2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN), 30 Sept. - 3 Oct. 2019, Pisa, Italy

InFo: Indoor localization using Fusion of Visual Information from Static and Dynamic Cameras

Chelhwon Kim*, Chidansh Bhatt*, Mitesh Patel, Don Kimber, Yulius Tjahjadi
FX Palo Alto Laboratory Inc.
Palo Alto, CA - 94304, U.S.A
{kim,bhatt,mitesh,kimber,yulius}@fxpal.com

***Abstract*—Localization in an indoor or Global Positioning System (GPS)-denied environment is paramount. It drives various applications that require locating humans or robots in an unknown environment. Various localization systems using different ubiquitous sensors such as camera, radio frequency, inertial measurement unit have been developed. Most of these systems cannot accommodate for scenarios which have substantial changes in the environment such as a large number of people (unpredictable) and sudden change in the environment floor plan (unstructured). In this paper, we propose a system, InFo that can leverage real-time visual information captured by surveillance cameras and augment that with images captured by the smart device user to deliver accurate discretized location information. Through our experiments, we demonstrate that our deep learning based InFo system provides an improvement of 10% as compared to a system that does not utilize this real-time information.**

***Index Terms*—indoor localization, deep learning, image matching / retrieval, conditional metric learning, triplet loss, NetVLAD**

I. Introduction

Location-aware applications for smart devices have gained a lot of traction in the last decade. This is primarily due to the rich level of sensing provided by smart devices. Global Positioning System (GPS) provides accurate outdoor localization which is currently being used to drive various outdoor applications on smart devices. However, GPS signals suffer from Non-Line of Sight (NLOS) issues [15], and hence GPS-based localization is not viable for indoor environments. Indoor localization systems using various sensors such as wireless local area networks (WLAN), Ultra Wide Band (UWB), RFID tags, Bluetooth Low Energy (BLE), Inertial Measurement Unit (IMU), and RGB images etc. have been extensively tested by various research groups [9], and none of these technologies provides a complete solution to all indoor positioning needs as each sensor has its own set of challenges. For example, Radio Frequency (RF) sensor based localization systems require deployment of new infrastructure in an indoor environment which can be labor intensive and costly to both deploy and maintenance in large public spaces. Further, RF sensor behavior varies with changes in environment such as a large number of people, partitions, etc., which make localization more challenging [29].

Lately, due to the advancements in the field of Computer Vision (CV), image based localization systems have gained traction [36] [24]. These image-based systems have their own challenges for recognizing the same place between the query and the reference database images collected in different time and different viewing conditions, which leads to a large variation of image appearance between them. This is mainly due to unexpected/unstructured changes in the environment over time such as changes in layout of furniture, occlusions introduced by different objects in different time (e.g. crowded shopping mall in day time), etc. This large variation between the query and the database images often introduces a significant amount of performance looses on the localization, especially when a system is trained with images collected under one or a few conditions. Figure 1 shows examples of failure cases due to this varying conditions between the query and the reference images reported in Taira *et. al.* [36] which gives state-of-the-art performance for indoor localization problem, as well as sample images in our dataset under different viewing conditions. It is impractical to expect the reference database can represent the scene under all the viewing conditions, and hence there is a need to develop a system that incorporates real-time change in environment.

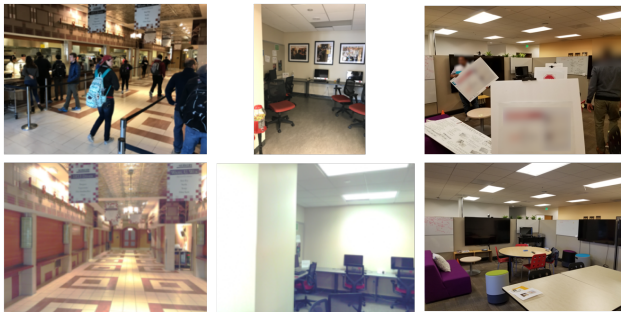

Fig. 1: Failure cases reported by Taira et al. in their indoor localization system [36]. The system fails to localize images shown in first two columns due to many moving objects, e.g. people or chairs, and highly dynamic scenes, e.g. opened/closed shutters or pictures on the wall removed. Last column shows sample images from the dataset we collected in our office space with moving objects and people to test our system for robust localization.

In this paper, we propose a vision based indoor localization system that is specifically designed for areas with surveillance camera infrastructure which is ubiquitous and is readily available in most commercial building spaces and large

978-1-7281-1788-1/19/$31.00 © 2019 IEEE

*these authors contributed equally to this work

public gathering venues. The system utilizes real-time images captured by the surveillance cameras which is fused with images captured by users' smartphone camera to provide their location information. It should be noted our proposed system provides coarse location information which is the areas that are covered by Field of View (FOV) of surveillance cameras.

II. Related Work

Over the past few years, an increasing number of approaches have been developed to address the challenging and important problem of indoor localization using different types of sensors such as RGB images [17], [24], fusion of RGB and depth images, RF sensors such as Blutooth Low Energy (BLE) [25], Wi-Fi [1] etc. Researchers have explored techniques which involves fusion of two or more sensors [15], [18], [26]. In this Section, we provide details on some of the state-of-art localization system by dividing them largely into two categories: IoT sensors based localization and computer vision based localization.

