Vision-Based Image Retrieval for Localization

Master Project
Deparment of Informatics

Students

Abhinav Aggarwal (18-748-079) & Mesut Ceylan (18-748-012)
(abhinav.aggarwal@uzh.ch, mesut.ceylan@uzh.ch)

Supervisors

Dr. Danda Paudel, Janine Thoma, Prof. Luc Van Gool, Prof. Davide Scaramuzza
(paudel@vision.ee.ethz.ch, jthoma@vision.ee.ethz.ch,
vangool@vision.ee.ethz.ch, sdavide@ifi.uzh.ch)

Department of Informatics
University of Zurich
Switzerland

01.04.2020

Distance and Orientation Disentangled Image Features for Retrieval-based Localization

Abhinav Aggarwal*†, Mesut Ceylan*†, Janine Thoma†, Danda Pani Paudel†, Luc Van Gool†‡
*Equal contribution, email: {abhinav.aggarwal, mesut.ceylan}@uzh.ch
†Computer Vision Lab, ETH Zurich, Switzerland
‡Katholieke Universiteit Leuven, Belgium

***Abstract*—Accurate localization is an essential task of mobile agents, such as autonomous cars or flying drones, for navigating through an environment and performing various functions. Localization by image retrieval is a promising vision-based solution due to its robustness and efficiency. Retrieval-based localization matches any query image of the environment to a set of reference images with known geographic information. The matching is usually based on a image feature embedding which is learned such that geometrically nearby images are also nearby in the feature space. In this paper, we propose a method for learning an image feature embedding which is capable of predicting both the geometric distance and the relative angle between the query and the retrieved image, while also improving the retrieval accuracy. Being able to predict distance and angle based on the distance between image features, allows to accurately calculate the location of a given query image. Our embedding is learned using a CNN that maps an image into a corresponding feature which consists of two disentangled parts. These two parts are separately responsible for distance and orientation aspects. Furthermore, we also propose a new loss formulation that exploits all available positive examples for higher retrieval accuracy. The experimental results on two benchmark datasets demonstrate the superiority of our method over the state-of-the-art.**

I. Introduction

Accurate localization is an essential task of mobile agents, such as humans, autonomous cars or flying drones, for navigating through the environment or in performing the intended assignments. In an outdoor environment, existing localization systems largely benefit from GPS information. However, solutions based on GPS alone are not suitable in cases requiring; higher localization accuracy; orientation; operation in a GPS denied environment such as indoors. Under such conditions, vision-based approaches offer promising solutions [1]. Vision-based localization [2] methods can broadly be classified into: (a) structure-based, such as Structure-from-Motion (SfM) [3], [4] and Simultaneous Localization and Mapping (SLAM) [5], [6]; (b) retrieval-based [7], [8]. The structure-based methods primarily focus on accurate relative localization, whereas, retrieval-based methods emphasize re-localization. At some cost of accuracy, the latter method is robust and salable due to its memory and computational efficiency.

Retrieval-based localization includes the process of matching any query image taken in an environment to a set of reference images whose geographic information is known. With the recent advancements on Convolutional Neural Networks (CNNs) and its capability over learning image features [9], [10], [11], [12], CNNs haven been used successfully for image matching problem [1]. In addition, CNN-based methods depict immense capability of learning features that are robust under strong appearance, illumination, and viewpoint changes [8], [7]. This has become possible with the availability of comprehensive localization datasets such as Oxford RobotCar [13] and Pittsburg 250k [14], COLD [15], IDOL [16], USC [17] and the Place Recognition Dataset [18].

Most of the existing methods for retrieval-based localization focus on place/landmark recognition [7], [19], [20], [21], where the features learned to recognize places are shown to serve for the task of image matching. However, features for place recognition are not often an ideal choice for accurate localization [8]. Beside, the compromise in localization accuracy, of the retrieval-based methods, also come from the experimental setup, where the location/pose of the query image is assumed to be the same as that of the best matching reference image. When a query image is captured from a different location than any of the reference images, the naive retrieval-based localization becomes inaccurate as an obvious consequence. To address this issue, *Thoma et al.* [1] developed an image retrieval method that also provides distance of the query image from the best matching reference one. This is carried out by learning feature embedding such that the feature distances between two images also represent the geometric distance between them. Doing so, it does not only offer more information with respect to the retrieved image, but also improves the retrieval accuracy.

The method developed in [1] provides only the geometric distances between images – which when used together with multiple reference images can predict the location of the query image – and cannot infer the camera orientation. In this work, we aim to learn the image feature embedding that also reveals the orientation of the camera with respect to the retrieved reference image. Being able to predict both distance and angle, directly by comparing image features, offers us fully localized images as well as improvement in the retrieval accuracy. In contrast to the pose regression methods [22], [23], learning feature embeddings that offer orientation and distance between images have potential to generalize and scale better. The generalizability and scalability are respectively due to map agnostic learning and offline created feature bank for retrieval.

In this paper, we proposed a method to learn image feature embedding capable of predicting both geometric distance and the relative angle between query and retrieved images, along

with features serving for retrieval. Such embedding is learned using a CNN that maps an image into its corresponding feature disentangled into two parts. These two parts are separately responsible for distance and orientation aspects. Our learning process enforces, for features of any two nearby images, distance between first feature segments to be proportional to the geometric distance. Similarly, distance between the second feature segments is proportional to the relative orientation. Feature disentanglement in this manner leads to more precise localization of query images both in terms of metric and the accuracy. To further improve the localization accuracy, we also propose a loss formulation that uses all available positive examples. The all positive loss term encourages the closer positives to play higher roles, by weighting the positive losses based on their proximity. The major contributions of this paper can be summarized as follows,

- A new consideration while learning image retrieval features capable of providing the camera orientations.
- Distance and orientation disentangled features for complete, in terms of 2D pose, and accurate localization.
- Proximity based weighted all positives loss term.

