Deep Radio-Visual Localization

Tatsuya Ishihara[⋆‡] Kris M. Kitani[⋆] Chieko Asakawa[⋆†] Michitaka Hirose[‡]
[⋆]Carnegie Mellon University [†]IBM Research [‡]The University of Tokyo
tishihar@cmu.edu, kkitani@cs.cmu.edu, chiekoa@us.ibm.com, hirose@cyber.t.u-tokyo.ac.jp

Abstract

For many automated navigation applications, the underlying localization algorithm must be able to continuously produce both accurate and stable results by using a spectrum of redundant sensing technologies. To this end, various sensors have been used for localization, such as Wi-Fi, Bluetooth, GPS, LiDAR and cameras. In particular, a class of vision-based localization techniques using Structure from Motion (SfM) has been shown to produce very accurate position estimates in the real-world with moderate assumptions about the motion of the camera and the amount of visual texture in the environment. However, when these assumptions are violated, SfM techniques can fail catastrophically (i.e., cannot generate any estimate). Recently, a deep convolutional neural network (CNN) has been applied to images to robustly regress 6-DOF camera poses at the cost of lower accuracy than SfM. In this work, we propose improving image-based localization accuracy of deep CNN by combining Bluetooth radio-wave signal readings. In our experiments, we show that our proposed dual-stream CNN can robustly regress 6-DOF poses from images and radio-wave signals better than one sensing modality alone. More importantly, we show that when both modes are used, the localization accuracy of the proposed deep CNN is comparable to that of SfM and significantly more robust than SfM.

1. Introduction

Localization is essential for various applications, such as pedestrian navigation, augmented reality, and autonomous robot navigation. Various sensors have been studied for realizing accurate and robust localization systems, such as GPS, radio-wave signals, laser ranging scanners, and cameras [41]. In real-world situations, these sensors are often affected by unexpected noises. Thus, an accurate and robust localization system that continuously gives stable results is necessary for deploying navigation applications in the real-world.

Image based localization is a promising approach because cameras are already installed in commodity mobile devices, such as smartphones. Structure from Motion (SfM) is a common approach for image based localization. By matching local keypoints in a query image with keypoints in a 3D model, SfM can estimate an accurate 6-DOF camera pose. In general, SfM can estimate more accurate locations than radio-wave localization [38]. However, SfM based approaches have problems when an environment does not have enough distinctive visual features. This is because SfM based approaches rely on hand-crafted local keypoint descriptors, such as SIFT [27]. When environments have fewer visual features or many repetitive features, distinctive local keypoints are difficult to find. SfM often produces large errors or sometimes fails to localize in these difficult situations. Because of these problems, SfM based localization systems are more appropriate for texture rich scenarios.

Figure 1. Images where feature-based SfM fails but our proposed radio-visual localization network is successful.

Since many commercial navigation systems require more robustness than accuracy in localization, GPS and radio-wave based localization are commonly used. Although GPS works without installing devices in environments, its localization error is usually more than several meters and at times more than 10 meters when there are many nearby building structures. Moreover, it does not work in indoor environments where a GPS signal is not available. For indoor environments, Wi-Fi based localization is a traditional approach because many buildings already have Wi-Fi. Although Wi-Fi localization works in indoor environments, the localization error is generally still

978-1-5386-4886-5/18/$31.00 ©2018 IEEE
DOI 10.1109/WACV.2018.00071

more than several meters [16]. Furthermore, the positions of Wi-Fi routers are not placed to support device localization but rather strategically placed for efficient data transfer.

Bluetooth Low Energy (BLE) beacons are becoming popular for pedestrian localization [6] because they are easy to install in new environments and most smartphones can read BLE signals. BLE beacons are commercially available at low prices and can be installed both in indoor and outdoor environments. By installing enough beacons in an environment, we can realize more accurate localization than Wi-Fi [6], but the error of BLE based localization is generally still a few meters.

A deep convolutional neural network (CNN) has recently been applied to image based localization [23]. CNN based approaches directly regress 6-DOF poses from input images and use global context in images for localization. Unlike SfM based approaches, CNN based approaches do not need to detect local keypoints and are more robust to difficult conditions, such as fewer visual features, motion blur, and lighting condition changes. Also, CNN based approaches have advantages in terms of speed and memory efficiency when localizing images. However, CNN based approaches are less accurate than SfM based approaches because they do not make explicit use of 3D geometry.

In this work, we propose an approach to improve accuracy of CNN based image localization by incorporating robust radio-wave information. In our approach, both images and radio-wave signals are input to a dual-stream CNN and the network directly regresses 6-DOF camera poses. As far as we know, this is the first work to integrate radio-wave signals to CNN based image localization. Through our experiment, we will show that robustness of BLE signals helps to learn a more accurate CNN based localization model. Our results show that the proposed approach is more accurate than the state-of-the-art CNN based image localization. The proposed approach is significantly more robust and has comparable localization accuracy to SfM based localization (Figure 1 shows example images where SfM fails but our proposed approach is successful).

We emphasize here that our approach is not limited to BLE signals but can be used with other radio-wave signals, such as Wi-Fi. We also note that our approach does not require any prior knowledge about the position of BLE beacons in the environment. Thus, our approach is easy to apply in environments where BLE beacons are already installed. Because most smartphones have cameras and BLE sensors, the assumptions of our approach for localizing pedestrians are both practical and realistic.

2. Related Work

2.1. Keypoint based Localization

Traditional image based localization can be categorized into two major types of approaches: SfM based and image retrieval based approaches.