A. IoT sensor-based approaches

Numerious localization and tracking systems have been developed using different IoT sensors such as BLE [15], Wi-Fi [5], [11] and Inertial Measurement Unit (IMU) [22]. Localization using fingerprinting technique for BLE and/or Wi-Fi sensor has been an approach for the indoor localization. In this technique a fingerprint map of the environment for the RSSI signal is generated offline, which is then used for real-time signals [11], [38]. In [21], Ma et al. proposed a RSSI ranking based fingerprinting method that uses Kendall Tau correlation coefficient to correlate the position with the signal strength ranking of multiple BLE devices deployed in a given environment. Similarly, Faragher and Harle [11] used a fingerprinting technique to analyze the performance of BLE based localization using K-NNs and proximity-based techniques. Carrillo et al. [8] developed a localization system that utilizes magnetic field map of an environment for localization. In model based localization systems, the propagation of raw RSSI through a space is modeled using various techniques such as Friis free space model [12], Gaussian processes [15], or Wasserstein distance model [13]. These sensor models are further utilized within different probabilistic frameworks to estimate the location of the user [25]. Most of the above mentioned systems generate a map of the environment beforehand and does not account real-time changes in the environment.

B. Computer Vision-based approaches

Localization using various computer vision based approaches such as feature based [4], [10], [20] and more recently deep learning based approaches [17], [24], [36] have been proposed by various researchers. For example, Bennewitz et al. proposed a localization system which utilizes local scale-invariant features (SIFT [20]) in the probabilistic filtering framework to localize a robot with a single perspective camera [4]. 3D structure-based approaches based on correspondences between 2D SIFT keypoints in a test image and 3D points in the scene also have shown promising results by [6], [7], [23], [37]. Lately, with the evolution of deep learning, researchers have also developed different location systems which learn to directly regress the absolute camera pose in the scene from a single or multiple images. For example, Kendall et al. [17] proposed an end-to-end approach based on convolutional deep features which provides 6DOF pose estimation from an input image. Similarly, Patel et al. [24] developed a recurrent neural network based system that learns the spatial and temporal features to estimate the precise camera pose from a sequence of images. These deep end-to-end systems are computationally efficient with a single forward pass pipeline, however, they are still less accurate than structured-based approaches [31], [32]. To the best of our knowledge, the localization system proposed by Yan et al. [39] is closely related to our work as they utilize an image captured by a single surveillance camera to perform localization. Their primary focus is to develop a vision-based pedestrian tracking system that uses map information along with the depth and RGB image captured by surveillance system.

Most of the localization systems proposed above learn from the images captured in the past and hence are not able to accommodate for environmental changes that are dynamic in nature. Our proposed indoor localization system ***InFo**: **In**door Localization using Image **Fusi**on* is designed to accommodate for these dynamic changes by fusing two sources of information captured by dynamic and static cameras in a deep metric learning framework [33]. In this paper, we propose two different ways of fusion that can be efficiently incorporated into the metric learning framework: 1) Explicit fusion: we explicitly fuse the visual information of the static surveillance camera images by adding them in the training input batch fed to our neural network. This enables us to fuse the different degree of visual dynamics in the static and dynamic images in the learning of the metric space. 2) Implicit fusion: we adjust the behaviour of the metric space by modulating our neural network with high-level contextual information extracted from the static images using a feature-wise transformation technique [27]. This *conditioned* metric space enables us to process the dynamic camera image in the context of current environmental changes monitored in real-time by the surveillance cameras.

Contributions of our work can be summarized as follows.

- We propose a novel system named ***InFo*** which is able to accommodate for real-time changes in the environment which are dynamic, unstructured and unpredictable in nature. Our system, fuses the real-time information captured by the surveillance system with image captured by smart device to provide zone level localization.
- Our system does not require the labor intensive process of labeling data for training the system as the image captured by each surveillance system is known beforehand which is used as labels to represent each zone.
- We compare the performance of two different variants of ***InFo*** system which has different ways to fuse image features with traditional feature-based systems such as SparseSIFT+VLAD and DenseSIFT+VLAD [36], [37].

III. INFO SYSTEM

Our proposed system follows the standard image retrieval pipeline where a query image is used to visually search a geo-tagged image database using an image matching algorithm, and the locations of matched database images are used to approximate the location of the query image. To achieve high accuracy and efficiency in the search performance the images are encoded into a compact visual feature space which has high discriminating features. Converse to traditional image retrieval systems which uses hand-engineering local features descriptors such as SIFT [19], Surf [3], and aggregation techniques, such as bag-of-visual-words [28], VLAD [16], we use a deep learning architecture that learns the compact embedding space using metric learning technique [33], where distances in the learned metric space directly correspond to a measure of visual similarity of images. This metric learning makes the visual matching task much simpler as it calculates the L2 Euclidean distance metric in the learned embedding space [33].