II. Related Work

Methods for vision-based localization [2] can be divided into retrieval-based approaches [20], [7], [24], [25], [8], [26] and structure-based approaches, such as Structure-from-Motion (SfM) [3], [27], [28], [29], [30] and Simultaneous Localization and Mapping (SLAM) [5], [31], [32], [6], [33]. In comparison to structure-based methods, retrieval-based methods profit form simpler mapping and matching, making them more scalable [7], [4], [34]. However, without additional refinement, the minimal pose estimation error for retrieval-based methods is given by the distance between query and nearest reference image, while structure-based methods allow for more accurate 6DOF relative pose calculation. This has lead to the rise of hybrid approaches with retrieval-based initialization and structure-based refinement [28]. In this paper, we address localization as an image retrieval problem. By encoding geometric and angle distances in the feature space, it becomes possible to predict relative poses more accurately than common retrieval-based approaches but without the cost of an additional structure-based refinement step. In the following we briefly summarize the most relevant related work on the topic of image retrieval-based localization. A more complete overview can be found in [35], [36].

Prior to the emergence of deep learning, [37] localized objects in videos by comparing a query image to a database of videos exploiting the Bag of Words representation. Similarly, [38] developed a mobile telephone application for localization based on a wide-baseline image matching algorithm. Inspired by [37], [39] defined a hierarchical vocabulary tree for a more efficient lookup of visual words. [40] introduced a fast spatial matching step to further improve retrieval results.

More recently, convolutional neural networks have become highly successful in a large number of computer vision tasks, including image retrieval [41]. Thus, the focus of research within the scope of image retrieval-based localization has shifted towards finding well-tailored neural network architectures [41], [42], varying pooling operations [43], [44], [7], [45], [25], [21] and novel loss functions [46], [1], [7], [47] for learning suitable features and image representations. First works showed that descriptors obtained from the top levels of large convolutional neural networks neural networks along with dimensionality reduction are highly useful for image retrieval [41] and indoor image matching [42]. [43] highlighted the benefit of max-pooling for particular object retrieval, exploiting the work of [44]. [45] extended the R-MAC descriptors of [43], integrating them into a single neural network, which they then trained for image retrieval using the Triplet loss [9]. [25] added further improvements on top of [45], specifically targeting landmark recognition. Similarly, [7] extended neural networks with a trainable VLAD layer, resulting in a significant performance boost for place recognition tasks when trained with weakly supervised Triplet loss. [21] trained a network to discard NetVLAD [7] features that are irrelevant for localization. The success of Triplet loss [9] has inspired the creation of many extensions, such as Quadruplet loss [47] and many others.

Our work is most similar to [1], who proposed a novel loss function that associates images both geometrically and visually and obtained notably better localization accuracy as well as geometrically mappable image features. The key differences between our work and [1] are the introduction of a novel loss function, the prediction of angle differences and the extension of our network architecture for feature disentanglement. These extensions further improve image-based retrieval accuracy. Somewhat similar to both [1] and our approach, [48] learn features whose distances correlate to the amount of co-visible points shared between two images. While co-visibility is a function of both distance and angle, it is not directly possible to calculate one from the other. Also, [48] require a complete 3D reconstruction to calculate their loss, something which our method does not need.

III. Problem Definition

We denote a set of geo-tagged reference images by $\{I_i\}_{i=1}^n$. whose 2D coordinates $\{x_i \in \mathbb{R}^2\}$ and angle $\alpha_i \in \mathbb{R}$ for all reference images. With the help of reference images, we wish to localize any given query image I_q. We have tackled this problem by creating a mapping function $\Gamma_\phi : I_i \mapsto f_i$ such that $d(f_q, f_*) < d(f_q, f_i)\ \forall\{i \setminus *\}$ where d is standard euclidean distance between two feature vectors and p is most similar image compared to query image. We model the mapping function Γ_ϕ using a CNN, whose parameters ϕ are learned using a supervised training process. For training, we make use of a set of image pairs $\mathscr{S} = \{\mathscr{S}_{ij}\} = \{(I_i, I_j)\}_{i \neq j}$ which are divided into sets of positive and negative pairs, say respectively $\mathscr{S}^+$ and $\mathscr{S}^-$, based on the location and orientation proximity, as per the definition,

$$\mathscr{S}^+ = \{\mathscr{S}_{ij} : \mathbf{d}(x_i, x_j) \leq r_1 \wedge \mathbf{a}(\alpha_i, \alpha_j) \leq \theta\}, \qquad (1)$$

$$\mathscr{S}^- = \{\mathscr{S}_{ij} : \mathbf{d}(x_i, x_j) \geq r_2\}. \quad (2)$$

where $\mathbf{d}(.)$ is the euclidean distance and $\mathbf{a}(.)$ is the corrected absolute difference between two angles defined as,

$$\mathbf{a}(\alpha_i, \alpha_j) = \begin{cases} |\alpha_i - \alpha_j|, & \text{if } |\alpha_i - \alpha_j| \leq \pi, \\ 2\pi - |\alpha_i - \alpha_j|, & \text{if } |\alpha_i - \alpha_j| > \pi. \end{cases} \quad (3)$$

Note that two images of any positive pair are within a predefined r_1 distance apart and also have angle difference smaller than θ. Similarly, two images of any negative pair lie outside the radius of r_2, where $r_1 < r_2$. Using a given set of image pairs $\mathscr{S} := \mathscr{S}^+ \cup \mathscr{S}^-$, we wish to learn a mapping function $\Gamma_\phi{:}I_i{\mapsto}f_i$, such that the mapped features are disentangled in a manner $f_i = [(f_i^1)^\top, (f_i^2)^\top]^\top$, separately responsible for distance and angle measure. In this context, the mapping function is desired to the following properties,

$$\Gamma_\phi{:}I_i{\mapsto}f_i{:=}\begin{cases} \mathbf{d}(x_i, x_j) \propto \mathbf{d}(f_i^1, f_j^1) \text{ and} & \\ \mathbf{a}(\alpha_i, \alpha_j) \propto \mathbf{d}(f_i^2, f_j^2), & \text{if } \mathscr{S}_{ij} \in \mathscr{S}^+, \\ \\ \mathbf{d}(f_j, f_k) > \mathbf{d}(f_i, f_j){+}\delta, & \text{if } \mathscr{S}_{ij} \in \mathscr{S}^+ \text{ and } \mathscr{S}_{jk} \in \mathscr{S}^-. \end{cases} \quad (4)$$

where δ is the safety margin which ensures that the negatives are at least a margin away when compared to the positives.

