Journal of Information Processing Vol.26 707–717 (Oct. 2018)

[DOI: 10.2197/ipsjjip.26.707]

Regular Paper

Deep Learning and Geometry-based Image Localization Enhanced by Bluetooth Signals

Tatsuya Ishihara[1,2,a)] Kris M. Kitani[2,b)] Chieko Asakawa[2,3,c)] Michitaka Hirose[1,d)]

Received: December 22, 2017, Accepted: July 10, 2018

Abstract: For many automated navigation applications, the underlying localization algorithm must be able to continuously produce results that are both accurate and stable. To date, various types of localization approaches including GPS, Wi-Fi, Bluetooth and cameras have been studied extensively. Image-based localization approaches have been developed by using commodity devices, such as smartphones, and these have been shown to produce accurate localization systems. However, image-based localization approaches do not work well in environments that lack visual features. Therefore, we propose a novel approach that combines the use of radio-wave information with computer vision-based localization. In particular, we assume that Bluetooth low energy (BLE) devices are already installed in the environment. We integrate radio-wave information with two types of well-known image-based localization approaches: a Structure from Motion (SfM) based approach and a deep convolutional neural network (CNN) based approach. Our experimental results show that both image-based localization approaches can be more accurate when combined with radio-wave signals. The results also show that the localization accuracy of the proposed deep CNN approach is comparable to that of SfM and significantly more robust than it. In addition, the proposed deep CNN approach was found to be robust to BLE device failures.

Keywords: deep learning, Structure from Motion, Bluetooth low energy beacon, localization system

1. Introduction

Localization is essential for various applications, such as pedestrian navigation, augmented reality, location based service, and autonomous robot navigation. Up till now, various sensors have been utilized to realize accurate and robust localization systems, such as GPS, radio-wave signals, laser ranging scanners, and cameras [10]. In real-world situations, these sensors are often affected by unexpected noises. Thus, an accurate and robust localization system that continuously produces stable results is essential for real-world deployment of navigation applications.

To deploy localization systems widely in the real-world, it is also important to realize localization systems by using only commodity mobile devices. Image-based localization is a promising approach because the cameras are already installed in most smartphones. Previous studies showed that image-based localization can accurately estimate locations in environments with rich visual features [27].

Approaches to image-based localization can be categorized into two well-known types of approaches. One is a Structure from Motion (SfM) based approach, which uses local descriptors for keypoints, and the other is a deep convolutional neural network (CNN) based approach, which extracts global features from entire input images. These two approaches have different characteristics.

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 based localization [35]. 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 [21], and when environments contain few visual features or many repetitive features, distinctive local keypoints are difficult to find. Consequently, SfM often produces large errors or fails to localize in these difficult situations. Because of these problems, SfM-based localization systems are more appropriate for texture rich scenarios.

A deep convolutional neural network (CNN) has recently been applied to image-based localization [17]. CNN-based approaches directly regress 6-DOF camera poses from input images and use global context in images for localization. CNN-based approaches are less accurate than SfM-based approaches because they do not make explicit use of 3D geometry. In spite of this disadvantage, CNN-based approaches do not need to detect local keypoints and are more robust to difficult conditions than SfM-based approaches, 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. CNN-based approaches are more appropriate for feature

1 The University of Tokyo, Bunkyo, Tokyo 113–8654, Japan
2 Carnegie Mellon University, 5000 Forbes Ave, Pittsburgh, PA 15213, USA
3 IBM Research, 1101 Kitchawan Rd, Yorktown Heights, NY 10598, USA
a) ishihara@cyber.t.u-tokyo.ac.jp
b) kkitani@cs.cmu.edu
c) chiekoa@us.ibm.com
d) hirose@cyber.t.u-tokyo.ac.jp

less scenarios.

Recent advances of these two image-based localization approaches have improved accuracy and robustness. However, it is generally difficult to distinguish scenes with similar appearances throughout the use of images alone. Especially in indoor scenes, there are many similar scenes, such as similar-looking corridors in the same building. As a result, both SfM-based approaches and CNN-based approaches will produce large localization errors in such a case. Therefore, we used an approach combining radio-wave signals with image-based localization. For radio-wave signals, we focused on Bluetooth signals because Bluetooth Low Energy (BLE) beacons are currently becoming popular for pedestrian localization [5]. BLE beacons are easy to install in new environments and most smartphones can read BLE signals.

In this work, we will show that both SfM-based localization and CNN-based image localization will be more accurate by incorporating robust radio-wave information. First, we propose an approach to combine SfM-based localization with BLE signals. In our SfM-based approach, BLE signals are used to restrict the area of a 3D model in a feature matching process. Then, we propose an approach to combine deep CNN-based image localization with BLE signals. In our CNN based-approach, both images and radio-wave signals are input to a dual-stream CNN and the network directly regresses 6-DOF camera poses. Through our experiment, we will show both of the proposed SfM-based approach and CNN-based approach are more accurate than existing image-based localization approaches. The proposed CNN-based approach is promising because it is significantly more robust and has comparable localization accuracy to SfM-based localization. Our CNN-based approach is also robust to BLE device failures. 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 regarding the position of BLE beacons in the environment. Thus, our approach is flexible and 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 Image-based Localization

2.1.1 Geometry-based Localization

In general, an SfM pipeline uses the following three main steps to build a 3D model. First, it extracts features from images. Next, it matches the features and finds pairs of images that contain overlapping views. Finally, it estimates camera poses and 3D points from these pairs and builds a 3D model. The second step generally requires checking all pairs of images, so the computational costs of building large 3D models is very high. Previous studies have found image pairs efficiently by using a vocabulary tree based approach [25], and have created 3D models from millions of photos from the Internet [2], [11].

In spite of these advances, it is still difficult to create 3D models in environments that have a large number of repetitive features or few visual features. In such environments, good feature matches are often discarded by a conventional ratio test [21]. Shah et al. proposed an approach that uses epipolar geometry to add matches that would otherwise be discarded by a ratio test [31]. In our approach, we assume accurate knowledge of the positions of all images that are used for 3D reconstruction. We used a LiDAR sensor to create accurate positions of images, and feature matching was done only for image pairs that are taken in close positions. This reduced the repetition of similar features in the matching process, and prevented good matches from being discarded by the ratio test.

The feature matching process also involves a high computational cost for the localization process. Thus, Schonberger et al. proposed a supervised approach that finds image pairs more efficiently [30]. To find image pairs efficiently, our SfM approach includes the use of radio-wave signals to identify images within close radial proximity in addition to the use of visual information.

2.1.2 Deep CNN-based Localization

A deep neural network was first successfully applied to object classification [18] and object detection [9]. It has also been applied in other areas, such as camera relocalization [17], visual odometory [37], and RANSAC pose estimation [6].

Kendall et al. first proposed a CNN-based image localization approach that directly regresses 6-DOF poses from input images [17]. Their approach is called PoseNet, and its network architecture is based on GoogLeNet [33]. 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 only less than 10 ms. Also, the localization speed and required memory do not change with the size of the environment.

To improve the accuracy of CNN-based image localization, various approaches have been proposed. For example, in Ref. [14], Kendall and Cipolla proposed an approach to improve the accuracy of PoseNet by introducing the concept of "uncertainty of prediction". In Ref. [15], 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 errors of a 3D point cloud. They showed that using both loss functions improved the localization accuracy, but the use of 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 introduce the concept of spatial context [36]. Clark et al. similarly attempted to improve the accuracy of CNN-based image localization by applying bidirectional LSTM to utilize temporal information [7]. These various current approaches complement our CNN based approach and will be able to be integrated with our approach.