SfM based approaches need to build 3D models from collections of images before localizing images. Localization is done by first matching keypoints from a query image to keypoints in a 3D model and then estimating a camera pose by solving a PnP problem [25]. In SfM, hand-crafted local features are typically used. Reconstructing a large 3D model takes longer time, but recent advances of SfM makes it possible to build 3D models from a large collection of photos [35, 13, 17]. In spite of these advances, SfM still requires high computational cost both for 3D reconstruction and localization when 3D models become large. Although the localization is accurate in the environment with enough visual features, error will be large or localization will fail if the environments do not have enough distinctive visual features. This is because SfM approaches rely on matching local feature points.

Image retrieval based approaches estimate a position of a query image by finding the most similar images from database images [9, 15]. Image retrieval based approaches can localize images more quickly but less accurately than SfM based approaches. Also, it cannot estimate 6-DOF poses directly. Image retrieval based approaches typically create a feature vector for a query image by aggregating local keypoint descriptors. Similar to SfM based approaches, an environment with few visual features or many repetitive features makes localization difficult [37, 32]. A CNN has recently been used to build visual vocabularies for image retrieval [7, 31]. Although these approaches are more accurate than traditional keypoint based image retrieval approaches, they are still less accurate than SfM based approaches and cannot estimate 6-DOF poses directly.

2.2. Supervised Learning for Localization

A deep neural network was first successfully applied to object classification [24] and object detection [14]. This has also been applied in other areas, such as camera relocalization [23], relative camera motion [28], visual odometory [40], and RANSAC pose estimation [8].

Kendall *et al*. first proposed a CNN based image localization approach that directly regresses 6-DOF poses from input images [23]. Their approach is called PoseNet, and its network architecture is based on GoogLeNet [36]. PoseNet is more robust than SfM based approaches under difficult image conditions, such as feature-less environments. The CNN based approach is more suitable for real time applications. When using a GPU, the CNN based approach can localize one image in less than $10\mathrm{ms}$. Also, the localization speed and required memory do not change with the size of environment. Because the required image resolution is smaller (e.g. 224×224) than that in SfM based approaches, communication between mobile devices and the server is fast as well.

To improve the accuracy of CNN based image localization, various approaches have been proposed. In [20], Kendall and Cipolla proposed an approach to improve the accuracy of PoseNet by introducing uncertainty of prediction. In [21], Kendall and Cipolla proposed two new loss functions: the first loss function improves the accuracy of PoseNet by estimating the hyperparameter of multi-task learning, and the second loss function minimizes the 2D projection error of a 3D point cloud. They showed that using both loss functions improved the localization accuracy, but the second loss function requires a 3D point cloud and is not suitable for a texture-less environment in which SfM reconstruction is difficult. Thus, our approach uses only the first loss function to train our network. Walch *et al*. applied LSTM to incorporate the information of spatial context [39]. Clark *et al*. proposed applying bidirectional LSTM to utilize temporal information [10]. These various current approaches complement our approach and will be able to be integrated with it.

Similar to CNN based image localization approaches, different supervised learning approaches are applied to localize from RGBD camera input. Shotton *et al*. proposed using random forest for localization that used an RGBD camera [33]. They predict 3D coordinates of each pixel to estimate a camera pose. Li *et al*. proposed using an RGBD camera for CNN localization [26]. Similar to our approach, they used dual-stream CNN for estimating a camera pose using an RGB image and a depth image. We focused on RGB cameras because they are installed in most smartphones and our approach can be applied in various applications, such as pedestrian navigation systems.

2.3. Sensor Fusion for Localization

To improve the efficiency and accuracy of image based localization, fusing different types of sensors with images has been studied [4, 11, 18]. Clark *et al*. applied a probabilistic approach to integrate Wi-Fi signals with SfM based localization [11]. They used Wi-Fi signals to estimate visible 3D keypoints, and accelerated the step of keypoint matching. Ishihara *et al*. proposed an approach for BLE guided SfM localization [18] that searches for candidate match keypoints in a 3D model using BLE signals and realizes accurate and efficient localization. Although these approaches reduced the computational cost of feature matching and improved the accuracy of SfM localization using additional sensors, localization may still fail if the environment does not have enough visual features.

Although deep learning has been successfully applied in various applications, few studies have applied deep learning for radio-wave based localization. Nowicki and Wietrzykowski proposed a deep learning approach for Wi-Fi place recognition [30], but they focused on estimating rough locations and used an auto encoder to recognize floors. Our work directly regresses 6-DOF poses from radio wave signals, and can be combined with different types of CNN based image localization approaches.

3. Approach

The advantage of a deep CNN is its high accuracy and flexibility of models. Our dual-stream network is composed of two networks with different modalities: one network regresses 6-DOF poses from images, and the other regresses 6-DOF poses from radio-wave signals. We will describe how these two different sensors are processed in the following sections.

3.1. Image Network Architecture

For processing image information, we use PoseNet architecture [23]. In PoseNet, the input value $\boldsymbol{I}$ is an image, and the output value is a three dimensional camera position $\boldsymbol{x} \in \mathbf{R}^3$ and a four dimensional camera orientation $\boldsymbol{q} \in \mathbf{R}^4$ represented by quaternion. Loss function $L_\beta(\boldsymbol{I})$ for the input image I is defined as follows:

$$\begin{aligned} L_\beta(\boldsymbol{I}) &= L(\boldsymbol{I})_x + \beta L(\boldsymbol{I})_q \\ L(\boldsymbol{I})_x &= \|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_\gamma \\ L(\boldsymbol{I})_q &= \|\hat{\boldsymbol{q}} - \frac{\boldsymbol{q}}{\|\boldsymbol{q}\|}\|_\gamma \end{aligned} \tag{1}$$

Here, $\boldsymbol{x}$ and $\boldsymbol{q}$ are ground truth camera positions and rotation, and $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{q}}$ are their estimated values. $L(\boldsymbol{I})_x$ and $L(\boldsymbol{I})_q$ are loss functions for camera positions and rotations, respectively. β is a constant parameter for balancing positional loss with rotational loss. $\|\|_\gamma$ is the L1 norm if γ is 1 and the L2 norm if γ is 2. Because equation (1) optimizes both $L(\boldsymbol{I})_x$ and $L(\boldsymbol{I})_q$, PoseNet solves multi-task learning. The optimal β can be found by grid search.

