A Neural Network Based Place Recognition Technique for a Crowded Indoor Environment

Asok Perera, Ravindra Ranasinghe and Gamini Dissanayake

***Abstract*— Place recognition in a crowded and cluttered environment is a challenging task due to its dynamic characteristics such as moving obstacles, varying lighting conditions and occlusions. This work presents a robust place recognition technique that could be applied into a similar environment, by combining well known Bag of Words technique with a feedforward neural network. The feedforward neural network we use have three layers with a single hidden layer and it relies on rectifier and softmax activation functions. We employ cross entropy function to model the cost of our neural network and utilize Adam algorithm for minimizing this cost at the training phase. The output layer with softmax activation in the neural network, produces a vector of probabilities which represent the likelihood of test image being captured from a given region. These values are further improved by incorporating a transition matrix which is based on the building layout. We have evaluated our neural network based place recognition technique with data collected from a crowded indoor shopping mall and promising results have been observed by this approach. We also have analyzed the behavior of neural network for changes in hyper-parameters and presented the results.**

I. Introduction

Place recognition is a prominent requirement for applications such as location based context-aware applications, mobile robot navigation and people monitoring where the location of a person or an equipment (commonly termed as the subject) is an important part of the task context. Challenges related with outdoor localization have been addressed by Global Positioning System (GPS), which is capable of providing location coordinates with a reasonable accuracy for many practical applications related to localization and navigation. Even though GPS performs reasonably well in outdoor environments, its effectiveness reduces drastically in indoors, mainly due to the low reception of GPS signals inside buildings and lack of floor identification capability as GPS is incapable of tracking elevation information [1][2]. Many works have been done in order to overcome these limitations using different sensors such as laser range finders [3], ultrasonic sensors [4] and wheel encoders (Odometry based localization) [5] to derive subject's location within indoor environments. Researchers have also come up with signal strength based indoor localization methods utilizing Wi-Fi sensors [6] and Zigbee modules [7]. Stereo vision sensors [8] and RGB-D sensors [9] are also very popular methods for indoor localization.

In general, computer vision based localization techniques use a camera image to recognize the user location. The complexity of computer vision based localizing approaches increase when an indoor crowded environment such as a shopping mall is considered for estimating the position of a user. Factors such as occlusions due continuous motion of large crowd, varying lighting conditions and dynamic shopfront setups increase the complexity of vision based indoor localization. The computer vision based indoor localization technique presented in this paper attempts to identify the regions in a shopping mall subjected to all of the aforementioned challenging environmental conditions (Fig. 1 and Fig. 2).

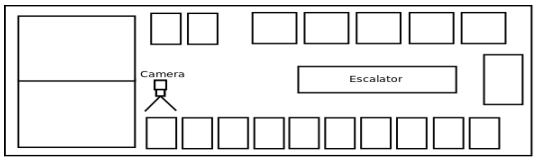

Fig. 1. Top view of Broadway shopping mall

Fig. 2. Sample images with crowd in motion

The proposed place recognition technique which relies on neural network based classification, could be integrated into applications such as autonomous robot navigation, human user localization (when they are assisted with camera mounted walking platforms), etc. The approach this work adopts for place recognition in a crowded shopping mall exploits strengths of Bag of Words (BoW) technique and feedforward neural networks. BoW technique that was commonly used for text categorization [10][11] has been later adopted by computer vision based applications as an image descriptor, to represent the content of an image. These BoW descriptors are predominantly used for image based classifications by applying various machine learning techniques. We have incorporated a feedforward neural network with a single hidden layer, for classifying the BoW descriptors derived from the captured images. The output layer of neural network produces a probability vector, where the value of each element represents the likelihoods of corresponding input image being captured from a particular region. As our primary goal is identifying regions where a subject traverses in a crowded indoor environment, we exploit the prior knowledge of building layout in the shopping mall in order to enhance the accuracy of place recognition task. Even though investigating sensor fusion techniques for improving the accuracy of location estimate is out of scope of this work,

Centre for Autonomous Systems, University of Technology, Sydney
KuranageAsokAravinda.Perera@student.uts.edu.au

the final probability vector generated by our approach could be combined with location estimates from different sensors in order to enhance the accuracy of localization.