Similar to CNN-based image localization approaches, different supervised learning approaches have been applied to localize from an RGBD camera input. For example, Shotton et al. proposed using random forest for localization that used an RGBD camera [32]. They predicted the 3D coordinates of each pixel to estimate a camera pose. Also, Li et al. proposed using an

RGBD camera for CNN localization [19]. 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 the use of RGB cameras because they are installed in most smartphones and our approach can thus be applied in various applications, such as pedestrian navigation systems.

2.2 Other Sensors for Localization

In many commercial navigation systems, GPS and radio-wave based localization are commonly used. Although GPS works without installing devices in environments, its localization error is large when there are many buildings nearby. 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 access points. Radio-wave based localization is typically done by collecting radio-wave signals at many points in the target area. This process is called "fingerprinting". If the environment changes or radio-wave transmitters have problems, it is necessary to do fingerprinting again to update the map of the radio-wave signals. Taniuchi et al. proposed an approach to update the map of Wi-Fi signals without an extensive fingerprinting process [34]. Although Wi-Fi localization works in indoor environments, the localization error is generally still more than several meters [10]. Furthermore, the positions of Wi-Fi access points are not placed to support device localization but rather are strategically placed for efficient data transfer.

Nowadays, BLE beacons are becoming popular for localization [3]. BLE beacons are available at a low cost, and can be easily installed in new environments. By installing enough beacons in an environment, we can produce more accurate localization than with Wi-Fi [5]. One disadvantage of the BLE-based approach is the high cost of maintaining BLE beacons, which require batteries to emit consistent BLE signals. Sano et al. proposed a BLE-based localization approach that is robust even in the case of BLE device failures [28]. Even so, they focused only on rough localization.

Although deep learning has been successfully applied in various applications, to date, few studies have applied deep learning for radio-wave based localization. Nowicki and Wietrzykowski proposed a deep learning approach for Wi-Fi place recognition [26], but they focused on estimating rough locations alone and used an auto encoder to recognize floors. By contrast, our work directly regresses 6-DOF poses from radio-wave signals, and can be combined with different types of CNN-based image localization approaches.

Magnetic fields that are specific to buildings can be also used for localization. For example, Murata et al. used magnetic data to produce an indoor pedestrian navigation system [23]. Because it is difficult to localize in large areas by using only magnetic data, Higashi et al. combined magnetic sensor data with Wi-Fi signals [12]. Their approach improved the localization accuracy in the areas where Wi-Fi localization does not work well, but the accuracy is limited because of the Wi-Fi signal fluctuation.

There are other devices that can be used for localization. For example, Nakamura et al. used common speakers and a microphone on a smartphone to localize the 3D position of the smartphone [24], focusing on small areas to localize. Although their approach produces accurate localization results, acoustic localization approaches have a problem in a large area because of the non-line-of-sight problem. Using a different approach, Sawada et al. proposed a method of Wi-Fi beacon localization [29]. They produced accurate Wi-Fi based localization, but the cost of installing these devices is higher than BLE beacons. In contrast to these approaches, our approach uses common BLE beacons and smartphones.

2.3 Sensor Fusion for Localization

To improve the efficiency and accuracy of image-based localization, an approach based on fusing different types of sensors with SfM has been studied. For example, Clark et al. applied a probabilistic approach to integrate Wi-Fi signals with SfM-based localization [8]. Similar to our approach, they used Wi-Fi signals to estimate visible 3D keypoints, and accelerated the step of keypoint matching. They used particle filter to restrict the area of feature matching. While their approach focused on sequences of images and Wi-Fi signals, our SfM approach can estimate location efficiently, requiring only a single observation of image and BLE signals.

An inertial measurement unit (IMU) can be used to produce pedestrian dead reckoning (PDR) [38]. PDR can estimate relative movement by using sequences of IMU data. By combining other localization methods that can estimate global positions, PDR helps reducing localization errors. In this work, we focused on estimating global positions from images and BLE signals alone but it is possible to combine our approach with PDR approaches.

3. Approach

3.1 Geometry-based Image Localization with BLE

In this section, we discuss the details of our SfM-based localization with BLE signals. For SfM-based localization, we need to build 3D models. First, we will discuss our 3D model reconstruction approach that can be applied in large environments; then, we will discuss our SfM-based localization approach.

3.1.1 Large 3D Model Reconstruction

To realize localization in large areas, we applied a hierarchical reconstruction approach by using groundtruth positions obtained by a LiDAR. First, we separated each training video into short 60-frames video clips and applied SfM to all of the video clips. By applying SfM to separated small video clips, matching feature points was easier even in the environments that had lots of repetitive features or fewer visual features.

Each small 3D model had different 3D coordinates, and we needed to convert their coordinates into the same coordinate in order to merge them. Because all of the video frames had ground truth positions estimated by the LiDAR, 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. Next, all 3D models are merged into one large 3D model by applying similarity transformation.

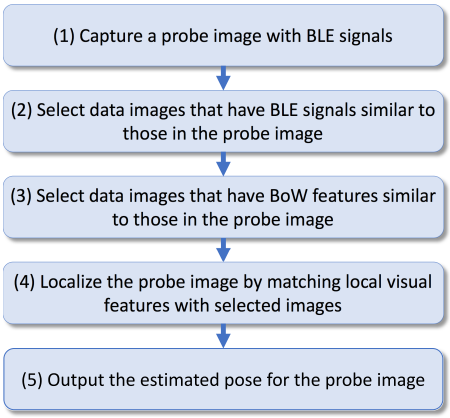

Fig. 1 Overview of SfM Localization process using BLE signals.

3.1.2 SfM-based Localization with BLE signals

We propose an approach that utilizes both radio-wave and visual information for SfM localization. Before performing localization, we assume that a 3D model of an environment has already been reconstructed as discussed in Section 3.1.1. The 3D model includes images that are used for 3D reconstruction, BLE signals associated with the images, and 3D structure points with corresponding local features extracted from the images. Here, we will refer to the image to be localized as the "probe image" and images used for reconstruction as "data images."

Figure 1 shows an overview of our localization process. First, a probe image is captured together with BLE signals and input into the system. Next, data images with similar BLE signals are selected to restrict the matching to only images in the same proximity as the probe image. The selected images are then further reduced to those having Bag-of-Words (BoW) features similar to the probe image. After that, feature matching is performed between the probe image and the remaining data images. Finally, pose estimation is performed as the final step.

Note that BLE beacon signals and visual information have different roles here. While BLE beacon signals can be used to select data images in the same proximity, they do not provide directional information. On the other hand, visual information can provide directional information, but it can be easily confused by scenes with similar appearances that are actually far apart, such as similar-looking corridors in different wings of a building. By combining both types of information, we are able to select data images that are recorded close to a probe image, and in a similar direction. Since we are able to calculate the BLE beacon signal and BoW similarity efficiently, this provides a significant improvement in computation time.