IV. Feature Learning

In this section, we present our method for learning distance and orientation disentangled image features for retrieval-based localization. In the following, we will first discuss the proposed Disentangler Convolutional Neural Network and followed by the loss function used to learn mapping function Γ_ϕ that maps images to the sought feature vectors.

A. Disentangler Network

The proposed network architecture, illustrated in Fig 1, consists of three main parts: (i) VGG-16 [19] network to generate intermediate feature map corresponding to input image, (ii) NetVLAD [7] layer to aggregate the intermediate feature map to global descriptors and (iii) Disentangler Network to segregate the global feature space into two parts.

For the first two parts of our network, we use partial VGG-16 (till conv5 layer) in combination with NetVLAD layer [50], similarly as in [1]. The third part of the network, namely the Disentangler, consists of multiple fully connected layers right after NetVLAD layer. However in our experiments, the fully connected alone for the Disentangler network did not serve its purpose of segregation. Therefore, motivated from [11], we added residual connection from NetVLAD layer output to disentangler network output. This connection eases the training process and as well as improves the generalization capability of our network. The output of the disentangler network is spitted equally into two halves, which are then normalized to the unit length independently, and used to derive the loss terms for distance and angle as discussed below.

B. Loss Functions

We train our network using a loss function that consists three different terms; Negative Visual Loss, Distance Loss, and Angle Loss. Here, we discuss these three losses in details.

*1) **Negative Visual Loss**:* The main goal of this loss function is to learn image features such that Euclidean distance between negative pairs should be greater than positive pair distance. We experiment with two variants which enforce this property. Please recall (4), the negative visual loss considers the second criteria of the mapping function. Two variants to we considered are Triplet loss as in [7], and the lazy Quadruplet loss [46]. Here we explain negative visual loss with the help of the Triplet formulation. Interested reader can find the details regarding the lazy Quadruple in [46]. In the Triplet setup, we modified the negative visual loss of [1], by introducing a weight factor as,

$$H(\mathscr{S}_{ij}, \mathscr{S}_{jk}) = l(w_i \mathbf{d}(f_i, f_j) + \delta - \mathbf{d}(f_j, f_k)), \quad (5)$$

where l is hinge loss, I_j is query image, I_i is positive image and I_k is negative image as per the definition (1). Here we use weighted all positives as opposed to best positive training used in [1]. This change is highlighted by w_i parameter in Eq (5) which represents weights assigned to different positives. The weight w_i is derived using,

$$w_i = \frac{1}{Z_i} \frac{1}{\mathbf{d}(x_i, x_j)}, \quad (6)$$

where Z_i is normalizing factor and $\frac{1}{d(x_i,x_j)}$ is euclidean distance between query image and positive image. The motivation for using the weighted positive loss comes from the fact that positive images usually share significant co-visibility with query image by definition (1). Finally by adding Hinge loss function for all combinations of positive and negative, we define negative visual loss as following:

$$L_{NV} = \sum_k \sum_i H(\mathscr{S}_{ij}, \mathscr{S}_{jk}). \quad (7)$$

.

*2) **Distance Loss**:* Recall (4), where we aim to enforce proportionality between geometric and feature distances. This constraint however is meaningful only for positive pairs as the negatives may not offer enough information to do the same. Therefore, we utilize modified version of [1]'s visual geometric loss function:

$$L_{VG} = \sum_{\mathscr{S}_{ij} \in \mathscr{S}^+} \rho(\mathbf{d}(x_i, x_j) - \lambda_1 \mathbf{d}({f_i}^1, {f_j}^1)) \quad (8)$$

where ρ is Huber loss function and λ_1 is the proportionality constant. Note that we use only the first half of feature space as opposed to the full feature vector used by [1].

*3) **Angle Loss**:* In conjunction with the adapted Negative Visual and Distance losses, we introduce the angle loss to benefit from angle information. This term indeed comes from the angle constraint of (4). Here again, we use the angle proportionality constraint only for the positive pairs, as it may

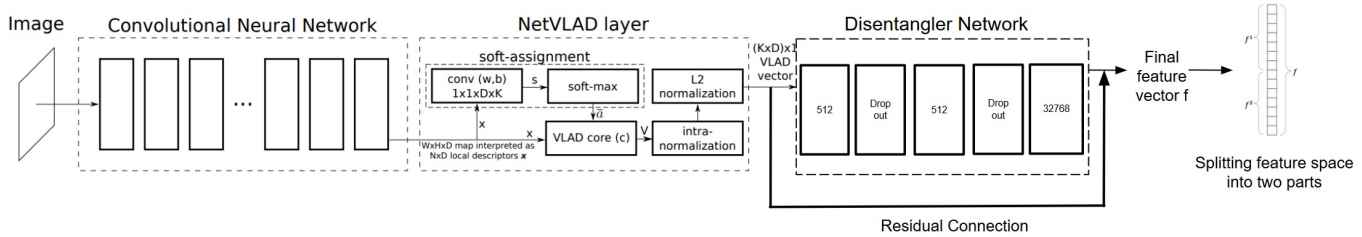

Fig. 1. Network Architecture for Disentangler Network and Feature Splitting. Proposed network consists of three main parts, (i) VGG-16 [19], (ii) NetVLAD [7] and (iii) Disentangler Network formed by inspiration from [49]. On top of fully connected layers, we implement residual connection from NetVLAD layer to outcome of disentangler network. Further on, final feature vector f is divided into two parts and dedicated to respective loss functions.

be unrealistic to expect the same from the negative pairs. We formulated the angle loss function as following:

$$L_A = \sum_{\mathscr{S}_{ij} \in \mathscr{S}^+} \rho(\mathbf{a}(\alpha_i, \alpha_j) - \lambda_2 \mathbf{d}({f_i}^2, {f_j}^2)) \tag{9}$$

where ρ is Huber loss function, λ_2 is the proportionality constant, $\mathbf{a}$ is angle correction function explained in Eq (3) and f_*^2 is the second half of feature space.