This paper is organized as follows: Section 2 describes the related works and in Section 3 we discuss the core concepts of our algorithm. Section 4 explains the setup and experimental results while Section 5 concludes the paper.

II. Related Work

Image based place recognition has been experimented in many ways within the research community, especially based on the surrounding where the algorithm is applied. McManus et al. [12] proposed a method for learning place-dependent features for long-term vision-based localization which is specifically targeted for outdoor place recognition in varying weather conditions. In this work, mid level patches are chosen within images in contrast to low level features such that these patches are stable regardless of the appearance conditions of the environment. An outdoor and indoor place recognition approach presented by Sunderhauf et al. [13], employs a reliable landmark proposal method together with strengths in Convolutional Neural Networks(CNN). This work relies on a CNN that was already being trained on 'ImageNet', a generic image database, hence no application specific offline training is needed. As this work depends on landmark regions in an image for describing a scene in contrast to relying on the whole image, it demonstrates an improved robustness against view point changes or partial occlusions.

A visual indoor localization method proposed by Piciarelli [14], detects SURF features [15] and stores them together with manually defined positional information. While this is performed in an offline phase, the localization is done in an online phase where visual features from each image are compared with stored data for identifying the best matching reference image. This algorithm employs a standard KLT point tracker [16] to track feature positions in image frames and RANSAC algorithm [17] is utilized to exclude outliers in the process of estimating projective transformation that best describes the displacement of features. However this work does not specifically assume a crowded indoor environment for localizing the subject where many occlusions and other feature inconsistencies may present for a given region. In addition to feature based recognition, descriptors have also been used for place recognition task. Work done by Sahdev and Tsotsos employs Histogram of Oriented Uniform Patterns (HOUP) descriptors and utilizes Support Vector Machine (SVM) classifier for place categorization. Another descriptor based place recognition approach is proposed by Sizikova et al. where CNNs was used for generating descriptors [18]. In order to derive depth descriptors and intensity descriptors, it assumes that the input images to be RGB-D images. Once these two types of descriptors are generated, they are combined for deriving a joint descriptor which represents depth and intensity properties of the scene. These joint descriptors are then matched for estimating the most suitable region for a test RGB-D image.

Another potential approach for vision based place recognition, especially in indoor environments such as shopping malls is to recognize logos/labels and similar identical symbols in shop fronts. There are several sound research activities for recognizing logos in real world, such as methods proposed by Romberg et al. [19][20], DeepLogo [21] by Iandola et al., etc. In work [19], a quantized representation of the regions in logos is derived based on the analysis of local features and the composition of basic spatial structures like edges and triangles. Romberg and Lienhart have suggested another logo recognition technique using Bundle min-hashing approach [20]. Min-hashing is a technique for quickly estimating the similarity of two sets that is very commonly used in document matching applications such as web page comparison etc. In this work, Min-hashing technique together with visual words (corresponding to visual features) have been applied for the recognition task. In DeepLogo [21] work by Iandola et al., it presents three CNN architectures, where two of them are based on well known GoogLeNet CNN while the remaining one closely follows the GoogLeNet. Even though this work attempts to address logo recognition problem, it does not specifically consider logos in shopfronts, but rather focuses on recognizing logos from general setups such as vehicles, newspapers, outdoor environments, indoor environments, etc. (FlickrLogos-32 data set). Even though logo recognition seems like a potential method for our specific application, in practice it is unrealistic to expect robust logo only images in a localization problem. In our approach, rather than expecting logo only images, we assume that we can retrieve less dynamic and more representative images for each region by cropping and removing the lower part of the image where people and dynamic objects are mostly present.