3.1.3 Comparing BLE signals

We now discuss the details of comparing BLE signals in step (2) of Fig. 1. BLE beacons provide a cost-efficient solution for short-range passive communication. A BLE beacon periodically broadcasts signals containing the beacon's ID [5]. A BLE receiver measures the received signal strength indication (RSSI), and the distance from the receiver to the beacon can be obtained from the signal strength. However, due to the instability of the RSSI, the average localization error of BLE-based approaches may be as many as several meters [10], preventing their direct use. Instead, we use BLE signals to estimate the proximity of the receiver in an environment by using two measures. We assume BLE beacons are installed in the environment such that they have overlapping ranges. Note that we are not required to know the specific positions of the beacons, making the installation of BLE beacons a simple task.

Let n be the number of BLE beacons, and $\hat{b}_i \in \mathbf{R}, i = 1, \ldots, n$ be the raw RSSI from beacon i. We set the valid range of $\hat{b}_i$ measured by smartphones from 0 (close to the beacon) to -99 (far from the beacon), and we set the RSSI signal as -100 for the beacon i that cannot be detected. We first normalize $\hat{b}_i$ to non-negative values with the median value of 100 with the following equation:

$$b_i = 100 \left(\frac{100 + \hat{b}_i}{100 + \underset{j:\hat{b}_j \neq -100}{\text{median}} \{\hat{b}_j\}} \right), j = 1, \ldots, n. \tag{1}$$

Note that the b_i of an undetected beacon will be normalized to 0. Let $\boldsymbol{b} = [b_1, \ldots, b_n] \in \mathbf{R}^n$ be a vector containing all normalized RSSIs.

To compare RSSI vectors, we define *beacon dissimilarity* and *beacon co-occurrence*. Suppose we have two RSSI vectors $\boldsymbol{b}^x$ and $\boldsymbol{b}^y$ recorded at two positions x and y; the beacon dissimilarity is defined as the mean absolute difference of mutually detected beacon signals:

$$d(\boldsymbol{b}^x, \boldsymbol{b}^y) = \frac{\sum_{i:b_i^x>0, b_i^y>0} |b_i^x - b_i^y|}{\sum_{i:b_i^x>0, b_i^y>0} 1}. \tag{2}$$

The reason that we compare only mutually detected signals is because of the instability of BLE signals: a receiver may not detect a signal even when it is close to the beacon, so comparing RSSI records at the same location but at different times could produce a big difference if all beacons are compared. However, one problem with beacon dissimilarity is that it can suggest a small difference even if the two positions are far away. For instance, if we have two beacon vectors from two different positions and the beacon vectors have only one mutually observed beacon, the dissimilarity can be small if the RSSIs are similar. To prevent this problem, we use beacon co-occurrence as a "soft" count of the numbers of mutually detected beacons. It is defined as follows:

$$c(\boldsymbol{b}^x, \boldsymbol{b}^y) = \frac{\sum_{i=1}^n \min\left(b_i^x, b_i^y\right)}{\sum_{i=1}^n \max\left(b_i^x, b_i^y\right)} \tag{3}$$

These two measures complement each other. Beacon dissimilarity measures the difference in RSSI values, while co-occurrence ensures that there are enough mutually detected beacons for the dissimilarity to be meaningful.

When selecting data images in step (2) of Fig. 1, we first select images whose beacon co-occurrence is above a certain threshold. Then, we select images with the smallest beacon dissimilarities from these images. As a result, we obtain data images in which similar sets of beacons are observed and that have similar RSSI signals.

3.2 Deep CNN-based Image Localization with BLE

In this section, we discuss the details of our deep CNN-based localization approach using BLE signals. The advantage of a deep CNN-based approach is its high accuracy and the flexibility of models. Our novel 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.

3.2.1 Image Localization Network

For processing image information, we use the PoseNet architecture [17]. In PoseNet, the input value $\boldsymbol{I}$ is raw pixel values of 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. The loss function $L_\beta(\boldsymbol{I})$ for the input image I is defined as follows:

$$\begin{aligned} L_\beta(\boldsymbol{I}) &= L(\boldsymbol{I})_x + \beta L(\boldsymbol{I})_q \\ L(\boldsymbol{I})_x &= |\hat{\boldsymbol{x}} - \boldsymbol{x}|_\gamma \\ L(\boldsymbol{I})_q &= \left| \hat{\boldsymbol{q}} - \frac{\boldsymbol{q}}{|\boldsymbol{q}|} \right|_\gamma \end{aligned} \tag{4}$$

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 Eq. (4) 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 [33], 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 Eq. (4).

Kendall and Cipolla [15] showed that the weighting parameter β in the loss function can be replaced with trainable parameters by introducing the concept of homoscedastic uncertainty [16]. By introducing additional scalar parameters $\hat{s}_x, \hat{s}_q$, the loss function can be replaced by the following function:

$$L_s(\boldsymbol{I}) = L(\boldsymbol{I})_x \exp(-\hat{s}_x) + \hat{s}_x + L(\boldsymbol{I})_q \exp(-\hat{s}_q) + \hat{s}_q \tag{5}$$

$\hat{s}_x, \hat{s}_q$ represents the task specific uncertainty, and these parameters will be learned from the training data. L1 norm is used for the loss function (5). Our approach can be applied in general CNN-based image localization approaches, so we used the approach that solves the loss function (5) because it is more accurate than other CNN-based image localization approaches.

3.2.2 Radio-Wave Localization Network

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

Before inputting RSSI values to our network, we pre-process BLE signals in the similar way to our SfM approach discussed in Section 3.1.3. 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.

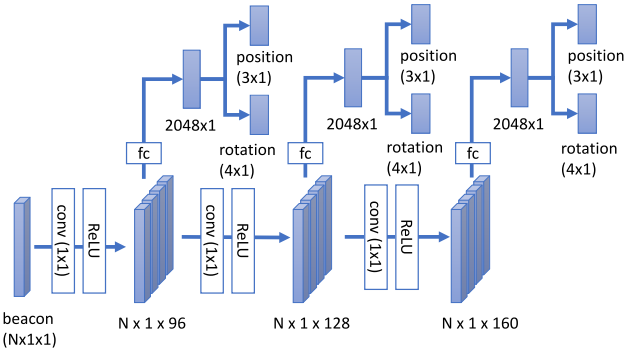

Fig. 2 Architecture of a network to process BLE signals. conv(1×1) represents a convolution layer. ReLU represents a rectified linear unit layer. fc represents a fully connected layer.

The network architecture for beacon signal is shown in **Fig. 2**. If the environment has N beacons, the input data for the network will be a $N \times 1 \times 1$ tensor. To combine a radio-wave network with an image network, we used an architecture similar to PoseNet. The architecture is composed of three sub-networks, each of which outputs a three dimensional position vector $\boldsymbol{x}$ and a four dimensional orientation vector $\boldsymbol{q}$. Each sub-network has one 1×1 convolution layer and a ReLU activation unit. The 1×1 convolution layer was originally proposed in Network-in-Network architecture [20] and helps to increase the accuracy by increasing the depth. GoogLeNet architecture [33] also uses 1×1 convolution layers to extract more features. Each output layer is connected with fully connected layers having 2,048 nodes.

Note that our approach does not assume any prior knowledge about environments, such as where radio-wave transmitters are located. It only assumes that all of the IDs of BLE devices installed in the environment are known. All of the IDs of BLE devices can easily be collected by scanning the BLE signals in the environment. This step can be done at the same time as collecting training data and requires no additional workload. Therefore, our approach is easy to apply in an environment where BLE beacons are already installed.

