighting conditions). The high quality images and the unique patterns are easy to handle for illumination invariant SIFT descriptors. The performance of DSAC++ without any color data augmentation drops significantly when training on one lighting condition; in detail, by a factor of around

Figure 9. *Left:* input images with overlaid detections of masks. *Right:* for each detected class, we warp the reference image according to the homography fitted from the regressed matches.

6, which is the number of different lighting conditions. This means that only the images with the same lighting as training are localized, but almost none of the others. However, when adding color jittering to DSAC++ at training, the performance slightly increases.

The performance of our method (plain blue curve, on the left plot) is significantly higher than DSAC++ and PoseNet when training on one lighting condition, with about 30% of the images localized with an error below 25cm and 5°, showing that we overfit less on the training lighting condition. We also try to incorporate color jittering at training (dotted blue curve on the left plot) and obtain a significant increase of performance, achieving almost 85% of localized images with an error below 25cm and 5°. This performance is pretty close to the one obtained when training on all lighting conditions (middle plot), which can be considered as an upper bound of achievable results. In practice, this means that for real-world applications our method can be used even if only one lighting condition is present in the training data.

Robustness to occlusions. To study robustness to occlusions, we compare the performance of all methods when training on all loops and all lighting conditions, and testing on (a) all lighting conditions without visitors, see Figure 8 (middle), and (b) all lighting conditions and various visitor densities, see Figure 8 (right). All methods show a slight drop of performance. For our approach, the decrease of performance mostly comes from images where (a) only one painting is present and (b) the OOI is mostly occluded. This causes the OOI detection to fail. In most cases however, our approach is robust despite not having seen any human at training. Figure 9 shows examples of instance segmentation (left) in presence of humans. To visualize the quality of the matches, we fitted homographies between the test image and the reference images and warped the reference images onto the query image plane. We observe that the masks and the matches remain accurate.

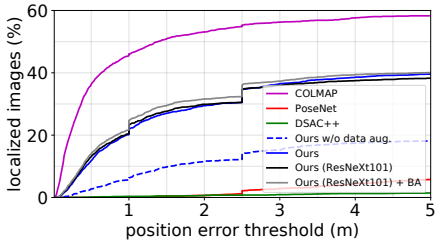

Figure 10. Percentages of localized images on the Baidu localization dataset for varying position error thresholds and an orientation error threshold of 5° for pos. error below 1m, 10° for pos. error between 1m and 2.5m, and 20° for pos. error above 2.5m.

5.2. Experiments on the Baidu localization dataset

Figure 10 presents results on the Baidu localization dataset [33]. This benchmark represents a realistic scenario which makes it extremely challenging: (a) the training data is limited to 689 images, captured in a mall of around 240m, (b) training and test images have different cameras and viewpoints, and (c) the environment has some changes in terms of lighting conditions and dynamic objects. Deep state-of-the-art approaches perform poorly, with less than 2% of images localized within 2m and 10°. COLMAP is able to localize more images, with about 45% at 1m and 5° and 58% at 5m and 20°. Our method is the first deep regression-based method able to compete with structured-based methods on this dataset. We successfully localize about 25% of the images within 1m and 10° and almost 40% within 5m and 20°. To further increase accuracy, we ran a non-linear least square optimization (sparse bundle adjustment [20]) as post processing (grey curve) obtaining a performance increase of about 2%. Figure 10 again highlights the benefit of homography data augmentation at training (plain *vs*. dotted blue curves). We are not able to localize about 10% of the query images where no OOI is detected.

6. Conclusion

We proposed a novel approach for visual localization that relies on densely matching a set of Objects-of-Interest. It is the first deep regression-based localization method that scales to large environments like the one of the Baidu localization dataset. Furthermore, our approach achieves high accuracy on smaller datasets like the newly introduced VirtualGallery. Since our method relies on OOIs, localization is only possible in the presence of OOIs. This putative drawback is a core characteristic of our approach because it enables the use of visual localization in fast changing and dynamic environments. For this to be true we assume that in such a situation at least the selected OOIs remain stable or can be tracked when being moved or changed. The learning component of our approach allows to increase robustness against changing lighting conditions and viewpoint variations. Future work include automatic mining of OOIs as well as generalizing to non-planar OOIs.

References

[1] Michael Bloesch, Michael Burri, Sammy Omari, Marco Hutter, and Roland Siegwart. Iterated extended kalman filter based visual-inertial odometry using direct photometric feedback. *IJRR*, 2017. 2

[2] Eric Brachmann, Alexander Krull, Sebastian Nowozin, Jamie Shotton, Frank Michel, Stefan Gumhold, and Carsten Rother. DSAC-differentiable RANSAC for camera localization. In *CVPR*, 2017. 1, 3

[3] Eric Brachmann and Carsten Rother. Learning less is more - 6D camera localization via 3D surface regression. In *CVPR*, 2018. 1, 3, 6

[4] Samarth Brahmbhatt, Jinwei Gu, Kihwan Kim, James Hays, and Jan Kautz. Geometry-aware learning of maps for camera localization. In *CVPR*, 2018. 1, 3

[5] Federico Camposeco, Andrea Cohen, Marc Pollefeys, and Torsten Sattler. Hybrid camera pose estimation. In *CVPR*, 2018. 3

[6] Robert Castle, Georg Klein, and David W Murray. Video-rate localization in multiple maps for wearable augmented reality. In *International Symposium on Wearable Computers*, 2008. 1

[7] Ronald Clark, Sen Wang, Andrew Markham, Niki Trigoni, and Hongkai Wen. Vidloc: A deep spatio-temporal model for 6-dof video-clip relocalization. In *CVPR*, 2017. 1, 3