A work done by Xu et al., performs feature fusion for shopfront recognition, in order to localize a person in a shopping mall [22]. This work incorporates style features and text features from an image, for identifying the user's position. In order to retrieve potential text features from the image, it employs a CNN based technique [23] and then executes a filtering method to reject false text candidates. Identifying style features is achieved by fine-tuning AlexNet CNN using the collected data set. Once the style features and text features are retrieved, these two types of features are fused to produce the final result for place recognition. The feature fusion is done by combining these two features into a new feature vector and training a classifier on the joint vectors. This work uses shop front images collected from Internet to evaluate its performance, where some of them may represent more controlled environments compared to an active and crowded shopping mall. Even though performance improvements have been observed by this approach, results indicate the presence of scenarios where feature fusion fails.

In the following section, we comprehensively discuss our neural network based approach for place recognition in a crowded shopping mall.

III. Neural Network Based Approach For Place Recognition In A Crowded Indoor Environment

A. Motivation

Place classification in a crowded environment especially in a place like shopping mall, poses many challenges due to its dynamic nature. Extensive crowd and motion of people, varying light conditions, varying shop front setups and posters are few examples for the aforementioned dynamic characteristics. In addition to these dynamic characteristics, it is possible for camera to capture images that include some parts of nearby regions. This may occur due to the change in orientation of camera or simply because the image is captured at a transitional stage between regions. Motion blur, due to the motion of platform which the camera is fixed as well as due to the fast moving crowd, might also increase the ambiguity of the estimated region. Therefore, having a proper method to valuate the accuracy of place recognition algorithm by incorporating a probability of the subject being in a particular region is important. In the proposed algorithm, a vector of probabilities is generated where each value represents the likelihood of subject being in the corresponding region. The accuracy of place recognition task could be enhanced by combining these outputs with various other location estimation methods based on different sensors, such as odometry, Wi-Fi based localization, etc., even though presenting such a sensor fusing technique is out of the scope of this work.

B. Method

This work incorporates well known Bag of Words model to generate image descriptors which are then fed into our neural network based classifier. The experiments we performed are based on images collected from a real crowded indoor shopping mall. We considered 20 different regions for training and testing our model and the images are collected from these regions. Then the collected images are cropped in order to remove the bottom portions(one third of the complete image is removed). The reason for cropping the bottom part of images is to remove sections with potentially dynamic characteristics such as moving crowd, varying advertisement boards, etc. The upper parts of the majority of images contain more persistent features coming from shop logos, fixed lights, roof patterns, etc. In parallel with the image cropping step, we perform manual classification of captured images by using an application we have developed, in order to label the training images. Once the classification is completed, we apply Bag of Words model to our images as illustrated in Fig. 3. For implementing BoW model, we extract Speeded-up robust features (SURF) [15] from all training images and cluster those collected features using K-Means algorithm. The purpose of this clustering step is to identify similar features in all images within the SURF space. Once the K-Means clustering is completed, the cluster centers are derived, such that each cluster center represents the features in that particular cluster. These cluster centers are selected as words in the BoW model and a vocabulary is generated