Because our radio-wave network can directly regress 6-DOF poses from radio-wave signals, the network architecture in Fig. 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.

3.2.3 Radio-Visual Localization Network

For inputting both image information and radio wave information, we combined PoseNet [17] and a radio-wave network as a dual-stream network. **Figure 3** shows the overall architecture. 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 Ref. [17].

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 ro-

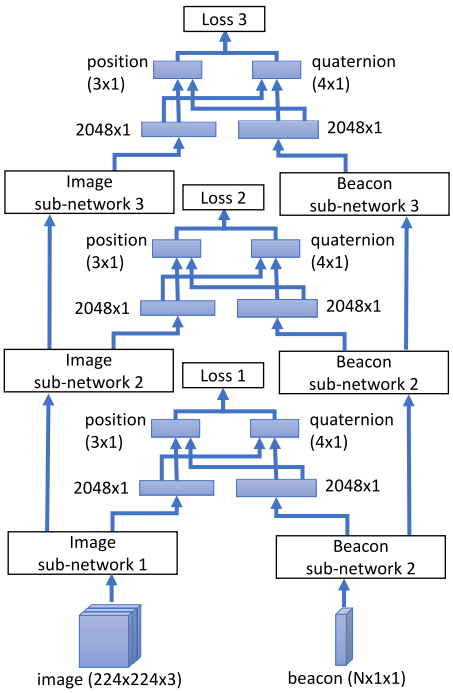

Fig. 3 Overall architecture of the proposed dual-stream network.

tation 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 Eq. (5). Following GoogLeNet [33], 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.

3.2.4 Beacon Data Augmentation

In image classification, data augmentation is often used to create additional training data from original training data [18]. 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 augmented data by changing the values of observed beacons only. To simulate BLE signals weakened by the interference, we randomly changed observed BLE signals to smaller values. When augmenting each data, we first randomly selected 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 beacons, the rate to weaken the RSSI signal was sampled by uniform random variables from 0 to 1. For each training data, we created 5 augmented data.

4. Experiments

4.1 Image and BLE Data Collection

For our proposed SfM and CNN approaches, we need images and BLE signals labeled with 6-DOF poses. Especially for the deep CNN approach, a large amount of training data is needed to improve the accuracy of results. Because manually labeling 6-DOF poses for a large set of images is practically impossible, we instead used a LiDAR to create ground truth 6-DOF poses. A LiDAR can generally achieve centimeter-level localization accuracy [13]. We used Velodyne VLP-16 for LiDAR. To associate positions estimated by a LiDAR and images, a LiDAR and a smartphone were connected to a tripod at fixed locations. At the same time as the LiDAR point cloud was recorded, images and BLE signals were 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 were calculated offline by using the LiDAR SLAM algorithm [13] that is based on the NDT algorithm for 3D point cloud matching [22]. To collect training and test datasets for different conditions, multiple recordings must be made of the same environment. Because the coordinates of the 3D map created by SLAM are 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 was then calculated by transforming the 6-DOF poses in the 3D map to this registered map.

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.

	Area size	# Beacons	# Training	# Test
D1	62 m × 60 m	90	2,297	741
D2	40 m × 32 m	59	2,542	828
D3	58 m × 60 m	55	4,147	1,350
D4	31 m × 31 m	92	2,490	766
D5	42 m × 56 m	91	2,578	867
D6	40 m × 50 m	112	2,491	833
D7	57 m × 40 m	66	4,484	1,442
D8	48 m × 45 m	90	3,699	1,091

4.2 Datasets

By following the data collection steps described in the section 4.1, we collected several large scale indoor datasets. Images were captured at the resolution of 1,280 × 720, and undistorted before the training and test. An iPhone7 was used for collecting data. For recording both training and test data, images were captured at 2 fps. Bluetooth signals were captured at 1 Hz by the iPhone (1 Hz is a fixed setting of the iPhone). All images are associated with Bluetooth signals that were recorded at the closest timestamp.

We created datasets for eight different locations. The area size, number of beacons, and number of video frames for each location are shown in **Table 1**. In each environment, BLE beacons were positioned every 4–6 meters. The locations of the BLE beacons were decided based on a previous study in order to balance the localization accuracy and the deployment cost [3]. **Figure 4** shows maps of tested environments. Solid lines show the positions of test videos. In all environments, we recorded four videos of the same path to keep training and testing separate from each other. Three of the videos were used for training and one was used for testing. We recorded facing in two opposite directions

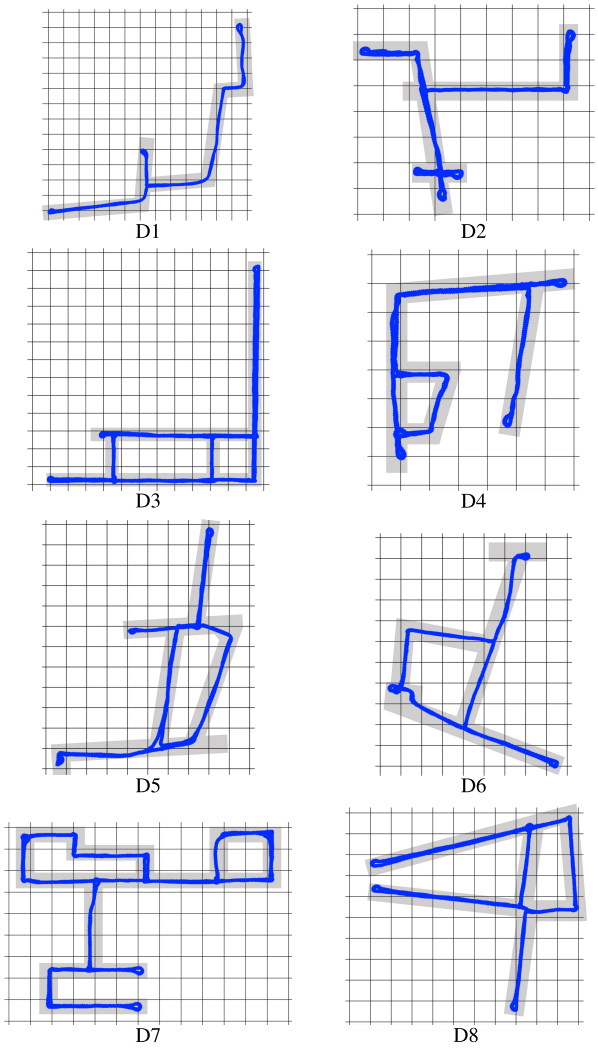

Fig. 4 Indoor maps of tested environments. Grid lines are drawn for 5 m^2 areas. Gray areas are corridors. Lines show groundtruth positions of test videos. For each environment, three training videos are recorded separately along the same routes.

on all paths. As shown in these maps, we recorded in different size areas and on different path shapes. **Figure 5** shows examples of images for all locations.

4.3 Baselines

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

- SfM BoW: SfM-based localization that uses BoW (bag-of-words) for image retrieval [25] to accelerate keypoint matching.
- SfM BLE: SfM-based localization that uses BLE signals to accelerate keypoint matching as proposed in Section 3.1.
- PoseNet β: PoseNet trained using the L2 norm loss function in Eq. (4), as proposed in Ref. [17].
- PoseNet σ: PoseNet trained using the L1 norm loss function in Eq. (5), as proposed in Ref. [15].
- DCNN BLE: CNN-based localization that regresses 6-DOF poses from image and BLE signals as proposed in Section 3.2.