*4) **Final Loss**:* The final loss function that we use to train our network is a simple combination of the three loss terms discussed above, which is given by,

$$L_F = L_{NV} + \eta L_A + \beta L_{VG} \tag{10}$$

where η and β are scaling factors that balance different loss terms, otherwise also know as hyper-parameters.

V. Experiments

We describe the used datasets and evaluation methodology in Section V-A, training details in Section V-B, our reference image subsampling strategy in Section V-C, angle distance linearity in Section V-D and provide quantitative and qualitative results in Sections V-E and V-F to validate our approach.

A. Benchmark Datasets

We conduct our experiments on two publicly available datasets: (i) the Oxford RobotCar dataset [13] and (ii) the COLD dataset [15] belonging to outdoor and indoor environments respectively. For the Oxford RobotCar dataset, we use the first *1.5 km* of [a] *2014-11-18-13-20-12* (clouds, sunny), [b] *2014-11-21-16-07-03* (dusk, rain), [c] *2014-12-02-15-30-08* (overcast) and [d] *2014-12-16-18-44-24* (night) as train reference images and the next *0.5 km* as test reference images. Similarly, we use the first *1.5 km* of [e] *2015-02-13-09-16-26* (overcast), [f] *2015-03-03-11-31-36* (sun) and [g] *2015-10-29-12-18-17* (rain) as train query images and the next *0.5 km* as test query images. Specifically, we use only those tracks in which RTK Ground Truth [51] is available. This selection criterion leads to 23500 train reference images, 19001 train query , 4916 test reference and 3273 test query images.

With regard to the COLD dataset, we focus on Freiburg section of the dataset and utilize Path A for training purposes and Path B for test purposes. Within Path A, we use all sequences and within each sequence we allocate sets 1,2 and 4 to the train reference while we use set 3 as train query images. To test the generalization capabilities, we dedicate Path B to testing. Specifically, we test using Sequence 3, using set 1 and set 2 as test reference images while set 3 is used as test query images. As a result, we ensure that test images are never seen before. Overall, we obtain 29237 train reference, 12831 train query, 7795 test reference and 4123 test query images.

B. Training

We initialize the weights up to the NetVLAD layer using the checkpoints of [7] trained on 30,000 images from Pittsburgh 250k [14]. We follow the training methodology of [1], setting the queries per batch to two and positives and negatives per query to six. We use a decaying learning rate with initial value 0.00005 and a decay 0.5 epoch-wise.

To sample positive and negative images, we use different parameters for the COLD Freiburg and the Oxford dataset due to their different scales. For the COLD Freiburg dataset, we set r_1 to 1 meter and r_2 to 4 meters, similar as in [1]. For the Oxford RobotCar dataset, we set r_1 to 10 meters and r_2 to 25 meters accordance to [7]. For both datasets, α is set to $60°$ such that sampled positive images also carry overlapping visibility. In addition, we sample hard positives from the set of positives images $\mathscr{S}^+$ such that they share only partial co-visibility with query images. In case of hard positives, we set the angle difference θ between two images as $30° - 60°$.

Overall, the training of one network takes 48 hours for both datasets. For COLD Freiburg, this training duration results in nine epochs while for Oxford RobotCar dataset it results in seven epochs. We only use train reference images while validation is conducted using only train query images.

C. Reference Images Sampling Strategy

Reference image selection from database images plays a key role in image retrieval task, since densely distributed reference image map might not be always available. Also utilization of densely distributed maps do not imitate real-life scenarios as it is not always possible to collect such a huge amount of reference images due to memory constraints. Therefore, sampling the right reference images and a sufficient amount of them to create the whole map is essential. Our disentangler network is capable of retrieving correct image in sparse image