PoseNet uses GoogLeNet architecture [36], and the L2 norm is used for the loss function. GoogLeNet has three output layers, and loss functions are calculated for all three to prevent the vanishing gradient problem. PoseNet also has three loss functions represented by equation (1).

Kendall and Cipolla [21] showed the weighting parameter β in the loss function can be replaced with trainable parameters by introducing homoscedastic uncertainty [22]. By introducing additional scalar parameters $\hat{s}_x, \hat{s}_q$, the loss function can be replaced by the following function:

$$L_s(\boldsymbol{I}) = L(\boldsymbol{I})_x \exp(-\hat{s}_x) + \hat{s_x} + L(\boldsymbol{I})_q \exp(-\hat{s}_q) + \hat{s_q} \tag{2}$$

$\hat{s}_x, \hat{s}_q$ represents the task specific uncertainty, and these parameters will be learned from the training data. L1 norm is used for the loss function (2). Our approach can be applied in general CNN based image localization approaches, so we used the approach that solves the loss function (2)

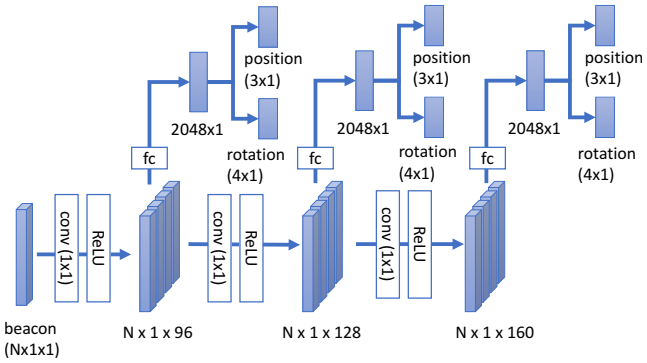

Figure 2. Architecture of network to process BLE signals. conv(1×1) represents a convolution layer. ReLU represents a rectified linear unit layer. fc represents a fully connected layer.

because it is more accurate than other CNN based image localization approaches.

3.2. Radio-Wave Network Architecture

Although CNN has been actively studied in many applications, it has not been studied thoroughly for radio-wave based localization. We propose a network architecture that directly regresses 6-DOF poses from radio-wave signals.

We focus on BLE devices because they are becoming commonly used in pedestrian navigation. BLE beacons continuously emits Bluetooth signals with a device ID at certain time intervals. The strength of BLE signal is represented by an RSSI (received signal strength indicator) value. In our implementation, we used the Apple iOS SDK and observe raw RSSI values from -99 (very weak) to 0 (very strong). Before inputting RSSI values to our network, we pre-process the values. For observed beacons, we add 100 to the raw RSSI value. For beacons that are not observed, we set the value as 0. Then, we will obtain the value from 0 to 100 for all beacons installed in the environment. In every time step, we have a fixed-size vector that can be input to a fixed-size network.

The network architecture for beacon signal is shown in Figure 2. If the environment has N beacons, the input data for network will be a $N \times 1 \times 1$ tensor. To combine a radio-wave network with an image network, we used a similar architecture to PoseNet. It is composed of three sub-networks, and each sub-network outputs a three dimensional position vector $\boldsymbol{x}$ and four dimensional orientation vector $\boldsymbol{q}$. Each sub-network has one 1×1 convolution layer and ReLU activation unit. Each output layer is connected with fully connected layers having 2048 nodes.

Note that our approach does not assume any prior knowledge about environments, such as where radio-wave transmitters are located. We only assume all IDs of BLE devices installed in the environment are known. Once we walk around the environment and scan the BLE signals, we can collect them. This step can be done at the same time as collecting training data, and there is no additional workload. Therefore, our approach is easy to apply in an environment where BLE beacons are already installed.

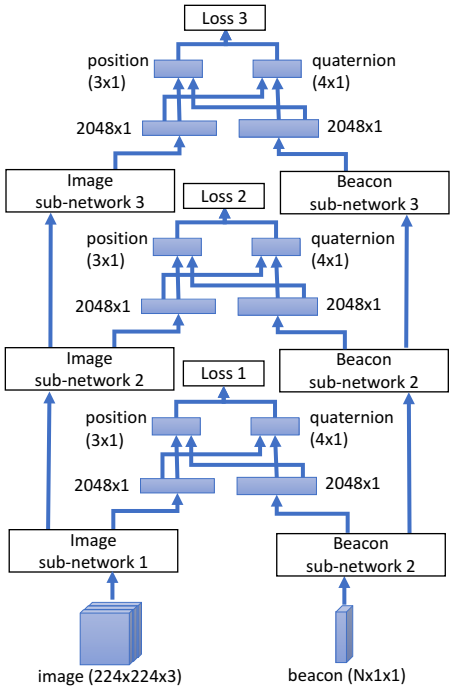

Figure 3. Overall architecture of the proposed dual-stream network

Because our radio-wave network can directly regress 6-DOF poses from radio-wave signals, the network architecture in Figure 2 can be used by itself for BLE based localization. In later experiments, we will show the localization results when only BLE signals were used.

3.3. Radio-Visual Localization Network

For inputting both image information and radio wave information, we combined PoseNet [23] and a radio-wave network as shown in Figure 3 as a dual-stream network. Multiple information has been successfully combined by using a dual-stream CNN in different areas, such as video action recognition [34, 12]. As far as we know, this is the first approach to combine radio-wave information and images in end-to-end learning. In our experiments, the input image is first resized to the resolution of 455×256, and then the center region of 224×224 is cropped in accordance with the settings of [23].