In an SfM-based localization, keypoint matching requires a large computational cost. The baseline "SfM BoW" matched keypoints in a query image with keypoints in a 3D model that were extracted only from visually similar images. This baseline corresponds to skipping the step (2) of Fig. 1. Other steps were the same as our proposed "SfM BLE". "SfM BLE" uses BLE signals to search for candidate matching images in addition to BoW as discussed in Section 3.1. "SfM BoW" and "SfM BLE" require 3D models for localization. For both approaches, 3D models were reconstructed by following the process discussed in Section 3.1.1.

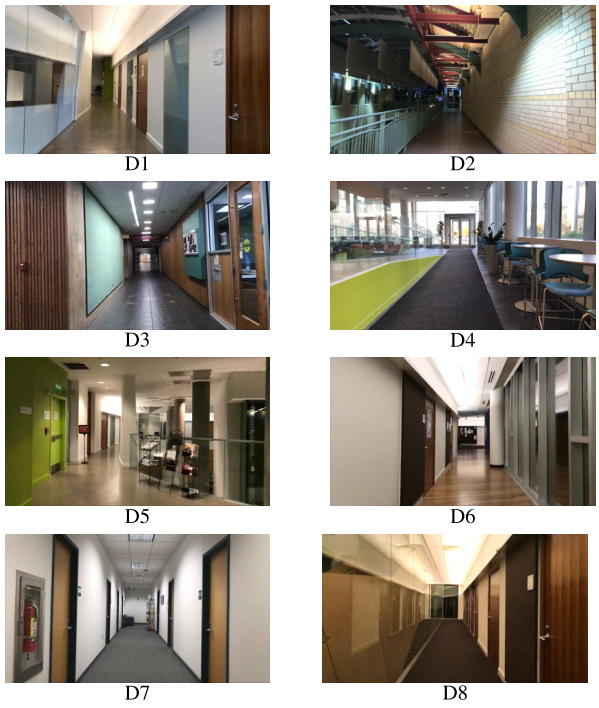

Fig. 5 Example image for each location.

For "SfM BoW", we selected 200 candidate matching images by using BoW image retrieval. For "SfM BLE", we first selected 400 candidate matching images by using BLE signals and then reduced the number of candidate images to 200 by using BoW image retrieval. When selecting candidate matching images by using BLE signals, the threshold of beacon co-occurrence (Eq. (3)) was set to 0.1. For both of these approaches, we used AKAZE [4] as a local feature detector and descriptors.

In all experiments of CNN-based localization, the network was trained by stochastic gradient descent using Adam solver. The learning rate was 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 was set as 500. For the baseline "PoseNet σ" and "DCNN BLE", we need initial values for s_x, s_q in Eq. (5). Following [15], we set these initial values as $s_x = 0.0$, $s_q = -3.0$. These CNN-based localization baselines and our proposed approach were implemented by TensorFlow [1].

As described in PoseNet [17], the network weight for CNN-based image localization was initialized by using the classification network trained by the Places database [39]. For initializing the network weight for BLE signals, we used random values for initial values.

4.4 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 Fig. 2, and only BLE signals as the input data. The loss function for this radio-wave CNN model was calculated by using the Eq. (5) in the same way as with 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 without using beacon data augmentation that was described in Section 3.2.4. "BLE Net (w/o Aug.)" shows the results. The results show that our CNN model could estimate a location only by one observation of BLE signals. The average localization error was 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, which show that beacon data augmentation improved the localization accuracy in general and reduced the localization error about 0.15 m at most. In following experimental results of "DCNN BLE", we used beacon data augmentation.

4.5 Evaluation of Proposed SfM and CNN Approaches

We then evaluated the accuracy of the two proposed approaches. **Table 3** compares CNN-based baselines and our proposed CNN-based approach. The results show average positional errors in meters, and average rotational errors in degrees. "PoseNet σ" estimated locations more accurately than "PoseNet β" because "PoseNet σ" learned the optimal weight for balancing positional errors and rotational errors from the training data. For all eight 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 0.4 m at most.

Table 4 shows 90 percentile localization errors for the "PoseNet σ" and our proposed approach and shows that our approach reduced the positional error at most about 0.7 m. As these results show, the proposed CNN-based approach consistently obtained better localization accuracy than the state-of-the-art approach. One limitation of our approach is that it has slightly worse rotational accuracy than baseline approaches. The difference is small (at most about 4 degrees), but it will be possible to use other baseline approaches only for the rotational estimation and to use our approach for the positional estimation for an application that requires an accurate rotational estimation. The additional computational cost for this will be very small because the CNN localization can process one image in less than 10 ms.

Table 5 compares the SfM-based baseline with our proposed SfM-based approach. We note that a CNN based approach can localize any images because it directly regresses 6-DOF camera poses, but an SfM-based approach cannot estimate a camera pose without first matching enough visual features of an input image with a 3D model. "Succ." in Table 5 shows the percentage of images which are successfully localized to all test images. "DCNN BLE, SfM BLE" in Table 5 show the results when "SfM BLE" was used at first and then "DCNN BLE" was used only for the image that "SfM BLE" could not localize. As for the SfM-based approaches, "SfM BLE" localized less images than "SfM BoW", but "SfM BLE" was more accurate than "SfM BoW" in general. "DCNN BLE" in Table 3 was significantly more robust for all datasets and even more accurate than SfM-based approaches for three datasets (D2, D3, and D7).

Figure 6 shows examples of typical images for which both "SfM BoW" and "SfM BLE" failed to localize. As in these examples, environments that lack sufficient visual features will be difficult to localize with SfM based approaches.

Figure 7 shows the cumulative localization errors for "SfM BLE," "PoseNet σ," and our proposed approach. "Ratio of im-

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 was not used and "BLE Net" shows the results when beacon data augmentation was used. "Pos. Error" shows positional errors, and "Rot. Error" shows rotational errors.

	BLE Net (w/o Aug.)		BLE Net	
	Pos. Error	Rot. Error	Pos. Error	Rot. Error
D1	**0.98** ± 0.7	72° ± 54	1.02 ± 0.8	68° ± 51
D2	1.09 ± 0.8	68° ± 53	**1.04** ± 0.7	69° ± 51
D3	**1.65** ± 4.7	69° ± 59	1.70 ± 4.6	71° ± 59
D4	1.16 ± 0.8	66° ± 53	**1.12** ± 0.8	65° ± 53
D5	1.38 ± 0.9	80° ± 55	**1.36** ± 1.0	82° ± 57
D6	1.03 ± 0.7	57° ± 48	**0.87** ± 0.6	56° ± 46
D7	1.33 ± 1.0	78° ± 53	**1.20** ± 0.9	79° ± 54
D8	1.74 ± 1.3	69° ± 53	**1.60** ± 1.2	67° ± 51

Table 4 90 percentile positional errors in meters and rotational errors in degrees.