Further, to project the image into a good embedding space we utilize triplet loss function as described in [2], [33]. The loss function reduces the euclidean distance between all images taken in the same place, irrespective of the viewing condition due to camera poses and environmental changes, and at the same time increases the distance between pair of images taken at different places. Further details of triplet loss can be found in [33]. For the feature extraction, we use a deep CNN architecture VGG-16 [34] to extract local descriptors from an image, followed by a global pooling layer called NetVLAD [2] which aggregate all the local descriptors into a single vector of d-dimensional space, $\phi(\mathbf{x}) \in \mathbb{R}^{\mathrm{d}}$, where ϕ is the learned embedding and $\mathbf{x}$ is the input image.

An illustration of our deep learning pipeline is shown in Fig. 2, where we sample, from a batch of training images, valid triplets of an anchor ($\mathbf{x}_a$), a positive image ($\mathbf{x}_+$) in the same zone as the anchor, and a negative image ($\mathbf{x}_-$) of a different zone, each of which is converted into a compact feature by our embedding network ϕ. Then, the embedding network ϕ is optimized with the following triplet loss as in [33]:

$$L = \sum_i \max\left(d(\phi(\mathbf{x}_a^i), \phi(\mathbf{x}_+^i)) - d(\phi(\mathbf{x}_a^i), \phi(\mathbf{x}_-^i) + \alpha, 0\right) \tag{1}$$

where $\phi(\mathbf{x}^i_{\{a,+,-\}})$ is the embedding vector of i-th triplet and the subscripts a, $+$, $-$ denote the anchor, positive, and negative samples respectively. $d(\cdot,\cdot)$ is the squared L2-distance and α is a margin that is enforced between positive and negative pair. Further details of the architecture and training parameters of our network are described in Appendix IX.

Built on this deep metric learning framework, we propose two different methods (explicit fusion and implicit fusion) to fuse information captured from the surveillance system which is static in nature and smartphone camera images which are dynamic in nature.

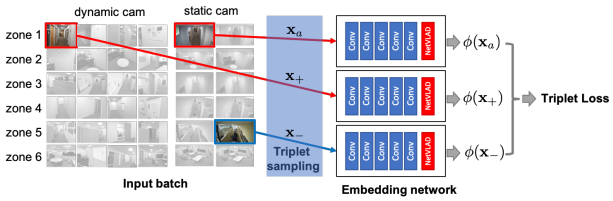

Fig. 2: Overview of embedding learning with our explicit fusion method. We build the valid triplets by randomly sampling images from both dynamic (left four columns) and static (right two columns) pools.

A. Explicit Fusion Method

In this method, we explicitly utilize the static camera images within the metric learning framework by directly including them in the input batch together with the dynamic camera images, and building the valid triplets by randomly sampling images from the both pools of static and dynamic camera images (refer Fig. 2). By learning embedding space with these triplets, both the static and dynamic images depicting the same place will be embedded close to each other in the learned space.

Once the embedding space is produced, we perform the image-to-image matching between the query image (smartphone camera) collected at time t with the real-time surveillance camera images collected at the same time t. It is important to note that we do not perform the K-nearest neighbor searching in the pre-collected database (training images), and aggregating the zone labels of the nearest ones to determine the zone label of the query image by a voting scheme. The rational behind this is, in the learned embedding space, the real-time surveillance image can still serve as a stable representative for depicting each zone under the environmental changes, and this will be discussed further in Section V-B. This search approach also enables us to reduce the query retrieval time with a few distances metric computations between the query image and the current surveillance camera images. (12k images in our database v/s 6 surveillance images). Our zone-level prediction label is computed as follows:

$$\arg\min_i ||\phi(\mathbf{x}_d(t)) - \phi(\mathbf{x}_{s_i}(t))||^2 \tag{2}$$

where $\mathbf{x}_d(t)$ is the dynamic query image captured by smartphone camera at time t, and $\mathbf{x}_{s_i}(t)$ is the image captured by the static surveillance camera in zone-i at the same time t.

B. Implicit Fusion Method

In the implicit fusion method, we learn an embedding space that can be adapted based on the visual appearance of the indoor scene. To achieve such adaptation, we are fusing the real-time information captured by the static surveillance cameras and dynamic cameras. The distance metric of the dynamic images in that adapted embedding space is invariant to the unpredictable/unstructured environment changes. To do this, we propose to use a conditioned embedding network modified by feature-wise transformation layers [14], [27] and

a separate deep network to capture the high-level context information of the visual appearance.

The embedding learning pipeline for our proposed implicit method is shown in Fig. 3a and the proposed embedding network architecture ϕ is shown in Fig. 3b. The main network (VGG16+NetVLAD, Fig. 3b top) takes as input the dynamic camera image, $\mathbf{x}_d$ and embeds it into a compact Euclidean space, which is the same as the explicit model. To extract high-level context information from the static camera images, we use a separate InceptionV3 [35] deep neural network architecture that takes as input all images captured from surveillance cameras, $\mathbf{x}_{s_1}, ..., \mathbf{x}_{s_n}$ and outputs features that are aggregated by a simple concatenation (Fig. 3b bottom). This information is then used to modulate the main embedding network using the feature-wise transformation [14], [27], where we transform features of intermediate layers of the main embedding network with additional Feature Transform (FT) layers in the middle of the embedding network. This feature-wise transformation can provide an efficient way of integrating the conditioning information into the network.