Both networks consist of three sub-networks and three output layers. To combine two different networks, we combined only output layers. Therefore, output variables for position and rotation are connected to two fully connected layers for both an image network and a radio-wave network. We have three loss functions. Each loss function is calculated by equation (2). Following GoogLeNet [36], the total

loss function is calculated by adding the first and the second auxiliary loss functions weighted by 0.3 to the last loss function. During test time, only the last output layer is used.

3.4. Image and Beacon Data Collection

A large amount of training data is needed to improve the accuracy of deep CNN. To train our network, we need images and BLE signals labeled with 6-DOF poses. Because manually labeling 6-DOF poses for a large set of images is practically impossible, we used a LiDAR to create ground truth 6-DOF poses. A LiDAR can generally achieve centimeter level localization accuracy [19]. We used Velodyne VLP-16 for LiDAR. To associate positions estimated by LiDAR and images, a LiDAR and a smartphone are fixed to a tripod at fixed locations. At the same time as the LiDAR point cloud is recorded, images and BLE signals are collected by the smartphone. We collected data by walking around the environments with this tripod.

3D maps of environments and 6-DOF poses of the LiDAR are calculated offline by using the LiDAR SLAM algorithm [19]. To collect large datasets, we need to record multiple times for the same environments. Because the coordinates of the 3D map created by SLAM will be different every time we record data, we need to align the coordinates of the 3D map. To align the coordinates, we first projected a 3D point cloud on a 2D map and then manually registered the 2D projected point cloud to the floor plan. The ground truth position of each image is calculated by transforming the 6-DOF poses in the 3D map to this registered map.

3.5. Beacon Data Augmentation

In image classification, data augmentation is often used to create more training data from existing training data [24]. By using data augmentation, we can prevent overfitting and improve the accuracy. When using data augmentation, the new data should be created by adding noises to original data while preserving the labels of the original data. In image classification, there are several means of data augmentation, such as flipping original images, cropping different areas of images, and changing intensities of RGB channels.

For beacon signals, signals fluctuate even at the same positions because of their interference with other radio-wave signals or obstacles. By considering this effect, we augment data by changing the values of observed beacons only. To simulate BLE signals weakened by the interference, we randomly change observed BLE signals to smaller values. When augmenting each data, we first randomly select a certain ratio of observed beacons and weakened selected signals by a random rate.

In our experiments, we set the ratio of randomly selected beacons to observed beacons as 0.1. For each randomly selected beacon, the rate to weaken the RSSI signal is sampled by uniform random variables from 0 to 1. For each training data, we create 5 augmented data.

	Area size	# Beacons	# Training	# Test
D1	63m × 42m	99	2265	739
D2	40m × 32m	59	2542	828
D3	58m × 60m	55	4147	1350
D4	29m × 28m	64	1492	507
D5	41m × 33m	90	2363	763
D6	40m × 50m	112	2491	833

Table 1. Dataset for localization evaluation. # Beacons, # Training, # Test show number of beacons, number of training video frames, and number of test video frames respectively.

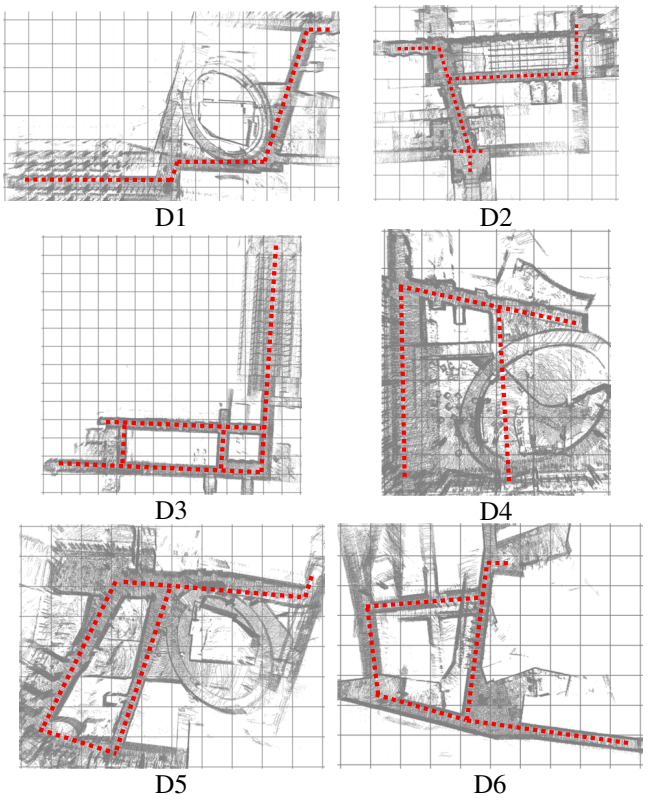

Figure 4. 3D map reconstructed by LiDAR SLAM. Dotted lines show paths along which training and test videos are recorded. Grid lines are drawn for $5\mathrm{m}^2$ areas.

4. Experiments

4.1. Dataset for Evaluation

By following the data collection steps described in the section 3.4, we collected several large scale indoor datasets. Images are captured at the resolution of 1280×720, and undistorted before training and test. An iPhone7 is used for collecting data. For recording both training and test data, images are captured at 2 fps. Bluetooth signals are captured at 1Hz by the iPhone (1Hz is a fixed setting of the iPhone). All images are associated with Bluetooth signals that are recorded at the closest timestamp.