	PoseNet σ		DCNN BLE	
	Pos. Error	Rot. Error	Pos. Error	Rot. Error
D1	2.21	5.3°	**1.62**	6.3°
D2	1.07	4.8°	**0.90**	5.1°
D3	1.26	8.2°	**1.13**	9.3°
D4	1.65	5.6°	**1.14**	6.5°
D5	1.91	6.2°	**1.32**	7.0°
D6	1.30	5.1°	**1.04**	7.4°
D7	0.86	15.1°	**0.82**	15.4°
D8	2.75	4.6°	**2.08**	7.5°

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

	PoseNet β [17]		PoseNet σ [15]		DCNN BLE	
	Pos. Error	Rot. Error	Pos. Error	Rot. Error	Pos. Error	Rot. Error
D1	1.18 ± 2.1	2.4° ± 4.4	1.11 ± 2.2	3.1° ± 6.2	**0.86** ± 1.8	4.1° ± 5.2
D2	0.95 ± 0.8	2.6° ± 6.5	0.69 ± 0.4	2.3° ± 3.3	**0.55** ± 0.3	2.8° ± 5.6
D3	1.19 ± 1.1	3.6° ± 4.0	0.71 ± 0.7	4.4° ± 7.9	**0.64** ± 0.6	5.9° ± 13.5
D4	0.97 ± 1.2	2.4° ± 3.7	0.84 ± 1.2	2.7° ± 4.4	**0.62** ± 0.8	5.0° ± 5.3
D5	1.10 ± 1.4	3.5° ± 12.5	0.87 ± 0.9	4.2° ± 13.6	**0.61** ± 0.8	5.3° ± 10.3
D6	1.06 ± 1.4	3.0° ± 5.3	0.73 ± 0.8	3.5° ± 8.3	**0.61** ± 0.8	5.0° ± 4.5
D7	3.81 ± 3.5	10.3° ± 12.3	0.48 ± 0.5	7.9° ± 6.8	**0.45** ± 0.4	8.1° ± 6.9
D8	1.40 ± 1.4	2.3° ± 3.2	1.31 ± 1.2	2.7° ± 5.3	**0.87** ± 0.9	6.6° ± 8.7

Table 5 Average and standard deviation of positional errors in meters and rotational errors in degrees for SfM-based baselines and our approaches. "DCNN BLE, SfM BLE" was evaluated by first using "SfM BLE" and then using our approach only for the images that "SfM BLE" could not localize. "Succ." shows the percentage of test frames that were localized.

	SfM BoW			SfM BLE			DCNN BLE, SfM BLE		
	Pos. Error	Rot. Error	Succ.	Pos. Error	Rot. Error	Succ.	Pos. Error	Rot. Error	Succ.
D1	0.39 ± 0.3	7.8° ± 10.7	85%	0.32 ± 0.3	6.6° ± 3.4	78%	0.65 ± 1.7	6.3° ± 5.1	**100%**
D2	0.58 ± 0.7	13.8° ± 25.6	91%	0.73 ± 1.0	6.0° ± 3.6	77%	0.69 ± 0.9	5.8° ± 6.4	**100%**
D3	1.04 ± 2.4	18.8° ± 32.5	84%	0.98 ± 2.3	17.1° ± 35.6	74%	0.91 ± 2.0	14.4° ± 31.6	**100%**
D4	0.34 ± 0.4	9.2° ± 9.9	91%	0.28 ± 0.4	7.5° ± 5.3	83%	0.33 ± 0.4	7.3° ± 6.9	**100%**
D5	0.52 ± 0.9	9.5° ± 16.5	81%	0.35 ± 0.3	9.3° ± 7.7	73%	0.47 ± 0.7	8.8° ± 11.5	**100%**
D6	0.46 ± 0.5	11.8° ± 5.3	89%	0.43 ± 0.3	11.9° ± 4.1	80%	0.49 ± 0.5	10.8° ± 5.0	**100%**
D7	1.53 ± 29.8	13.6° ± 17.3	87%	1.19 ± 4.4	13.1° ± 18.0	57%	0.87 ± 3.4	9.8° ± 14.5	**100%**
D8	0.55 ± 1.1	12.0° ± 14.7	89%	0.44 ± 0.97	13.8° ± 20.5	82%	0.58 ± 1.1	12.4° ± 18.9	**100%**

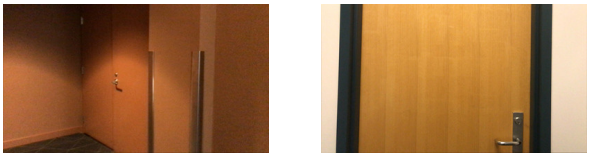

Fig. 6 Example of images which SfM based approaches failed to localize (Left: example from D4, Right: example from D7).

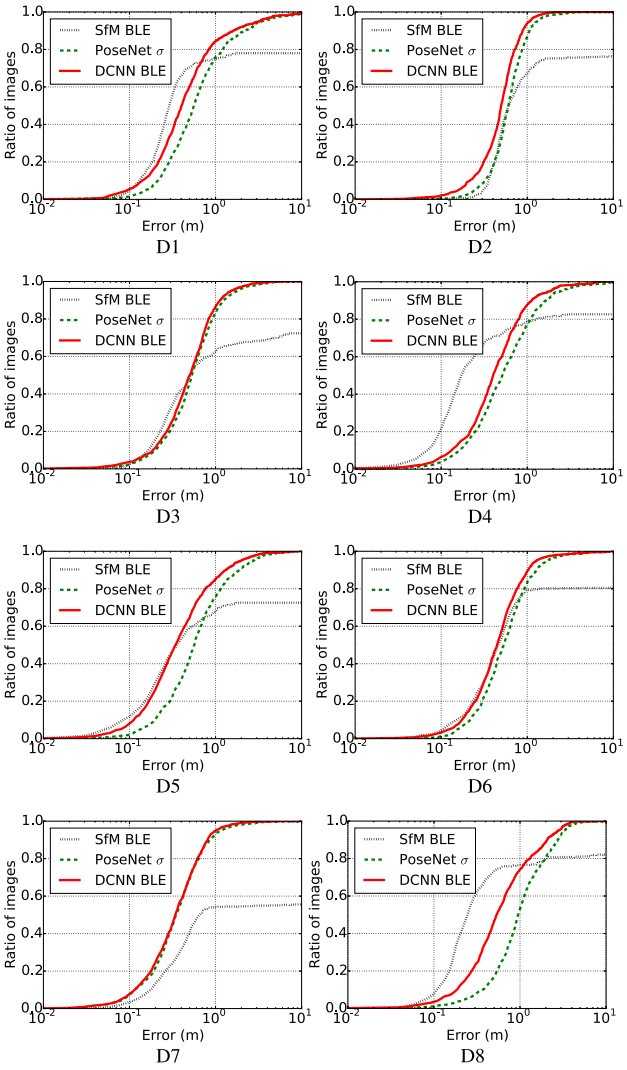

Fig. 7 Cumulative localization errors.

ages" shows the percentage of images to all tested images. The results show that our approach is consistently more accurate than "PoseNet σ." We note that the 80 percentile localization error of our approach is less than 1 meter except for dataset D8. As these results indicate, CNN-based approaches were significantly more robust than "SfM BLE," and our proposed "DCNN BLE" was more accurate than "PoseNet σ."

When SfM works well (i.e., in feature rich environments), "DCNN BLE, SfM BLE" can be the more accurate choice than "DCNN BLE" as shown in Table 5. However, automatically deciding which of the approaches will be more accurate is left as a future work. In summary, "DCNN BLE" can generally produce accurate localization results while maintaining robustness of CNN-based localization. This makes "DCNN BLE" a promising localization approach for navigation purposes in the real-world.