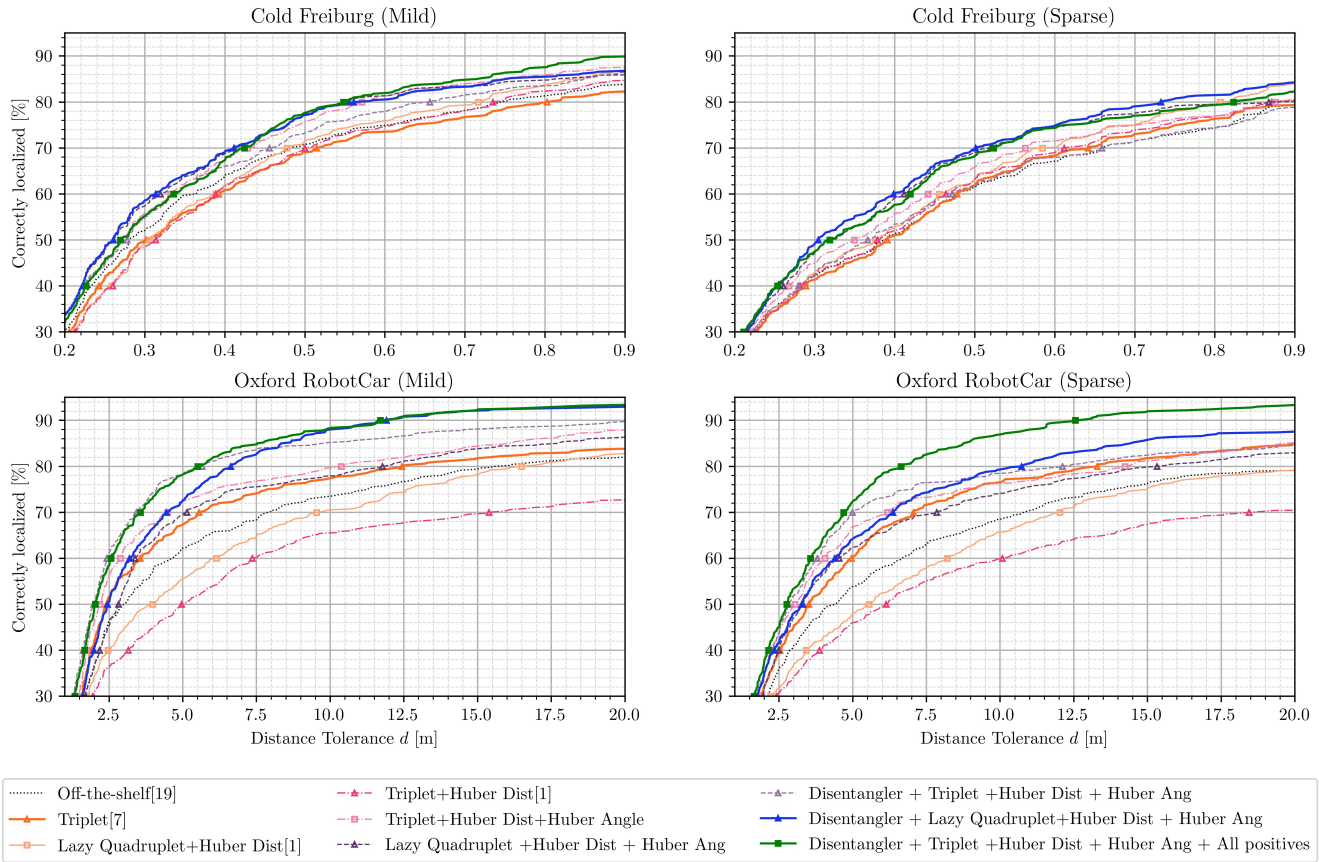

Fig. 2. Localization accuracy as function of distance tolerance

maps. To emulate a sparse image map, we introduce a novel sampling strategy that ensures mapping a complete indoor environment by using two sets of parameters. The sampling strategy segregates the number of sampled images according to agents movement. The fundamental idea is to sparsely sample reference images where overlapping between subsequent image is high while sample densely where overlapping is low. With this approach, sampling occurs sparsely during the linear motion of the agent, while sampling occurs densely during the rotational motion. We used recursive methodology using *Frame Threshold* ω and *Angle Threshold* γ to control linear and rotational subsampling respectively. To this end, the aim is to sample minimal reference images to map the whole reference map such that the number of sampled images is low.

D. Angle Distance Linearity

As it is mentioned earlier, the disentangler network boosts linearity between angle and feature distance as well as linearity between geometric distance and feature distance by segregating feature space into two variants. To observe the effect of the new network on the linearity, one can compute Pearson‘s correlation w.r.t to angle difference and geometric distance against feature distance. Fig 3. clearly proves that the disentangler network and involvement of orientation information in our model provides significant improvement on angle as well as distance linearity. Triplet [7] loss results in Pearson‘s correlation value of 0.27 between angle difference and feature distance and 0.51 between geometric distance and feature distance while we obtain Pearson‘s correlation value of 0.47 between angle difference and feature distance and 0.71 between geometric distance and feature distance by utilizing our disentangler network and angle loss.

E. Quantitative Localization Performance

For quantitative measure and to compare with state-of-the-art results, we use the same performance measurement metrics as [1]. However, the positives are defined while also taking angles into consideration. The definition of an image being correctly localized therefore is amended: it must not only fall within a certain radius but also the angle difference between query and reference should not be greater than 60° for COLD Freiburg and 30° for Oxford Robotcar. The latter dataset generally has much smaller angle deviations in comparison to distance travelled, justifying the smaller threshold.

We conduct experiments with two different subsampling methodologies, namely mild subsampling in which the frame threshold is set to ω=32 and the angle threshold is set to γ=0.3 and sparse subsampling in which the frame threshold is set to ω=48 and the angle threshold is set to γ=0.6. For the COLD Freiburg dataset, results are computed by

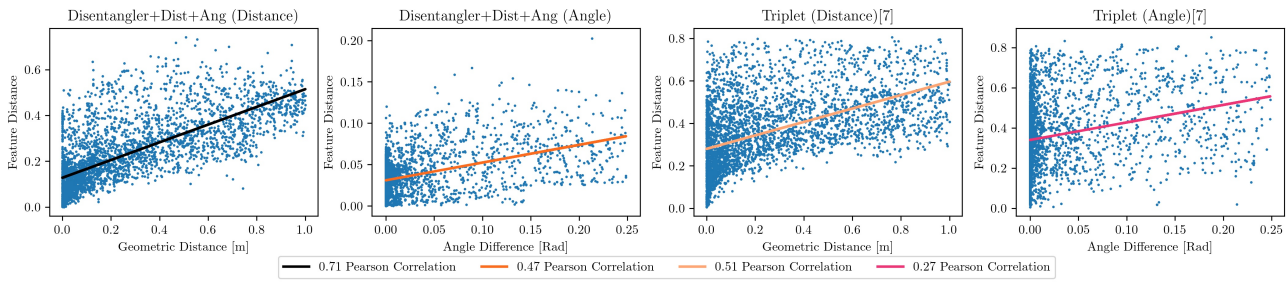

Fig. 3. Pearson's correlation values w.r.t angle difference and geometric distance against feature distance. First two plots, from the left, illustrate the outcome of our model (Disentangler Network with Angle Loss), where as latter two are the outcomes of the Base NetVLAD network [7]. Higher the better.

TABLE I
LOCALIZATION ACCURACY COMPARISON FOR **COLD FREIBURG** AND **OXFORD ROBOTCAR**.
NOTE THAT LOSSES STATED IN BOLD FONTS ARE OUR LOSS FUNCTION VARIATIONS.