D1 D2 D3 D4 D5 D6

Figure 5. Example image for each location

We created datasets for six different locations. The area size, number of beacons, and number of video frames are shown in Table 1. For each location, we recorded three videos for training and one video for testing. In each environment, BLE beacons were positioned every 4-6 meters. The locations of the BLE beacons were decided on the basis of a previous study to balance the localization accuracy and deployment cost [3]. Figure 4 shows a 3D point cloud created by LiDAR SLAM. Dotted lines show paths where training and test videos are recorded. We recorded two opposite directions on all paths. In all locations, we recorded the same path for training and test. As shown in this 3D map, we recorded in different area sizes and different path shapes. Figure 5 shows example images for all locations.

4.2. Baselines

In our following experiments, we compared the following baseline approaches with our proposed approach.

- SfM BoW : SfM based localization that uses BoW (bag-of-words) for image retrieval [29] to accelerate keypoint matching
- SfM BLE : SfM based localization that uses BLE signals to accelerate keypoint matching as proposed in [18]
- PoseNet β : PoseNet trained using the loss function in equation (1) as proposed in [23]
- PoseNet σ : PoseNet trained using the loss function in equation (2) as proposed in [21]

In SfM based baselines, keypoint matching requires a large computational cost. The baseline "SfM BoW" matches keypoints in a query image with keypoints in a 3D model that are extracted only from visually similar images. The visually similar images in the 3D model are selected by a common BoW based image retrieval approach [29]. The step of searching for candidate matching images significantly reduces the localization time. The baseline "SfM BLE" uses BLE signals to search for candidate matching images in addition to BoW [18]. The experimental results of [18] showed that "SfM BLE" is more accurate than "SfM BoW" even though computational cost is almost the same.

The implementation for baselines "SfM BoW" and "SfM BLE" is based on the open source implementation [1]. Because our dataset covers large areas, we modified a 3D reconstruction pipeline of the original implementation as follows. First, we separated each training video into short 60 frames video clips and applied SfM for all video clips. Each small 3D model has different 3D coordinates, and we need to convert them into the same coordinate to merge them. Because all video frames have ground truth positions estimated by LiDAR SLAM, we merged all 3D models by using these positions. For each 3D model, similarity transformation that convert camera positions in 3D model to ground truth positions is calculated. Then, all 3D models are merged into one large 3D model by applying similarity transformation.

For the baseline "SfM BoW", we selected 200 candidate matching images by using BoW image retrieval. For the baseline "SfM BLE", we first selected 400 candidate matching images by using BLE signals and then reduced candidate images to 200 by using BoW image retrieval. For both of these baselines, we used AKAZE [5] as a local feature detector and descriptors.

In all experiments for CNN based localization, the network is trained by stochastic gradient descent using Adam solver. The learning rate is set as 10^{-4}, the batch size as 64, and the number of training iterations as $30k$. For the baseline "PoseNet β", parameter β for weighting positional loss and rotational loss is set as 500. For the baseline "PoseNet σ", we need initial values for s_x, s_q. Following [21], we set these initial values as $s_x = 0.0$, $s_q = -3.0$. These CNN based baselines and our proposed approach are implemented in TensorFlow [2].

As described in PoseNet [23], the network weight for CNN based image localization is initialized by using the classification network trained by the Places database [42]. For initializing network weight for BLE beacons, we set random values as initial values.

4.3. Evaluation of Radio-Wave Network

First, we evaluated the localization accuracy of the proposed radio-wave CNN model. In this experiment, we used only the CNN model shown in Figure 2, and only BLE signals as the input data. The loss function for this radio-wave CNN model is calculated by using the equation (2) in the same way to the proposed dual-stream network.

Table 2 shows the results for average positional errors in meters and average rotational errors in degrees. We first evaluated the accuracy when we did not use beacon data augmentation described in Section 3.5. "BLE Net (w/o Aug.)" shows the results. The results show that our CNN model can estimate location only by one observation of

	BLE Net (w/o Aug.)		BLE Net	
	Pos. Error	Rot. Error	Pos. Error	Rot. Error
D1	1.26 ± 3.2	$67° \pm 52$	$\mathbf{1.12} \pm 3.2$	$68° \pm 52$
D2	1.09 ± 0.8	$68° \pm 53$	$\mathbf{1.04} \pm 0.7$	$69° \pm 51$
D3	$\mathbf{1.65} \pm 4.7$	$69° \pm 59$	1.70 ± 4.6	$71° \pm 59$
D4	1.16 ± 0.8	$66° \pm 53$	$\mathbf{1.12} \pm 0.8$	$65° \pm 53$
D5	1.05 ± 0.7	$69° \pm 54$	$\mathbf{1.00} \pm 0.7$	$66° \pm 53$
D6	1.03 ± 0.7	$57° \pm 48$	$\mathbf{0.87} \pm 0.6$	$56° \pm 46$

Table 2. Average and standard deviation of positional errors in meters and rotational errors in degrees for radio-wave network. Only BLE signals are input for the proposed radio-wave network. "BLE Net (w/o Aug.)" shows the results when beacon data augmentation is not used and "BLE Net" shows the results when beacon data augmentation is used. "Pos. Error" shows positional errors, and "Rot. Error" shows rotational errors.

BLE signals. The average localization error is less than 2.0 meters for all locations.

We also evaluated the accuracy when we used the proposed beacon data augmentation. "BLE Net" shows the results. The results shows that beacon data augmentation improved the localization accuracy in general and reduced localization error about $0.15\mathrm{m}$ at most. In following experimental results with BLE beacons, we used beacon data augmentation.

4.4. Evaluation of Radio-Visual Network

We then evaluated the accuracy of proposed dual-stream network. Table 3 compares CNN based baselines and our proposed approach. The results show average positional errors in meters, and average rotational errors in degrees. "PoseNet σ" can estimate locations more accurately than "PoseNet β" because it learns the optimal weight from the training data. For all six datasets, our approach improved the localization accuracy even more with the help of robust BLE signals. Our proposed approach reduced the average positional error about $0.2\mathrm{m}$ at most.