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

SfM BLE	PoseNet σ	BLE Net	DCNN BLE
0.587	0.008	0.004	0.008

4.6 Evaluation of Localization Speed

We evaluated the speed of localization for our datasets. CNN-based image localization is much faster than SfM-based localization, and the speed is not dependent on the area size. **Table 6** shows the average time to localize one image for all eight 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 CNN-based approach when inputting only BLE signals. "DCNN BLE" shows the result for our proposed CNN-based 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 "DCNN BLE" and "PoseNet σ" can localize an image in less than 10 ms and are much faster than "SfM BLE". For this reason, our approach and other CNN-based approaches will be suitable for real time applications.

4.7 Evaluation of Robustness to Environmental Changes

In actual navigation systems, some BLE beacons stop working due to battery exhaustion or device failure. Maintaining BLE beacons frequently needs large costs, so it is important to realize a localization system that is robust to BLE device failures.

We evaluated the robustness of our CNN-based approach in such a situation by randomly selecting BLE beacons and ignoring the signals of these beacons. To simulate a situation in which

Table 7 Average and standard deviation of positional errors in meters and rotational errors in degrees for our "DCNN BLE". Percentage values show the ratio of ignored BLE beacons in the testing phase.

	5%		10%		15%		20%	
	Pos. Error	Rot. Error	Pos. Error	Rot. Error	Pos. Error	Rot. Error	Pos. Error	Rot. Error
D1	0.89 ± 1.8	4.1° ± 5.2	0.89 ± 1.8	4.1° ± 5.2	0.90 ± 1.8	4.1° ± 5.2	0.96 ± 1.9	4.1° ± 5.2
D2	0.55 ± 0.3	2.9° ± 6.0	0.56 ± 0.3	2.9° ± 6.0	0.55 ± 0.3	2.9° ± 6.0	0.62 ± 0.4	2.9° ± 5.9
D3	0.65 ± 0.6	5.9° ± 13.5	0.66 ± 0.6	5.9° ± 13.5	0.68 ± 0.6	5.9° ± 13.5	0.67 ± 0.6	5.9° ± 13.5
D4	0.63 ± 0.8	5.0° ± 5.3	0.66 ± 0.8	4.9° ± 5.3	0.68 ± 0.8	5.0° ± 5.3	0.69 ± 0.8	4.9° ± 5.3
D5	0.61 ± 0.8	5.3° ± 10.3	0.62 ± 0.8	5.3° ± 10.3	0.63 ± 0.8	5.3° ± 10.3	0.65 ± 0.8	5.3° ± 10.3
D6	0.62 ± 0.8	5.0° ± 4.5	0.63 ± 0.8	4.9° ± 4.5	0.68 ± 0.8	4.9° ± 4.5	0.68 ± 0.8	4.9° ± 4.5
D7	0.45 ± 0.4	8.1° ± 6.9	0.44 ± 0.4	8.1° ± 7.0	0.49 ± 0.5	8.1° ± 7.0	0.51 ± 0.5	8.1° ± 7.0
D8	0.88 ± 0.9	6.6° ± 8.7	0.89 ± 0.9	6.5° ± 8.7	0.90 ± 0.9	6.5° ± 8.7	0.93 ± 0.9	6.6° ± 8.7

BLE devices fail after collecting training data, the selected BLE beacons were ignored only in the testing phase.

Table 7 shows the results. We evaluated by different percentages of BLE device failures. Each setting was repeatedly tested 10 times because the ignored BLE beacons were randomly selected. By increasing the percentage of BLE device failures, the accuracy become slightly worse. However, even when 20% of BLE beacons are removed from test data, the accuracy is still better than CNN-based baseline approaches except the dataset "D5". When less than 10% of BLE beacons are removed, the accuracy is better than CNN-based baseline approaches for all datasets. The results show that our proposed CNN-based approach is robust to BLE device failures.

We note here that we evaluated the effect of reducing BLE beacons only for the testing phase. This is more challenging than reducing BLE beacons both for training and testing phase. The results indicate that our approach is also robust to environments with fewer BLE beacons. The optimal number of BLE beacons depends on how they are placed in an environment. Investigation of how much we can reduce BLE beacons is left as a future work.

In addition to the changes of BLE signals, the visual appearance of environments will change over time in real-world applications. Although previous work showed that CNN based localization approach is robust to environmental changes [17], the accuracy will be worse if the environments undergo large visual changes. The trained model can be made more robust to visual changes by using training videos recorded in different conditions. We left the question of how many training videos will be needed for real-world applications as a future work.

5. Conclusion

We proposed two approaches to improve the accuracy of SfM-based image localization and 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 that the proposed SfM approach and CNN approach are more accurate than their baseline approaches. Our CNN-based approach reduced the average localization error about 0.4 m at most, and the 90 percentile localization error about 0.7 m at most compared to the state-of-the-art CNN-based image localization. Compared with SfM-based approaches, our CNN-based approach is significantly more robust and has close localization accuracy. In addition to its robustness, the CNN-based approach is superior to the SfM-based approach in computational speed and memory efficiency. The results also showed that our CNN-based approach is robust to BLE device failures.

In this work, we focused on localizing from an image and BLE signals. Our approach is not limited by BLE signals, and can be used with other radio-wave signals, such as Wi-Fi. 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. Our future works include applying our proposed localization approach to navigation systems in the real-world.

Acknowledgments This work was supported in part by Shimizu Corporation and NSF NRI grant (1637927).

References

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

[2] Agarwal, S., Furukawa, Y., Snavely, N., Simon, I., Curless, B., Seitz, S.M. and Szeliski, R.: Building rome in a day, *Comm. ACM*, Vol.54, No.10, pp.105–112 (2011).

[3] Ahmetovic, D., Gleason, C., Ruan, C., Kitani, K., Takagi, H. and Asakawa, C.: NavCog: A Navigational Cognitive Assistant for the Blind, *Int. Conf. Human-Computer Interaction with Mobile Devices and Services*, pp.90–99, ACM (2016).

[4] Alcantarilla, P.F., Nuevo, J. and Bartoli, A.: Fast Explicit Diffusion for Accelerated Features in Nonlinear Scale Spaces, *British Machine Vision Conf.* (*BMVC*) (2013).

[5] Alliance, S.C.: Bluetooth Low Energy (BLE) 101: A Technology Primer with Example Use Cases (2014).

[6] Brachmann, E., Krull, A., Nowozin, S., Shotton, J., Michel, F., Gumhold, S. and Rother, C.: DSAC: Differentiable RANSAC for Camera Localization, *IEEE Conf. Computer Vision and Pattern Recognition*, IEEE (2017).

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

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

[9] Girshick, R., Donahue, J., Darrell, T. and Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation, *IEEE Conf. Computer Vision and Pattern Recognition*, pp.580–587, IEEE (2014).

[10] He, S. and Chan, S.-H.G.: Wi-Fi fingerprint-based indoor positioning: Recent advances and comparisons, *IEEE Commun. Surveys & Tutorials*, Vol.18, No.1, pp.466–490 (2016).

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

[12] Higashi, K. and Arai, I.: Evaluation of Complementary Indoor Positioning System with Wi-Fi and Geomagnetic Fingerprinting (in Japanese), *Journal of Information Processing*, Vol.58, No.2, pp.384–395 (2017).