Inspired by previous work [14], [27], [38], [40], we design the feature transformation with a simple feature-wise affine transformation:

$$FT(\mathbf{h}) = \gamma * \mathbf{h} + \beta \tag{3}$$

where $\mathbf{h}$ is the feature activation of the intermediate layer of the embedding network, and γ and β are scaling and shifting parameters respectively, which are produced by the condition network.

Fig. 3a illustrates the triplet sampling in the training phase, where we randomly sample a valid triplet of an anchor set ($\mathbf{s}_a$), a positive set ($\mathbf{s}_+$), and a negative set ($\mathbf{s}_-$), where the set $\mathbf{s}$ consists of one dynamic and all static camera images captured at the same time belonging to the same zone. All the weights of the embedding network with the feature transformation layers (as shown on Fig. 3b) are trained in an end-to-end manner with the triplet loss to directly optimize the embedding space for the visual search task.

Once this conditioned embedding space is produced after the training, similar to the explicit method, we do not perform the K-NN search, instead, we store mean embeddings (i.e. cluster centroid) for all zones in a database offline, and at test time, a query set $\mathbf{s} = \{\mathbf{x}_d, \mathbf{x}_{s_1}, ..., \mathbf{x}_{s_n}\}$ is converted to the embedding vector through our trained embedding network. The nearest database mean embedding vector to the query is used to determine the zone label. Our predicted zone label is computed as follows:

$$\arg\min_i ||\phi(\mathbf{s}(t)) - \mathbf{c}_i||^2 \tag{4}$$

where $\mathbf{c}_i$ is the database mean embedding vector of zone-i, $\mathbf{s}(t)$ is the set of dynamic and static camera images captured at time t, $\{\mathbf{x}_d(t), \mathbf{x}_{s_1}(t), ..., \mathbf{x}_{s_n}(t)\}$.

IV. Experimental Setup

We perform quantitative and qualitative assessment of our ***InFo*** system on a dataset collected in an office space with static surveillance cameras and dynamic smartphone cameras. We also compare our method to various state-of-the-art methods.

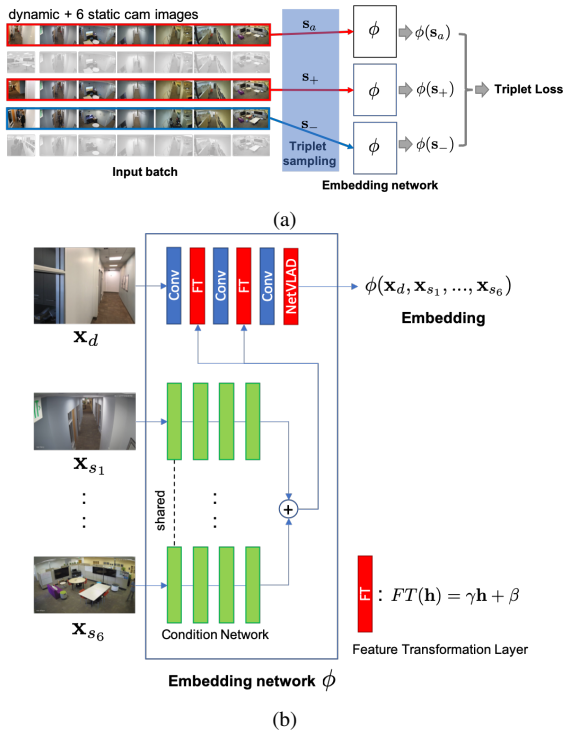

Fig. 3: Proposed conditional embedding learning. The overall embedding learning pipeline is shown in (a) whereas proposed implicit fusion network architecture with the embedding network and the condition network is shown in (b).

A. Dataset collection

As shown in the Fig. 4, we installed 6 surveillance cameras in the office environment to capture different parts of the office building. Areas covered by the surveillance camera's FOV are annotated with the corresponding zone numbers. For each zone, a set of query images were captured with different types of smartphones at multiple random locations by two different users. At each location, each user captures 4∼5 images with different viewing angles in both portrait and landscape mode. For each query image, we extracted key frames from the surveillance videos recorded during the data collection using the query image's timestamp so that each query image is paired with 6 surveillance images in the dataset. We performed the dataset collection multiple times periodically over a three months period.

To evaluate the robustness of our proposed system, our dataset contains various complex situations, e.g., unpredictable and unstructured dynamic environmental changes with multi-person and multi-object movements over time across different zones. In particular, we collected the dataset under three different viewing conditions.

- **normal**: Consists of images of a scene in the normal condition. i.e. mostly static areas without any dynamic objects moving/changing in the area.

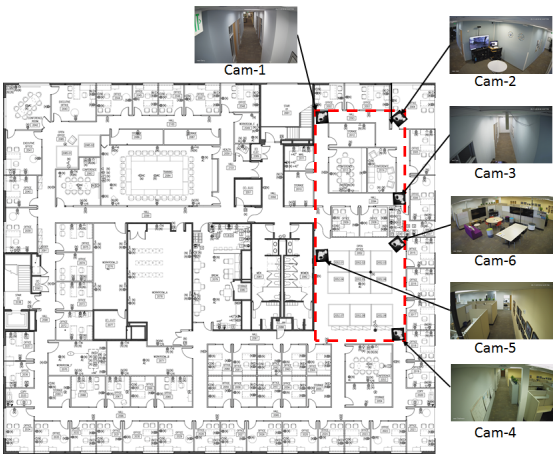

Fig. 4: Static surveillance camera placements in the office environment for data collection

Fig. 5: Examples of images of the same scene captured by dynamic (top) and static (bottom) cameras under different conditions.