Table 5 shows 90 percentile localization errors for the "PoseNet σ" and our proposed approach. In the case of 90 percentile error, our approach reduced the positional error at most about $0.4\mathrm{m}$. As shown in these results, our proposed approach consistently has better localization accuracy than the state-of-the-art approach. One limitation of our approach is it has slightly worse rotational accuracy than baseline approaches. The difference is small (at most about 2 degrees), but it will be possible to use other baseline approaches only for rotation estimation and use our approach for position estimation for an application that requires accurate rotation estimation. The additional computational cost for this will be very small because CNN localization can process one image in less than $10\mathrm{ms}$.

Table 4 compares SfM based baselines and our proposed approach. "PROPOSED, SfM BLE" in Table 4 shows the

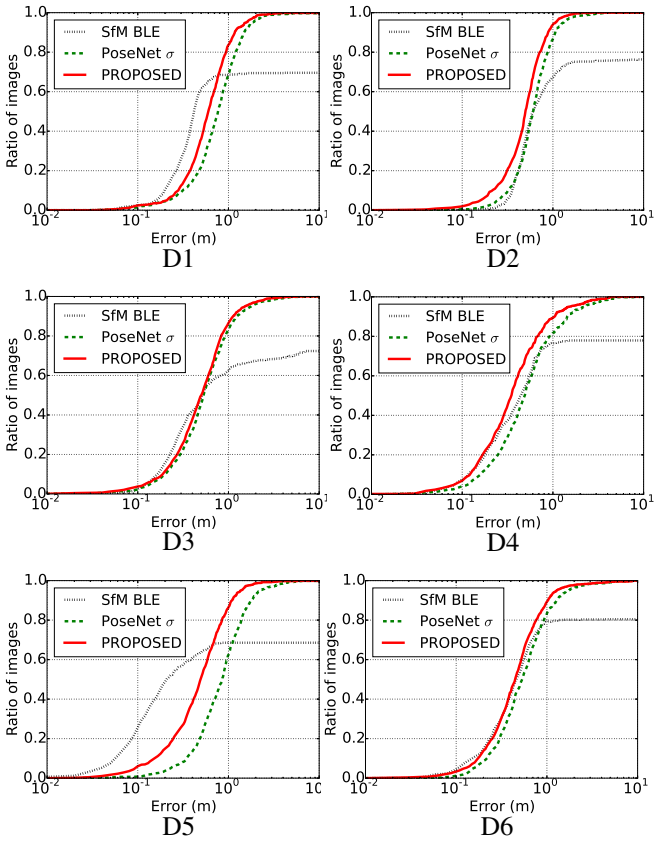

Figure 6. Cumulative localization errors.

results when "SfM BLE" is used at first and then our approach is used only for the image that "SfM BLE" cannot localize. As for the SfM baseline approaches, "SfM BLE" is more accurate than "SfM BoW" in general. "PROPOSED" in Table 3 is significantly more robust for all datasets and even more accurate than SfM based approaches for two datasets (D2, D3). Figure 6 shows the cumulative localization error for "SfM BLE", "PoseNet σ" and our proposed approach. As shown in these results, CNN based approaches are significantly more robust than "SfM BLE" and our proposed approach is more accurate than "PoseNet σ".

When SfM works well (i.e. in feature rich environments), "PROPOSED, SfM BLE" is slower than CNN based approached but can be the more accurate choice as shown in Table 4. Because SfM baselines will require about 0.5 seconds for localizing a image, "PROPOSED, SfM BLE" will take about 0.5 seconds longer than "PROPOSED".

4.5. Evaluation of Localization Speed

We evaluated the speed of localization. CNN based image localization is much faster than SfM based localization, and the speed is not dependent on area size. We evaluated localization speed for our datasets. Table 6 shows the aver-

	PoseNet β [23]		PoseNet σ [21]		PROPOSED	
	Pos. Error	Rot. Error	Pos. Error	Rot. Error	Pos. Error	Rot. Error
D1	1.06 ± 1.1	$4.2° \pm 4.8$	0.91 ± 0.8	$4.2° \pm 4.9$	$\mathbf{0.72} \pm 0.6$	$6.3° \pm 7.3$
D2	0.95 ± 0.8	$2.6° \pm 6.5$	0.69 ± 0.4	$2.3° \pm 3.3$	$\mathbf{0.55} \pm 0.3$	$2.8° \pm 5.6$
D3	1.19 ± 1.1	$3.6° \pm 4.0$	0.71 ± 0.7	$4.4° \pm 7.9$	$\mathbf{0.64} \pm 0.6$	$5.9° \pm 13.5$
D4	0.96 ± 0.8	$3.2° \pm 5.8$	0.73 ± 0.8	$3.5° \pm 5.7$	$\mathbf{0.53} \pm 0.6$	$4.8° \pm 8.3$
D5	1.01 ± 1.4	$3.8° \pm 9.9$	0.80 ± 1.2	$3.7° \pm 7.8$	$\mathbf{0.63} \pm 0.9$	$4.2° \pm 7.7$
D6	1.06 ± 1.4	$3.0° \pm 5.3$	0.73 ± 0.8	$3.5° \pm 8.3$	$\mathbf{0.61} \pm 0.8$	$5.0° \pm 4.5$

Table 3. Average and standard deviation of positional errors in meters and rotational errors in degrees for CNN based baselines and our approach