	COLD FreiburgCOLD Freiburg				Oxford RobotcarOxford Robotcar			
Distance Threshold	0.5m/1m	0.5m/1m	0.5m/1m	0.5m/1m	10m/25m	10m/25m	10m/25m	10m/25m
Angle Constraint	-	(-30°, 30°)	-	(-30°, 30°)	-	(-15°, 15°)	-	(-15°, 15°)
Subsampling Parameters	ω=32, γ=0.3	ω=32, γ=0.3	ω=48, γ=0.6	ω=48, γ=0.6	ω=10, γ=0.3	ω=10, γ=0.3	ω=20, γ=0.3	ω=20, γ=0.3
Off-the-shelf [19]	0.71/0.85	0.70/0.82	0.62/0.81	0.61/0.78	0.74/0.83	0.73/0.81	0.69/0.79	0.68/0.78
Triplet [7]	0.69/0.84	0.69/0.83	0.62/0.81	0.62/0.79	0.78/0.85	0.77/0.85	0.77/0.85	0.76/0.84
Triplet+Huber Distance [1]	0.70/0.86	0.70/0.85	0.63/0.82	0.62/0.80	0.66/0.75	0.65/0.74	0.60/0.72	0.60/0.72
Lazy Quadruplet+Huber Distance [1]	0.72/0.89	0.72/**0.88**	0.64/**0.86**	0.63/**0.85**	0.71/0.86	0.70/0.86	0.66/0.82	0.66/0.82
Triplet + Huber Distance + Huber Angle	0.76/0.89	0.75/0.87	0.66/0.82	0.65/0.80	0.80/0.89	0.80/0.89	0.76/0.87	0.76/0.86
Disentangler + Triplet + Huber Distance + Huber Angle	0.73/0.88	0.73/0.86	0.62/0.82	0.62/0.79	0.85/0.92	0.85/0.92	0.79/0.86	0.78/0.86
Lazy Quadruplet + Huber Distance + Huber Angle	0.77/0.87	0.76/0.85	0.69/0.81	0.68/0.80	0.78/0.88	0.78/0.88	0.74/0.85	0.74/0.85
Disentangler + Lazy Quadruplet + Huber Dist + Huber Angle	0.77/0.87	**0.77**/0.86	**0.70**/0.85	**0.69**/0.84	**0.88**/0.94	**0.88/0.94**	0.79/0.89	0.79/0.87
Disentangler + Triplet + Huber Distance + Huber Angle + All positives	**0.79/0.91**	**0.77/0.88**	0.69/0.84	0.66/0.80	**0.88/0.95**	**0.88**/0.93	**0.87/0.95**	**0.87/0.93**

comparing all test images to their closest 97 and 183 reference images separately which are subsampled according to sparse and mild subsampling methodologies respectively. For Oxford RobotCar dataset, we tested all test images from all weather conditions by matching them with the closest 68 sparsely and 135 mildly subsampled reference images.

Table I shows the exact localization accuracy results of our method with respect to different loss functions and datasets. Note that for the Off-the-shelf loss, we use the best weights provided by [7], while for the Triplet, Triplet + Huber Distance and the Lazy Quadruplet + Huber Distance losses we use the weights trained according to the procedure of [1]. Within the table, we report the localization accuracy results for a distance threshold of 0.5m and 1m for COLD Freiburg and 10m and 25m for Oxford RobotCar. We also include results with and without an angle constraints. An angle constraint of $-n°$ to $+n°$ means that retrieved images are only considered correctly localized if the angle difference between query image and retrieved reference lies in the range of $-n°$ to $+n°$. The first two columns of each dataset show the results for mild sampling, while the third and fourth column per dataset show the results for sparse sampling.

*1) **Angle Loss**:* Clearly, enforcing angle proportionality on top of geometric-visual proportionality boosts the model to result in better localization w.r.t angle as well as geometric distance. It is proven that our novel angle loss function results in significant improvements over state-of-the-art on both datasets when angle loss is combined with Triplet or Lazy Quadruplet loss. On top of that, while it performs almost as good as state-of-the-art within 1m distance range, our loss functions provide accuracy improvement especially within 0.5m distance threshold on indoor dataset COLD Freiburg where short range sensitivity is crucial for localization. For Oxford RobotCar dataset, one can observe the same effect of our loss, resulting accuracy improvement both in sparse and mild sampling conditions and outperforming state-of-the-art.

*2) **Disentangler Network**:* Furthermore, one can observe contributions of disentangler network for further significant improvements on accuracy compared to all models specifically in Oxford RobotCar dataset. Although it performs same or provide slight contribution to accuracy in some cases in COLD Freiburg dataset, our disentangler network steps forward distinctively when feature variety rich datasets are used.

*3) **Weighted All Positive Loss Term**:* Utilization of weighted all positive loss term is yet another factor for higher accuracy. In addition to that, the term also brings robustness to our model. When accuracies corresponding to different sampling strategies compared within Oxford RobotCar dataset, we observe accuracy drop in all models when sparse sampling applied except the last model that carries the term.