- **people**: Consists of images of a scene captured with actors (e.g., people moving around the area).
- **complex**: Consists of images of the most complex scene captured with actors moving around and holding poster signs (e.g., multi-person and multi-object movements).

Fig. 5 shows examples of our dataset captured by dynamic and static cameras under the above conditions.

In the normal scenario, we collected total 705 query images with different smartphones. To increase our dataset size, we also used a 360 degree video camera to synthesize total 12775 perspective images of the scene (See Appendix for more details). With the smartphone camera images, total 13480 query images were collected. In the people and complex scenarios, we collected total 334 smartphone camera images for each.

V. Results

We compared the performance of our ***InFo*** system to the following baseline methods (Further details about implementation of SIFT based baseline methods can be found in Appendix IX):

- **Im2Im**: In this method, we perform the standard image-to-image matching between the query smartphone image and all 6 surveillance images captured at the same time as the query. The image matching is based on the number of detected SIFT [20] key-point matches, and only geometrically verified matches (inliers) using homographies with RANSAC are kept. The zone label of the surveillance image having the highest number of inlier matches with the query image is used as the prediction.
- **SparseSIFT+VLAD**: In this method, we construct the bag-of-visual-words [28] of SIFT features extracted from the smartphone images in the training set. SIFT features are computed at interest points in the image using the difference of Gaussian (DoG) feature detector which is usually sparse (hence we call this as SparseSIFT). At test time, SIFT features in the query image is aggregated into a single descriptor using VLAD [16] based on the learned visual words. The prediction of zone label is determined by finding the closest training image based on the Euclidean distances between VLAD descriptors.
- **DenseSIFT+VLAD**: Same as SparseSIFT+VLAD except that SIFT features are computed at densely and uniformly sampled key-points in the image.
- **VGG16+NetVLAD**: This baseline model is same as our explicit model described in Section III except that the valid triplets are sampled from only dynamic camera images i.e. there is no fusion included.

TABLE I: Zone-level position prediction accuracy for different unexpected viewing conditions. The reported numbers for deep learning based methods (last three rows) are the mean and standard deviation in parenthesis of the prediction accuracy for 10 different trained models with the same parameters.

Method	People	Complex	Average
Im2Im	41.9	28.8	35.4
SparseSIFT+VLAD	51.2	32.0	41.6
DenseSIFT+VLAD	88.3	71.2	79.8
VGG16+NetVLAD	94.2 (2.5)	78.1 (3.5)	86.1
Explicit Fusion	**96.9** (1.6)	82.9 (2.3)	89.9
Implicit Fusion	94.6 (1.8)	**85.7** (3.3)	**90.2**

To assess the impact of the unexpected viewing conditions on the baseline models and our proposed fusion approaches, we divided our dataset into training and test sets, each of which presents different viewing conditions. Namely, the models are trained[1] on the set of images collected during 'normal' condition and tested on set of images with 'people' and 'complex' conditions.

A. Zone Prediction

We first compare our approaches to the baseline models in terms of zone-level position prediction accuracy. Table I shows the accuracy values for the baseline methods and our proposed fusion methods under different conditions (i.e. dataset 'people' and 'complex'). It is evident from the results in Table I that there is a drastic improvement using the state-of-the-art image retrieval method built on densely sampled

[1]Im2Im method doesn't require the training phase. For SparseSIFT+VLAD and DenseSIFT+VLAD methods, we learned the bag-of-visual-words from the set of 'normal' images and used it for computing VLAD.

local invariant features followed by compact VLAD encoding, i.e. DenseSIFT+VLAD with average accuracy of 79.8%, compared to the result obtained with the traditional feature descriptor-based image-to-image matching method, i.e. Im2Im with average accuracy of 35.4%. As in [37], we also found that, as compared to SparseSIFT+VLAD (average accuracy of 41.6%), DenseSIFT+VLAD representing the image using the densely sampled SIFT features shows better accuracy, as it does not rely on repeatable detection of local invariant features (e.g. DoG keypoints), and is more robust to large changes in visual appearance due to the camera poses.

The last three rows of Table I are the methods built on the state-of-the-art deep learning based CNN features, followed by VLAD encoding (NetVLAD) without/with our proposed explicit and implicit fusion. All methods are trained with the triplet loss for learning the deep embedding space as described in Section III. The reported numbers in the table for these methods are the mean and standard deviation of the prediction accuracy for 10 different trained models with the same training parameters. See Appendix for more details about training.