[8] Mark Cummins and Paul Newman. FAB-MAP: Probabilistic localization and mapping in the space of appearance. *IJRR*, 2008. 1

[9] Rıza Alp Güler, Natalia Neverova, and Iasonas Kokkinos. Densepose: Dense human pose estimation in the wild. In *CVPR*, 2018. 2, 4

[10] Qiang Hao, Rui Cai, Zhiwei Li, Lei Zhang, Yanwei Pang, and Feng Wu. 3D visual phrases for landmark recognition. In *CVPR*, 2012. 1, 3

[11] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *ICCV*, 2017. 1, 2, 4

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 4

[13] Ronghang Hu, Piotr Dollár, Kaiming He, Trevor Darrell, and Ross Girshick. Learning to segment every thing. In *CVPR*, 2018. 2

[14] Alex Kendall and Roberto Cipolla. Modelling uncertainty in deep learning for camera relocalization. In *ICRA*, 2016. 3

[15] Alex Kendall and Roberto Cipolla. Geometric loss functions for camera pose regression with deep learning. In *CVPR*, 2017. 1, 3, 6, 7

[16] Alex Kendall, Matthew Grimes, and Roberto Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *ICCV*, 2015. 1, 3

[17] Yunpeng Li, Noah Snavely, and Dan Huttenlocher. Location recognition using prioritized feature matching. In *ECCV*, 2010. 1, 3

[18] Yunpeng Li, Noah Snavely, Dan Huttenlocher, and Pascal Fua. Worldwide pose estimation using 3D point clouds. In *ECCV*, 2012. 1, 3

[19] Tsung-Yi Lin, Piotr Dollár, Ross B Girshick, Kaiming He, Bharath Hariharan, and Serge J Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 4

[20] Manolis Lourakis and Antonis Argyros. The design and implementation of a generic sparse bundle adjustment software package based on the levenberg-marquardt algorithm. Technical report, Institute of Computer Science-FORTH, Heraklion, Crete , 2004. 8

[21] David G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004. 3, 5

[22] Simon Lynen, Torsten Sattler, Michael Bosse, Joel A Hesch, Marc Pollefeys, and Roland Siegwart. Get out of my lab: Large-scale, real-time visual-inertial localization. In *Robotics: Science and Systems*, 2015. 1

[23] Iaroslav Melekhov, Juha Ylioinas, Juho Kannala, and Esa Rahtu. Image-based localization using hourglass networks. *ICCV workshops*, 2017. 3

[24] Pierre Moulon, Pascal Monasse, and Renaud Marlet. Global fusion of relative motions for robust, accurate and scalable structure from motion. In *ICCV*, 2013. 1, 3

[25] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NIPS*, 2015. 4

[26] Torsten Sattler, Bastian Leibe, and Leif Kobbelt. Improving image-based localization by active correspondence search. In *ECCV*, 2012. 1, 3

[27] Torsten Sattler, Bastian Leibe, and Leif Kobbelt. Efficient & effective prioritized matching for large-scale image-based localization. *IEEE Trans. PAMI*, 2017. 1, 3

[28] Torsten Sattler, Akihiko Torii, Josef Sivic, Marc Pollefeys, Hajime Taira, Masatoshi Okutomi, and Tomas Pajdla. Are large-scale 3D models really necessary for accurate visual localization? In *CVPR*, 2017. 3

[29] Grant Schindler, Matthew Brown, and Richard Szeliski. City-scale location recognition. In *CVPR*, 2007. 1, 3

[30] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016. 4, 6

[31] Johannes Lutz Schönberger, True Price, Torsten Sattler, Jan-Michael Frahm, and Marc Pollefeys. A vote-and-verify strategy for fast spatial verification in image retrieval. In *ACCV*, 2016. 6

[32] Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in RGB-D images. In *CVPR*, 2013. 3

[33] Xun Sun, Yuanfan Xie, Pei Luo, and Liang Wang. A dataset for benchmarking image-based localization. In *CVPR*, 2017. 2, 6, 8

[34] Linus Svärm, Olof Enqvist, Magnus Oskarsson, and Fredrik Kahl. Accurate localization and pose estimation for large 3D models. In *CVPR*, 2014. 1, 3

[35] Hajime Taira, Masatoshi Okutomi, Torsten Sattler, Mircea Cimpoi, Marc Pollefeys, Josef Sivic, Tomas Pajdla, and Akihiko Torii. Inloc: Indoor visual localization with dense matching and view synthesis. In *CVPR*, 2018. 1, 3

[36] Akihiko Torii, Josef Sivic, Masatoshi Okutomi, and Thomas Pajdla. Visual place recognition with repetitive structures. *IEEE Trans. PAMI*, 2015. 3

[37] Akihiko Torii, Josef Sivic, and Thomas Pajdla. Visual localization by linear combination of image descriptors. In *ICCV Workshop*, 2011. 3

[38] Julien Valentin, Matthias Nießner, Jamie Shotton, Andrew Fitzgibbon, Shahram Izadi, and Philip HS Torr. Exploiting uncertainty in regression forests for accurate camera relocalization. In *CVPR*, 2015. 3

[39] Florian Walch, Caner Hazirbas, Laura Leal-Taixe, Torsten Sattler, Sebastian Hilsenbeck, and Daniel Cremers. Image-based localization using LSTMs for structured feature correlation. In *ICCV*, 2017. 1, 3

[40] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *CVPR*, 2017. 4

[41] Bernhard Zeisl, Torsten Sattler, and Marc Pollefeys. Camera pose voting for largescale image-based localization. In *ICCV*, 2015. 3

[42] Wei Zhang and Jana Kosecka. Image based localization in urban environments. In *International Symposium on on 3D Data Processing, Visualization, and Transmission (3DPVT)*, 2006. 3