	SfM BoW			SfM BLE [18]			PROPOSED, SfM BLE		
	Pos. Error	Rot. Error	Succ.	Pos. Error	Rot. Error	Succ.	Pos. Error	Rot. Error	Succ.
D1	0.67 ± 0.6	$12.9° \pm 19.3$	83%	0.41 ± 0.7	$14.9° \pm 31.6$	70%	0.51 ± 0.8	$12.0° \pm 27.0$	**100%**
D2	0.58 ± 0.7	$13.8° \pm 25.6$	91%	0.73 ± 1.0	$6.0° \pm 3.6$	77%	0.69 ± 0.9	$5.8° \pm 6.4$	**100%**
D3	1.04 ± 2.4	$18.8° \pm 32.5$	84%	0.98 ± 2.3	$17.1° \pm 35.6$	74%	0.91 ± 2.0	$14.4° \pm 31.6$	**100%**
D4	0.48 ± 0.4	$13.1° \pm 11.6$	88%	0.38 ± 0.2	$21.3° \pm 26.6$	78%	0.42 ± 0.3	$18.1° \pm 24.7$	**100%**
D5	0.34 ± 0.4	$9.7° \pm 6.0$	85%	0.19 ± 0.2	$9.4° \pm 5.6$	69%	0.37 ± 0.9	$8.1° \pm 8.5$	**100%**
D6	0.46 ± 0.5	$11.8° \pm 5.3$	89%	0.43 ± 0.3	$11.9° \pm 4.1$	80%	0.49 ± 0.5	$10.8° \pm 5.0$	**100%**

Table 4. Average and standard deviation of positional errors in meters and rotational errors in degrees for SfM based baselines and our approach combined with "SfM BLE". "PROPOSED, SfM BLE" is evaluated by using "SfM BLE" at first and then using our approach only for the images that "SfM BLE" cannot localize. "Succ." shows the percentage of test frames that are localized.

	PoseNet σ		PROPOSED	
	Pos. Error	Rot. Error	Pos. Error	Rot. Error
D1	1.50	6.2°	**1.18**	4.9°
D2	1.07	4.8°	**0.90**	5.1°
D3	1.26	8.2°	**1.13**	9.3°
D4	1.44	7.6°	**1.05**	9.7°
D5	1.48	5.5°	**1.10**	6.1°
D6	1.30	5.1°	**1.04**	7.4°

Table 5. 90 percentile positional errors in meters and rotational errors in degrees

SfM BLE	PoseNet σ	BLE Net	PROPOSED
0.545	0.007	0.002	0.007

Table 6. Average time to localize one image (seconds).

age time to localize one image for all six datasets. For the localization server, we used a PC with an Intel Xeon CPU E5-2660 v3 2.60 GHz (10 cores) processor with an NVIDIA TITAN X (Pascal) GPU.

"BLE Net" shows the result of our proposed approach when inputting only BLE signals. "PROPOSED" shows the result for our proposed approach when inputting both images and BLE signals. Although our approach has a dual-stream network and requires slightly more computational cost than "PoseNet σ", the average time to localize an image is same. Both "PROPOSED" and "PoseNet σ" can localize one image in less than $10\mathrm{ms}$ and are much faster than "SfM BLE". Our approach and other CNN based approaches will be suitable for real time applications.

5. Conclusion

We proposed an approach to improve the accuracy of deep CNN based image localization by integrating radio-wave information. In our experiments, we first showed the proposed radio-wave CNN model can directly regress 6-DOF poses only by using radio-wave signals. Then, we showed the proposed radio-visual network is more accurate than the state-of-the-art CNN based image localization. Our approach reduced the average localization error about $0.2\mathrm{m}$ at most, and the 90 percentile localization error about $0.4\mathrm{m}$ at most compared to the state-of-the-art CNN based image localization. We also showed that our approach is significantly more robust than SfM based approaches. We integrate our approach to a PoseNet based model, but it can be applied to other CNN based image localization approaches that considers spatial context or temporal context of images.

Our approach is not limited by BLE signals, and can be used with other radio-wave signals, such as Wi-Fi. Also, our approach does not require any additional prior knowledge about environments, such as locations of radio-wave transmitters. Because most smartphones have cameras and BLE sensors, our approach can be applied in wider application areas.

Acknowledgment

This work was supported in part by Shimizu Corporation, JST CREST grant (JPMJCR14E1) and NSF NRI grant (1637927).

References

[1] Human-scale localization platform (HULOP). https://github.com/hulop/SfMLocalization.

[2] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.

[3] D. Ahmetovic, C. Gleason, C. Ruan, K. Kitani, H. Takagi, and C. Asakawa. NavCog: A navigational cognitive assistant for the blind. In *Int. Conf. on Human-Computer Interaction with Mobile Devices and Services*, pages 90–99. ACM, 2016.

[4] A. Alahi, A. Haque, and L. Fei-Fei. RGB-W: When vision meets wireless. In *IEEE Int. Conf. on Computer Vision*, pages 3289–3297, 2015.

[5] P. F. Alcantarilla, J. Nuevo, and A. Bartoli. Fast explicit diffusion for accelerated features in nonlinear scale spaces. In *British Machine Vision Conf. (BMVC)*, 2013.

[6] S. C. Alliance. Bluetooth Low Energy (BLE) 101: A technology primer with example use cases. 2014.

[7] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic. NetVLAD: CNN architecture for weakly supervised place recognition. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 5297–5307. IEEE, 2016.

[8] E. Brachmann, A. Krull, S. Nowozin, J. Shotton, F. Michel, S. Gumhold, and C. Rother. DSAC - Differentiable RANSAC for camera localization. In *IEEE Conf. on Computer Vision and Pattern Recognition*. IEEE, 2017.

[9] D. M. Chen, G. Baatz, K. Köser, S. S. Tsai, R. Vedantham, T. Pylvänäinen, K. Roimela, X. Chen, J. Bach, M. Pollefeys, et al. City-scale landmark identification on mobile devices. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 737–744. IEEE, 2011.