As compared to DenseSIFT+VLAD which is built on the hand engineered features, the deep learning based baseline model trained with the triplet loss, VGG16+NetVLAD shows around 6% improvement (from 79.8% to 86.1%) in average, and 7% improvement for 'complex' condition (from 71.2% to 78.1%). With using our proposed fusion methods, we achieved around **10**% improvement in average for both explicit (from 79.8% to 89.9%) and implicit (from 79.8% to 90.2%) methods, and 12% and 15% improvements in the 'complex' scenario for explicit (from 71.2% to 82.9%) and implicit (from 71.2% to 85.7%) method respectively. In the 'people' condition, the accuracy of all deep learning based methods are saturated over 94%.

B. Clustering evaluation

To further test the robustness of our system to the unexpected environmental changes, we used the criterion of measuring how well each query image captured under the **complex** condition lies within its corresponding cluster (cohesion) compared to other clusters (separation) in the learned embedding space under the **normal** condition. To measure this quantitatively, we use Silhouette value which is presented by [30] and has been used as a means for clustering evaluation.

More precisely, to compute the silhouette value for a query embedding e_q belonging to Zone A, we compare the average distance $a(e_q)$ between the query and all embeddings belonging to the same zone (or a cluster C_A) with the average distance $b(e_q)$ between the query and all embeddings in the neighbor cluster C_B:

$$
\begin{aligned}
a(e_q) &= \frac{1}{|C_A|} \sum_{i \in C_A} d(e_q, e_i) \\
b(e_q) &= \min_{C_B \neq C_A} \frac{1}{|C_B|} \sum_{i \in C_B} d(e_q, e_i) \\
s(e_q) &= \frac{b(e_q) - a(e_q)}{\max\{a(e_q), b(e_q)\}}
\end{aligned}
\tag{5}
$$

TABLE II: The average silhouette value of the 'complex' dynamic query embeddings for the deep learning based methods.

Method	Silhouette-Value
VGG+NetVLAD	0.32
Explicit Fusion	0.43
Implicit Fusion	0.71

where $|C|$ is the number of embeddings in the cluster, and $d(\cdot, \cdot)$ is L2 Euclidean distance in the learned embedding space. The silhouette value $s(e_q)$ ranges between -1 and 1. If $s(e_q)$ is large, the query image is well classified, otherwise it can be misclassified. By taking the average of the silhouette values of all the 'complex' query images, we can provide an assessment of overall quality of the learned embedding space in terms of how well the 'complex' query images can be classified, and this indicates the robustness of the system to the complex condition when the system is trained under the normal condition.

Table II shows the average silhouette value of the 'complex' dynamic query images within the embedding spaces learned by the baseline models with/without our fusion methods. The implicit fusion method achieves the highest silhouette value.The baseline model without fusion shows the worst. This result is consistent with our visualization[2] of the embedding spaces as shown in Fig. 6. In each plot, dots represent the 'normal' dynamic embeddings that are color-coded according to their zone labels, and the 'x' markers represent the 'complex' dynamic query embeddings. We observed that, by training the models with the triplet loss, the 'normal' embeddings (dots) for each zone is internally dense and separated well from the rest of embeddings in the learned embedding space for all models. However, 'complex' query embeddings (x) are spread away from the 'normal' clusters in both the baseline and explicit fusion-embedding space, and their average silhouette values are 0.32 and 0.43, respectively. See Fig. 6 (a) and (c). In the embedding space of the implicit fusion (Fig. 6 (b)), the 'complex' embeddings (x) are relatively close to the clusters, and its average silhouette value is 0.71 which is higher than other embedding spaces.

We also observed that clusters of 'normal' **static** surveillance images in the embedding space of the explicit fusion (Fig. 6 (d)) are more tightly packed (color-coded dots) relative to the 'normal' **dynamic** images (gray dots) due to their different degree of diversity of visual appearance in the images: i.e. the environmental changes in the scene such as people, moving objects etc. are appeared relatively small in the surveillance image due to its far distance to the scene structure, large FOV, and fixed camera angle, whereas in the dynamic images the moving objects can introduce larger occlusion when the images are captured by the user's handheld smartphone with much shorter distance, smaller FOV, and dynamic camera viewing directions. We also plot the 'complex' **static** surveillance images with color-coded cross markers '+' in Fig. 6 (d), which are much less diverse relative

[2]The visualization is done by projecting 4096-dim NetVLAD descriptors into 2-D space by using PCA analysis. Note that the silhouette value is computed in the original space.