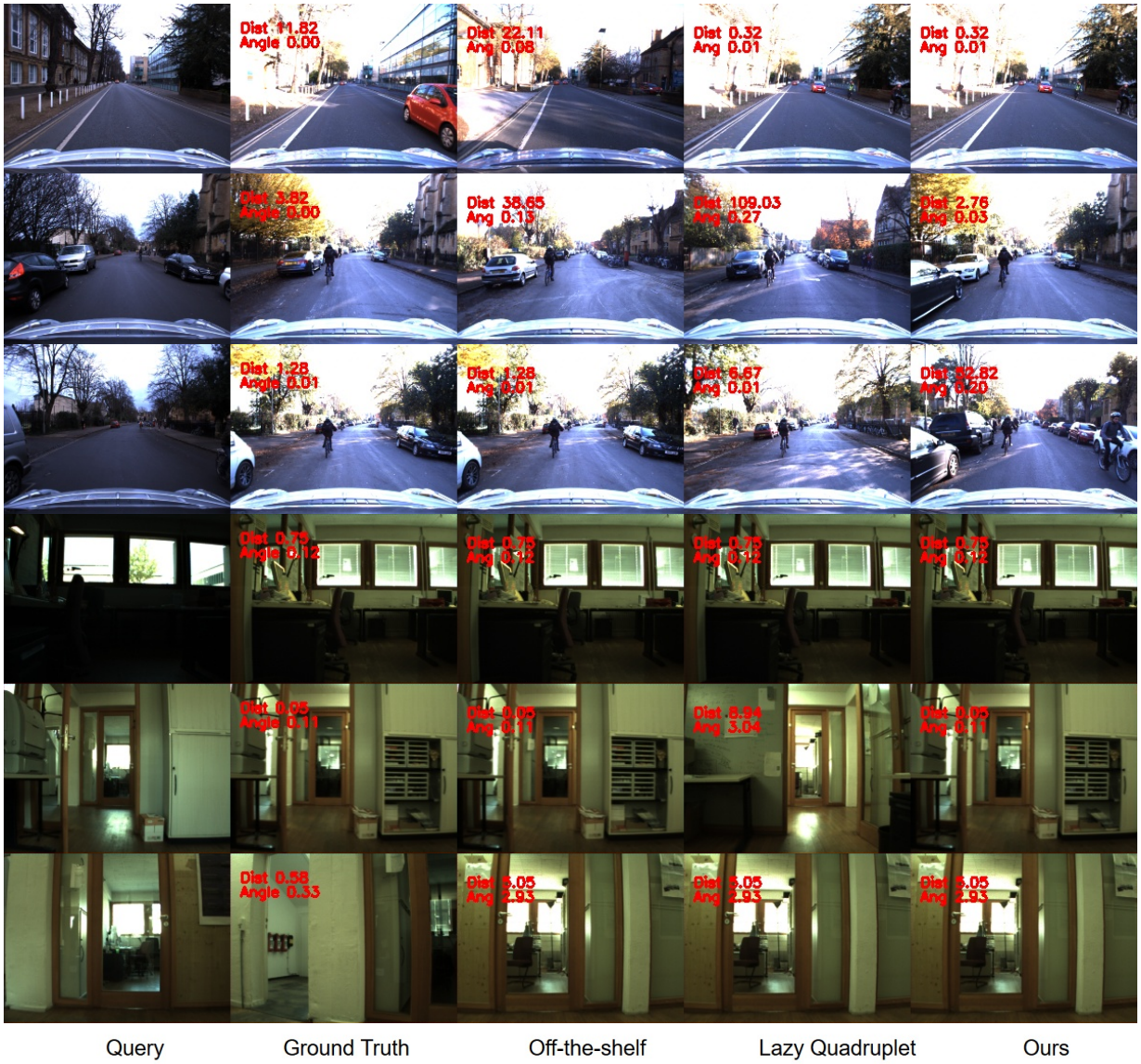

Fig. 4. Visual results from two benchmark datasets, Oxford RobotCar (top) and COLD Freiburg (bottom) datasets. Ground truth is computed as following: Top-3 closest images w.r.t geometric distance to query image are fetched and final ground truth image is chosen as the one that has the least angle distance w.r.t.query image. Images carries distance and angle tags w.r.t query image. Comparison elements: Off-the-shelf [19], Lazy Quadruplet [1], and Ours (Disentangler Network with Huber Angle Loss with Weighted All Positive Loss Term)

F. Qualitative Performance

In Fig. 4 we show selected visual examples of the retrieval performance of different models. From left to right, each row shows a query image, followed by the closest ground truth reference image, the reference image retrieved with off-the-shelf features, lazy quadruplet loss [1] and our novel loss function. Compared to the lazy quadruplet and off-the-shelf models, our model is robust to strong illumination changes, such as dim lighting and sun light glares, ase well as scene changes, such as an open document shelf, which was previously closed.

VI. Conclusion

In this work, we introduced an efficient and robust vision-based solution for retrieval-based localization. We designed a novel angle loss function, a disentangler network and an all positive loss term to ensure proportionality among orientation, geometric and feature distance. The resulting models successfully improved image retrieval-based localization. We showed that our disentanglement architecture and training make it possible to produce features which separate the original distance and angle relations, theoretically allowing for a more precise calculation of localization, in comparison to traditional retrieval based methods. Furthermore, our disentangled feature embedding with its inclusion of orientation information significantly improved localization accuracy within our experimental setup on two publicly available different benchmark datasets. In conclusion, our training methodology—including

angle loss, disentangler network and weighted all positive loss—outperformed the state-of-the-art in in retrieval-based localization in a clear majority of our experiments.

REFERENCES

[1] J. Thoma, D. P. Paudel, A. Chhatkuli, and L. Van Gool, "Geometrically mappable image features," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2062–2069, 2020.

[2] X. Xin, J. Jiang, and Y. Zou, "A review of visual-based localization," in *Proceedings of the 2019 International Conference on Robotics, Intelligent Control and Artificial Intelligence*, 2019, pp. 94–105.

[3] A. Irschara, C. Zach, J.-M. Frahm, and H. Bischof, "From structure-from-motion point clouds to fast location recognition," in *CVPR*, 2009, pp. 2599–2606.

[4] T. Sattler, A. Torii, J. Sivic, M. Pollefeys, H. Taira, M. Okutomi, and T. Pajdla, "Are large-scale 3d models really necessary for accurate visual localization?" in *CVPR*, 2017, pp. 1637–1646.

[5] E. Mouragnon, M. Lhuillier, M. Dhome, F. Dekeyser, and P. Sayd, "Real time localization and 3d reconstruction," in *CVPR*, 2006, pp. 363–370.

[6] G. Bresson, Z. Alsayed, L. Yu, and S. Glaser, "Simultaneous localization and mapping: A survey of current trends in autonomous driving," *IEEE Transactions on Intelligent Vehicles*, pp. 194–220, 2017.

[7] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "Netvlad: Cnn architecture for weakly supervised place recognition," in *CVPR*, 2016, pp. 5297–5307.

[8] T. Sattler, W. Maddern, C. Toft, A. Torii, L. Hammarstrand, E. Stenborg, D. Safari, M. Okutomi, M. Pollefeys, J. Sivic *et al.*, "Benchmarking 6dof outdoor visual localization in changing conditions," in *CVPR*, 2018, pp. 8601–8610.

[9] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *CVPR*, 2015, pp. 815–823.

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS*, 2012.

[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778.

[12] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *CVPR*, 2015, pp. 1–9.

[13] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 year, 1000 km: The oxford robotcar dataset," *The International Journal of Robotics Research*, vol. 36, no. 1, pp. 3–15, 11 2016.

[14] A. Torii, J. Sivic, T. Pajdla, and M. Okutomi, "Visual place recognition with repetitive structures," in *CVPR*, vol. 37, 06 2013, pp. 883–890.

[15] A. Pronobis and B. Caputo, "Cold: The cosy localization database," *The International Journal of Robotics Research*, vol. 28, no. 5, pp. 588–594, 05 2009.

[16] A. Pronobis, B. Caputo, P. Jensfelt, and H. I. Christensen, "A discriminative approach to robust visual place recognition," in *IROS*. IEEE, 2006, pp. 3829–3836.