[10] R. Clark, S. Wang, A. Markham, N. Trigoni, and H. Wen. VidLoc: 6-DoF video-clip relocalization. In *IEEE Conf. on Computer Vision and Pattern Recognition*. IEEE, 2017.

[11] R. Clark, S. Wang, H. Wen, N. Trigoni, and A. Markham. Increasing the efficiency of 6-DoF visual localization using multi-modal sensory data. In *IEEE Conf. on Humanoid Robots*, pages 973–980. IEEE, 2016.

[12] C. Feichtenhofer, A. Pinz, and A. Zisserman. Convolutional two-stream network fusion for video action recognition. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1933–1941. IEEE, 2016.

[13] J.-M. Frahm, P. Fite-Georgel, D. Gallup, T. Johnson, R. Raguram, C. Wu, Y.-H. Jen, E. Dunn, B. Clipp, S. Lazebnik, et al. Building rome on a cloudless day. In *European Conf. on Computer Vision*, pages 368–381. Springer, 2010.

[14] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 580–587. IEEE, 2014.

[15] J. Hays and A. A. Efros. Large-scale image geolocalization. In *Multimodal Location Estimation of Videos and Images*, pages 41–62. Springer, 2015.

[16] S. He and S.-H. G. Chan. Wi-Fi fingerprint-based indoor positioning: Recent advances and comparisons. *IEEE Communications Surveys & Tutorials*, 18(1):466–490, 2016.

[17] J. Heinly, J. L. Schonberger, E. Dunn, and J.-M. Frahm. Reconstructing the world* in six days*(as captured by the yahoo 100 million image dataset). In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 3287–3295. IEEE, 2015.

[18] T. Ishihara, J. Vongkulbhisal, K. M. Kitani, and C. Asakawa. Beacon-guided structure from motion for smartphone-based navigation. In *IEEE Winter Conf. on Applications of Computer Vision*, pages 769–777. IEEE, 2017.

[19] S. Kato, E. Takeuchi, Y. Ishiguro, Y. Ninomiya, K. Takeda, and T. Hamada. An open approach to autonomous vehicles. *IEEE Micro*, 35(6):60–68, 2015.

[20] A. Kendall and R. Cipolla. Modelling uncertainty in deep learning for camera relocalization. In *IEEE Int. Conf. on Robotics and Automation*, pages 4762–4769. IEEE, 2016.

[21] A. Kendall and R. Cipolla. Geometric loss functions for camera pose regression with deep learning. In *IEEE Conf. on Computer Vision and Pattern Recognition*. IEEE, 2017.

[22] A. Kendall, Y. Gal, and R. Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. *arXiv preprint arXiv:1705.07115*, 2017.

[23] A. Kendall, M. Grimes, and R. Cipolla. PoseNet: A convolutional network for real-time 6-DOF camera relocalization. In *IEEE Int. Conf. on Computer Vision*, pages 2938–2946. IEEE, 2015.

[24] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 1097–1105, 2012.

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

[26] R. Li, Q. Liu, J. Gui, D. Gu, and H. Hu. Indoor relocalization in challenging environments with dual-stream convolutional neural networks. *IEEE Transactions on Automation Science and Engineering*, 2017.

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

[28] I. Melekhov, J. Kannala, and E. Rahtu. Relative camera pose estimation using convolutional neural networks. *arXiv preprint arXiv:1702.01381*, 2017.

[29] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *IEEE Conf. on Computer Vision and Pattern Recognition*, volume 2, pages 2161–2168. IEEE, 2006.

[30] M. Nowicki and J. Wietrzykowski. Low-effort place recognition with WiFi fingerprints using deep learning. In *International Conference Automation*, pages 575–584. Springer, 2017.

[31] F. Radenović, G. Tolias, and O. Chum. CNN image retrieval learns from BoW: Unsupervised fine-tuning with hard examples. In *European Conference on Computer Vision*, pages 3–20. Springer, 2016.

[32] T. Sattler, M. Havlena, K. Schindler, and M. Pollefeys. Large-scale location recognition and the geometric burstiness problem. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1582–1590. IEEE, 2016.

[33] J. Shotton, B. Glocker, C. Zach, S. Izadi, A. Criminisi, and A. Fitzgibbon. Scene coordinate regression forests for camera relocalization in RGB-D images. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 2930–2937. IEEE, 2013.

[34] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in Neural Information Processing Systems*, pages 568–576, 2014.

[35] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: exploring photo collections in 3D. In *ACM Transactions on Graphics*, volume 25, pages 835–846. ACM, 2006.

[36] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1–9, 2015.

[37] A. Torii, J. Sivic, M. Okutomi, and T. Pajdla. Visual place recognition with repetitive structures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(37):2346–2359, 2015.

[38] S. Treuillet and E. Royer. Outdoor/indoor vision-based localization for blind pedestrian navigation assistance. *Int. J. of Image and Graphics*, 10(04):481–496, 2010.

[39] F. Walch, C. Hazirbas, L. Leal-Taixé, T. Sattler, S. Hilsenbeck, and D. Cremers. Image-based localization with spatial LSTMs. *arXiv preprint arXiv:1611.07890*, 2016.

[40] S. Wang, R. Clark, H. Wen, and N. Trigoni. DeepVO: Towards end-to-end visual odometry with deep recurrent convolutional neural networks. In *IEEE Int. Conf. on Robotics and Automation*, pages 2043–2050. IEEE, 2017.

[41] J. Xiao, Z. Zhou, Y. Yi, and L. M. Ni. A survey on wireless indoor localization from the device perspective. *ACM Computing Surveys (CSUR)*, 49(2):25, 2016.

[42] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In *Advances in Neural Information Processing Systems*, pages 487–495, 2014.