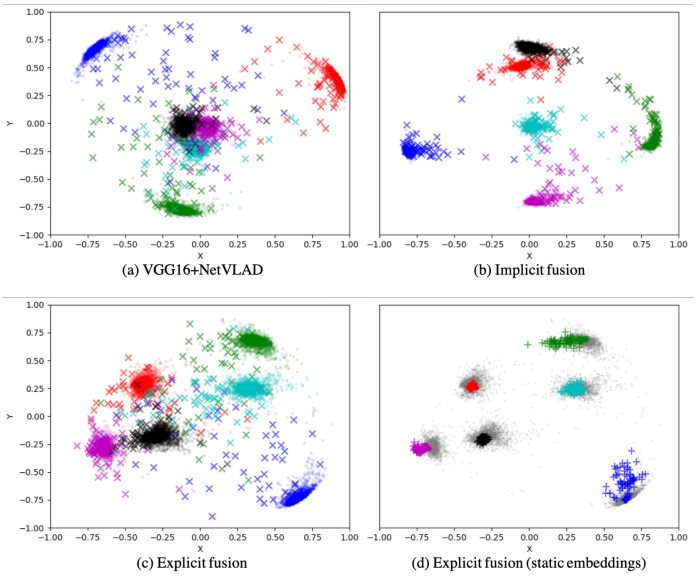

Fig. 6: Visualization of the learned embedding spaces of Method (a) VGG16+NetVLAD (b) implicit fusion (c) explicit fusion (d) explicit fusion with static embeddings. Dots represent the 'normal' dynamic embeddings, which are color-coded according to their zone labels. Marker 'x' represents the 'complex' query dynamic embeddings. In (d), the gray dots represent the 'normal' dynamic embeddings, and the color-coded dots and '+' markers represent the 'normal' and the 'complex' static embeddings respectively.

to the 'complex' **dynamic** query images ('x' markers in Fig. 6 (c)). This explains why the explicit model perform better than the baseline model without fusion in the zone-level prediction as the explicit method predicts the zone label by searching the most closest current **static** surveillance images as in Eq. 2, each of which can serve as a stable representatives for depicting each zone with much more less diversity of visual appearance.

VI. Scope and Limitations

As demonstrated the proposed ***InFo System*** is novel and able to outperform systems that do not utilize real-time information from the surveillance system. In particular, the proposed system can provide a robust implementation for long-term indoor positioning system at zone-level with richer context information for intelligent mobile applications. One of the limitations is in terms of required surveillance infrastructure is the assumption that it covers all the area of interests in the indoor environment. If there are no training data due to lack of static surveillance cameras or dynamic query images for certain parts of indoor environments then the accuracy of the localization system will suffer. *InFo system* provides the localization at zone-level and not as the six degrees of freedom(6-DoF), which is still important for many real-world applications where zone-level context information is sufficient and rather more cost-effective.Also, the proposed model is flexible and can be extendable with the incorporation of 3D-model details to provide localization with 6-DoF details.

VII. Future Work

In our future work, we will leverage existing zone-level localization to reduce the search space and provide efficient computing for localization with camera pose details at 6-DoF. Also, we will incorporate self-supervised semantic activity detection and monitoring system within each zone and across all the zones to enhance the quality of context beyond just localization information and the user or device-centric activity patterns. Thus, our proposed *InFo system* is the first building block in our larger vision of developing low-cost accurate indoor localization system with detail pose estimation enabling many other applications like indoor navigation, robot re-localization, calibration-free large-scale AR/VR in future. We also plan to collect much larger and challenging dataset from the extremely dynamic environments like busy train stations, crowded shopping malls, exhibition halls and make it publicly accessible for benchmarking purpose to extend the research in indoor localization using real-time context fusion of visual information from static and dynamic cameras.

VIII. Conclusion

In this paper, we proposed a zone level localization system that leverage on real-time images captured by the surveillance system to provide localization to a smart device user. Our system is capable to incorporate real-time environmental context to overcome large variations in image matching due to unpredictable and unstructured changes within highly dynamic environments. Our contribution to collect a novel dataset for such dynamic and static camera images for indoor localization research will be helpful for the community to explore more scalable, low-cost and intuitive context-aware applications. Proposed fusion methods that can be incorporated into the deep metric learning framework will be further useful to similar problems and extended for the better results with adding novelty to improve the accuracy further for many challenging scenarios.

IX. Appendix

Appendix section is available online at https://sharedocs.paldeploy.com/s/CsiwzzKLLQtooTL

References

[1] F. Adib, Z. Kabelac, and D. Katabi. Multi-person localization via rf body reflections. In *Proceedings of the 12th USENIX Conference on Networked Systems Design and Implementation*, NSDI'15, pages 279–292. USENIX Association, 2015.

[2] R. Arandjelović, P. Gronat, A. Torii, T. Pajdla, and J. Sivic. NetVLAD: CNN architecture for weakly supervised place recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[3] H. Bay, T. Tuytelaars, and L. V. Gool. Surf: Speeded up robust features. In *In ECCV*, pages 404–417, 2006.

[4] M. Bennewitz, C. Stachniss, W. Burgard, and S. Behnke. Metric localization with scale-invariant visual features using a single perspective camera. In H. I. Christensen, editor, *European Robotics Symposium 2006*, pages 195–209, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[5] J. T. Biehl, M. Cooper, G. Filby, and S. Kratz. LoCo: A ready-to-deploy framework for efficient room localization using Wi-Fi. In *Proc. of the ACM Int. Joint Conf. on Pervasive and Ubiquitous Computing*, pages 183 – 187, 2014.

[6] E. Brachmann, A. Krull, S. Nowozin, J. Shotton, F. Michel, S. Gumhold, and C. Rother. Dsac-differentiable ransac for camera localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6684–6692, 2017.

[7] E. Brachmann and C. Rother. Learning less is more-6d camera localization via 3d surface regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4654–4662, 2018.

[8] D. Carrillo, V. Moreno, B. Ú. Miñaro, and A. F. Skarmeta. Magicfinger: 3d magnetic fingerprints for indoor location. *Sensors*, 15(7):17168–17194, 2015.