[13] Kato, S., Takeuchi, E., Ishiguro, Y., Ninomiya, Y., Takeda, K. and Hamada, T.: An open approach to autonomous vehicles, *IEEE Micro*,

Vol.35, No.6, pp.60–68 (2015).
[14] Kendall, A. and Cipolla, R.: Modelling uncertainty in deep learning for camera relocalization, *IEEE Int. Conf. Robotics and Automation*, pp.4762–4769, IEEE (2016).
[15] Kendall, A. and Cipolla, R.: Geometric loss functions for camera pose regression with deep learning, *IEEE Conf. Computer Vision and Pattern Recognition*, IEEE (2017).
[16] Kendall, A., Gal, Y. and Cipolla, R.: Multi-Task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics, arXiv preprint arXiv:1705.07115 (2017).
[17] Kendall, A., Grimes, M. and Cipolla, R.: PoseNet: A convolutional network for real-time 6-DOF camera relocalization, *IEEE Int. Conf. Computer Vision*, pp.2938–2946, IEEE (2015).
[18] Krizhevsky, A., Sutskever, I. and Hinton, G.E.: ImageNet classification with deep convolutional neural networks, *Advances in Neural Information Processing Systems*, pp.1097–1105 (2012).
[19] Li, R., Liu, Q., Gui, J., Gu, D. and Hu, H.: Indoor relocalization in challenging environments with dual-stream convolutional neural networks, *IEEE Trans. Automation Science and Engineering* (2017).
[20] Lin, M., Chen, Q. and Yan, S.: Network in network, arXiv preprint arXiv:1312.4400 (2013).
[21] Lowe, D.G.: Distinctive image features from scale-invariant keypoints, *Int. J. Computer Vision*, Vol.60, No.2, pp.91–110 (2004).
[22] Magnusson, M., Lilienthal, A. and Duckett, T.: Scan registration for autonomous mining vehicles using 3D-NDT, *Journal of Field Robotics*, Vol.24, No.10, pp.803–827 (2007).
[23] Murata, Y., Kaji, K., Hiroi, K., Kawaguchi, N., Kamiyama, T., Ohta, K. and Inamura, H.: Pedestrian Indoor Positioning Method Using Magnetic Data (in Japanese), *Journal of Information Processing*, Vol.58, No.1, pp.57–67 (2017).
[24] Nakamura, M., Akiyama, T., Sugimoto, M. and Hashizume, H.: Rapid and Precise Indoor 3D Localization Using Acoustic Signal for Smartphone (in Japanese), *Journal of Information Processing*, Vol.57, No.11, pp.2489–2500 (2016).
[25] Nister, D. and Stewenius, H.: Scalable recognition with a vocabulary tree, *IEEE Conf. Computer Vision and Pattern Recognition*, Vol.2, pp.2161–2168, IEEE (2006).
[26] Nowicki, M. and Wietrzykowski, J.: Low-effort place recognition with WiFi fingerprints using deep learning, *International Conference Automation*, pp.575–584, Springer (2017).
[27] Piasco, N., Sidibé, D., Demonceaux, C. and Gouet-Brunet, V.: A survey on Visual-Based Localization: On the benefit of heterogeneous data, *Pattern Recognition*, Vol.74, pp.90–109 (2018).
[28] Sano, H., Tsukamoto, M., Katagiri, M., Ikeda, D. and Ohta, K.: Improving Robustness against BLE Beacon Failures in Indoor Positioning System (in Japanese), *Journal of Information Processing*, Vol.58, No.5, pp.1138–1150 (2017).
[29] Sawada, K., Hanada, Y. and Mori, S.: User-installable Indoor Positioning System Using a Wi-Fi Beacon and PDR Module, *Journal of Information Processing*, Vol.24, No.6, pp.843–852 (2016).
[30] Schonberger, J.L., Berg, A.C. and Frahm, J.-M.: Paige: Pairwise image geometry encoding for improved efficiency in structure-from-motion, *IEEE Conf. Computer Vision and Pattern Recognition*, pp.1009–1018 (2015).
[31] Shah, R., Srivastava, V. and Narayanan, P.: Geometry-aware Feature Matching for Structure from Motion Applications, *IEEE Winter Conf. Applications of Computer Vision*, pp.278–285, IEEE (2015).
[32] Shotton, J., Glocker, B., Zach, C., Izadi, S., Criminisi, A. and Fitzgibbon, A.: Scene coordinate regression forests for camera relocalization in RGB-D images, *IEEE Conf. Computer Vision and Pattern Recognition*, pp.2930–2937, IEEE (2013).
[33] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V. and Rabinovich, A.: Going deeper with convolutions, *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pp.1–9 (2015).
[34] Taniuchi, D. and Maekawa, T.: Robust Indoor Positioning Method Based on Automatic Update of Wi-Fi Fingerprints (in Japanese), *Journal of Information Processing*, Vol.55, No.1, pp.280–288 (2014).
[35] Treuillet, S. and Royer, E.: Outdoor/indoor vision-based localization for blind pedestrian navigation assistance, *Int. J. Image and Graphics*, Vol.10, No.4, pp.481–496 (2010).
[36] Walch, F., Hazirbas, C., Leal-Taixé, L., Sattler, T., Hilsenbeck, S. and Cremers, D.: Image-Based Localization Using LSTMs for Structured Feature Correlation, *IEEE Conf. Computer Vision*, pp.627–637, IEEE (2017).
[37] Wang, S., Clark, R., Wen, H. and Trigoni, N.: DeepVO: Towards end-to-end visual odometry with deep recurrent convolutional neural networks, *IEEE Int. Conf. Robotics and Automation*, pp.2043–2050, IEEE (2017).
[38] Yang, Z., Wu, C., Zhou, Z., Zhang, X., Wang, X. and Liu, Y.: Mobility increases localizability: A survey on wireless indoor localization using inertial sensors, *ACM Computing Surveys* (*Csur*), Vol.47, No.3, p.54 (2015).
[39] Zhou, B., Lapedriza, A., Xiao, J., Torralba, A. and Oliva, A.: Learning deep features for scene recognition using places database, *Advances in Neural Information Processing Systems*, pp.487–495 (2014).

Tatsuya Ishihara is a Ph.D. candidate at the University of Tokyo. He received his BE and ME degree from the University of Tokyo. His primary research interest is in applying computer vision to create more accessible human interface. He is a member of the IPSJ, IEEE and ACM.

Kris M. Kitani is an assistant research professor in the Robotics Institute at Carnegie Mellon University. He received his B.S. at the University of Southern California and his M.S. and Ph.D. at the University of Tokyo. His research projects span the areas of computer vision, machine learning and human computer interaction. In particular, his research interests lie at the intersection of first-person vision, human activity modeling and inverse reinforcement learning.

Chieko Asakawa is an IBM Fellow at IBM Research and an IBM Distinguished Service Professor at Carnegie Mellon University. She received a B.A. degree in English literature from Otemon Gakuin University, and a Ph.D in Engineering from the University of Tokyo. Her research interests include accessibility research.

Michitaka Hirose is a professor of human interface and systems engineering in the Graduate School of Information Science and Technology, University of Tokyo. He received B.E., M.E., Ph.D. in Mechanical Engineering from the University of Tokyo. His research interests cover Multimedia, Human Interface and Virtual Reality.