[17] C. Siagian and L. Itti, "Rapid biologically-inspired scene classification using features shared with visual attention," *IEEE transactions on pattern analysis and machine intelligence*, vol. 29, pp. 300–12, 03 2007.

[18] R. Sahdev and J. K. Tsotsos, "Indoor place recognition system for localization of mobile robots," in *2016 13th Conference on Computer and Robot Vision (CRV)*. IEEE, 2016, pp. 53–60.

[19] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *arXiv e-prints*, p. arXiv:1409.1556, Sep. 2014.

[20] R. Arandjelović and A. Zisserman, "Dislocation: Scalable descriptor distinctiveness for location recognition," in *ACCV*, 2014, pp. 188–204.

[21] H. Jin Kim, E. Dunn, and J.-M. Frahm, "Learned contextual feature reweighting for image geo-localization," in *CVPR*, 2017, pp. 2136–2145.

[22] A. Kendall, M. Grimes, and R. Cipolla, "Posenet: A convolutional network for real-time 6-dof camera relocalization," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2938–2946.

[23] S. Brahmbhatt, J. Gu, K. Kim, J. Hays, and J. Kautz, "Geometry-aware learning of maps for camera localization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2616–2625.

[24] M. Dusmanu, I. Rocco, T. Pajdla, M. Pollefeys, J. Sivic, A. Torii, and T. Sattler, "D2-net: A trainable cnn for joint description and detection of local features," in *CVPR*, 2019, pp. 8092–8101.

[25] F. Magliani and A. Prati, "An accurate retrieval through r-mac+ descriptors for landmark recognition," in *Proceedings of the 12th International Conference on Distributed Smart Cameras*, 2018, pp. 1–6.

[26] A. Hermans, L. Beyer, and B. Leibe, "In defense of the triplet loss for person re-identification," *arXiv preprint arXiv:1703.07737*, 2017.

[27] Y. Li, N. Snavely, and D. P. Huttenlocher, "Location recognition using prioritized feature matching," in *ECCV*, 2010, pp. 791–804.

[28] T. Sattler, B. Leibe, and L. Kobbelt, "Efficient & effective prioritized matching for large-scale image-based localization," *PAMI*, pp. 1744–1756, 2017.

[29] B. Zeisl, T. Sattler, and M. Pollefeys, "Camera pose voting for large-scale image-based localization," in *ICCV*, 2015, pp. 2704–2712.

[30] S. Choudhary and P. Narayanan, "Visibility probability structure from sfm datasets and applications," in *ECCV*. Springer, 2012, pp. 130–143.

[31] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "Monoslam: Real-time single camera slam," *PAMI*, pp. 1052–1067, 2007.

[32] R. Castle, G. Klein, and D. W. Murray, "Video-rate localization in multiple maps for wearable augmented reality," in *2008 12th IEEE International Symposium on Wearable Computers*. IEEE, 2008, pp. 15–22.

[33] E. Eade and T. Drummond, "Scalable monocular slam," in *CVPR*, 2006, pp. 469–476.

[34] J. Thoma, D. Pani Paudel, A. Chhatkuli, T. Probst, and L. Van Gool, "Mapping, localization and path planning for image-based navigation using visual features and map," in *CVPR*, 2019.

[35] L. Zheng, Y. Yang, and Q. Tian, "Sift meets cnn: A decade survey of instance retrieval," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 5, pp. 1224–1244, 2017.

[36] N. Piasco, D. Sidibé, C. Demonceaux, and V. Gouet-Brunet, "A survey on visual-based localization: On the benefit of heterogeneous data," *Pattern Recognition*, vol. 74, pp. 90–109, 2018.

[37] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *ICCV*, 2003, p. 1470.

[38] D. P. Robertson and R. Cipolla, "An image-based system for urban navigation," in *Bmvc*, vol. 19, no. 51. Citeseer, 2004, p. 165.

[39] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 2. Ieee, 2006, pp. 2161–2168.

[40] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *CVPR*, 2007.

[41] A. Babenko, A. Slesarev, A. Chigorin, and V. Lempitsky, "Neural codes for image retrieval," in *ECCV*. Springer, 2014, pp. 584–599.

[42] I. Ha, H. Kim, S. Park, and H. Kim, "Image retrieval using bim and features from pretrained vgg network for indoor localization," *Building and Environment*, vol. 140, 05 2018.

[43] G. Tolias, R. Sicre, and H. Jégou, "Particular object retrieval with integral max-pooling of CNN activations," *arXiv e-prints*, p. arXiv:1511.05879, Nov. 2015.

[44] P. Dollár, Z. Tu, P. Perona, and S. Belongie, "Integral channel features," *British Machine Vision Conference (BMVC)*, 01 2009.

[45] A. Gordo, J. Almazán, J. Revaud, and D. Larlus, "End-to-end learning of deep visual representations for image retrieval," *Int. J. Comput. Vision*, vol. 124, no. 2, pp. 237–254, Sep. 2017.

[46] M. Angelina Uy and G. Hee Lee, "Pointnetvlad: Deep point cloud based retrieval for large-scale place recognition," in *CVPR*, 2018, pp. 4470–4479.

[47] W. Chen, X. Chen, J. Zhang, and K. Huang, "Beyond triplet loss: a deep quadruplet network for person re-identification," in *CVPR*, 2017, pp. 403–412.

[48] T. Shen, Z. Luo, L. Zhou, R. Zhang, S. Zhu, T. Fang, and L. Quan, "Matchable image retrieval by learning from surface reconstruction," in *Asian Conference on Computer Vision*. Springer, 2018, pp. 415–431.

[49] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *CVPR*, 2019, pp. 4401–4410.

[50] T. Cieslewski, S. Choudhary, and D. Scaramuzza, "Data-efficient decentralized visual slam," in *ICRA*. IEEE, 2018, pp. 2466–2473.

[51] W. Maddern, G. Pascoe, M. Gadd, D. Barnes, B. Yeomans, and P. Newman, "Real-time kinematic ground truth for the oxford robotcar dataset," *arXiv preprint arXiv: 2002.10152*, 2020.