[9] P. Davidson and R. Pich. A survey of selected indoor positioning methods for smartphones. *IEEE Communications Surveys Tutorials*, 19(2):1347–1370, Secondquarter 2017.

[10] D. DeTone, T. Malisiewicz, and A. Rabinovich. Superpoint: Self-supervised interest point detection and description. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2018.

[11] R. Faragher and R. Harle. Location Fingerprinting With Bluetooth Low Energy Beacons. *IEEE Journal on Selected Areas in Communications*, 33(11):2418–2428, 2015.

[12] H. Friis. A Note on a Simple Transmission Formula. *Proceedings of the I.R.E. and Waves and Electrons*, 34(5):254–256, 1946.

[13] D. F.S. and C. A. T. Model-based localization and tracking using bluetooth low-energy beacons. *Sensors*, 17(11), 2017.

[14] G. Ghiasi, H. Lee, M. Kudlur, V. Dumoulin, and J. Shlens. Exploring the structure of a real-time, arbitrary neural artistic stylization network. 2017.

[15] M. G. Jadidi, M. Patel, and J. V. Miro. Gaussian processes online observation classification for rssi-based low-cost indoor positioning systems. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6269–6275, 2017.

[16] H. Jegou, F. Perronnin, M. Douze, J. Sanchez, P. Perez, and C. Schmid. Aggregating local image descriptors into compact codes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(9), Sept. 2012.

[17] A. Kendall, M. Grimes, and R. Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *ICCV*, pages 2938–2946. IEEE Computer Society, 2015.

[18] N. Y. Ko and T.-Y. Kuc. Fusing range measurements from ultrasonic beacons and a laser range finder for localization of a mobile robot. In *Sensors*, volume 15, pages 11050 – 11075, 2015.

[19] D. G. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 2, Sept 1999.

[20] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[21] Z. Ma, S. Poslad, J. Bigham, X. Zhang, and L. Men. A ble rssi ranking based indoor positioning system for generic smartphones. In *Wireless Telecommunications Symposium (WTS)*, pages 1–8, 2017.

[22] V. Malyavej, W. Kumkeaw, and M. Aorpimai. Indoor robot localization by rssi/imu sensor fusion. In *2013 10th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology*, pages 1–6, 2013.

[23] D. Massiceti, A. Krull, E. Brachmann, C. Rother, and P. H. Torr. Random forests versus neural networkswhat's best for camera localization? In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5118–5125. IEEE, 2017.

[24] M. Patel, B. Emery, and Y. Chen. Contextualnet: Exploiting contextual information using lstms to improve image-based localization. In *ICRA*, pages 1–7. IEEE, 2018.

[25] M. Patel, A. Girgensohn, and J. Biehl. Fusing Map Information with a Probabilistic Sensor Model for Indoor Localization Using RF Beacons. *International Conference on Indoor Positioning and Indoor Navigation (IPIN'18)*, pages 1–8, 2018.

[26] A. Perera, J. Arukgoda, R. Ranasinghe, and G. Dissanayake. Localization system for carers to track elderly people in visits to a crowded shopping mall. In *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–8, 2017.

[27] E. Perez, F. Strub, H. de Vries, V. Dumoulin, and A. C. Courville. Film: Visual reasoning with a general conditioning layer. In *AAAI*, 2018.

[28] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *CVPR*. IEEE Computer Society, 2007.

[29] K. Qian, C. Wu, Y. Zhang, G. Zhang, Z. Yang, and Y. Liu. Widar2.0: Passive human tracking with a single wi-fi link. In *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '18, pages 350–361, New York, NY, USA, 2018. ACM.

[30] P. J. Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.

[31] T. Sattler, W. Maddern, C. Toft, A. Torii, L. Hammarstrand, E. Stenborg, D. Safari, M. Okutomi, M. Pollefeys, J. Sivic, F. Kahl, and T. Pajdla. Benchmarking 6dof outdoor visual localization in changing conditions. In *CVPR*, pages 8601–8610. IEEE Computer Society, 2018.

[32] T. Sattler, Q. Zhou, M. Pollefeys, and L. Leal-Taixe. Understanding the limitations of cnn-based absolute camera pose regression. *arXiv preprint arXiv:1903.07504*, 2019.

[33] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. *CoRR*, abs/1503.03832, 2015.

[34] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[35] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[36] H. Taira, M. Okutomi, T. Sattler, M. Cimpoi, M. Pollefeys, J. Sivic, T. Pajdla, and A. Torii. Inloc: Indoor visual localization with dense matching and view synthesis. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[37] A. Torii, R. Arandjelovic, J. Sivic, M. Okutomi, and T. Pajdla. 24/7 place recognition by view synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1808–1817, 2015.

[38] X. Wang, K. Yu, C. Dong, and C. C. Loy. Recovering realistic texture in image super-resolution by deep spatial feature transform. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[39] J. Yan, G. He, and C. Hancock. Low-cost vision-based positioning system. In *International Conference on Location Based Services (LBS)*, January 2018.

[40] L. Yang, Y. Wang, X. Xiong, J. Yang, and A. K. Katsaggelos. Efficient video object segmentation via network modulation. *algorithms*, 29:15, 2018.