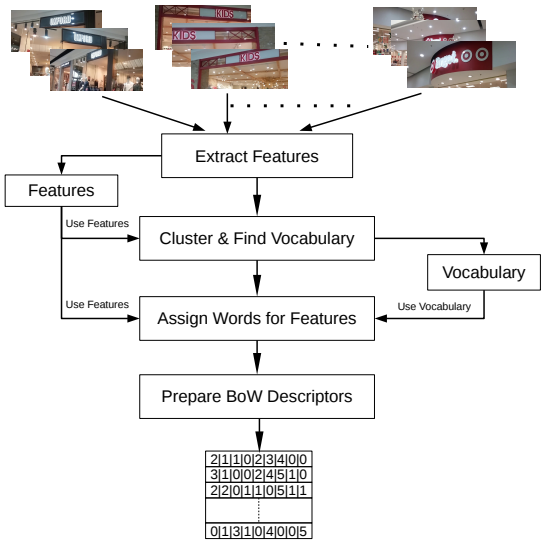

Fig. 3. Bag of Words Extraction

out of these words. Once the BoW vocabulary is created, all the detected features of training images are labeled as words from this vocabulary. This is known as the feature quantizing step in BoW model which reduces the scope of features by labeling them as representative words. In this way, the training images are now represented by a list of words instead of features. After this step, frequency of each word in each image is calculated and a list of word histograms is created for training images. As the final step of our BoW model, we normalize the word frequencies in these histograms. Each histogram from the list, which represents the normalized frequencies of words for a particular image can be treated as a BoW image descriptor.

Once the normalized BoW descriptors are generated from the training images, we feed them into the input layer of our feedforward neural network. This neural network consists of an input layer, a single hidden layer and an output layer (Fig. 4). The input layer has 1500 inputs where each input corresponds to a word in the BoW descriptor. The normalized BoW descriptor values are directly entered into the input layer of neural network. Each inputs in input layer is connected to all the neurons in hidden layer, while each hidden layer neuron is connected to all outputs in the output layer (fully connected layers). In our network, we have 150 neurons in the hidden layer with rectifier as the activation function (1).

$$\text{rectifier}(x) = \max(0, x) \tag{1}$$

Here x denotes the input to the rectifier function. According to the rectifier function, the hidden layer neurons produce 0 for each negative input value while it outputs input as it is for positive values. In addition to the rectifier function in hidden layer, we have employed softmax function as the activation function at the output layer (2).

$$\text{softmax}(x)_i = \frac{e^{x_i}}{\sum_{j \in L} e^{x_j}} \tag{2}$$

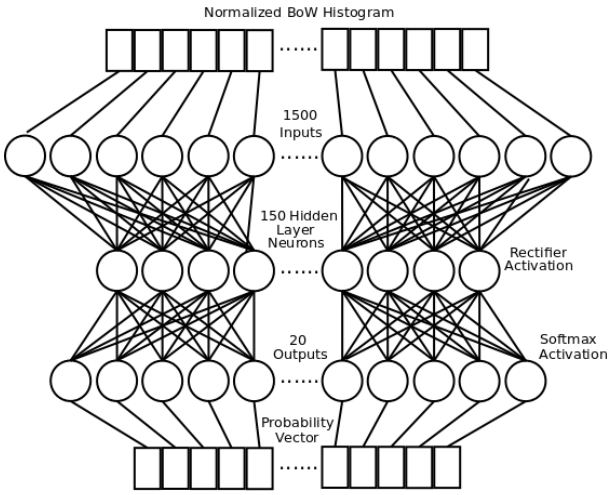

Fig. 4. Neural Network Architecture

Here x denotes the input to the softmax function while, L and i stand for the set of neurons and index in the output layer respectively. The purpose of using softmax function at the output layer of our network is to derive a probability vector which represents the likelihood of each class being the correct one. As the output layer of our neural network consists of 20 outputs, the softmax activation function generates a 20 element vector with each value representing the classification probability.

At the neural network training phase, we set the number of epochs to be 450 and batch size to be 15. We employ two loops for training the network; i.e., outer loops with 450 steps for epochs and inner loop with i steps where,

$$i = \frac{\text{number of training samples}}{\text{batch size}} \tag{3}$$

At each iteration of the inner loop we select 15 random training data rows from the loaded training data matrix to feed into the network. In order to receive persistent results we set a loop dependent seed value before calling the random function.

Our implementation adopts well known cross entropy function to model the cost of our network output at training iterations(4).

$$H(l,p) = -\sum_{x} l(x) \log p(x) \tag{4}$$

Here, $H(l,p)$ denotes the cross entropy between probability distributions l and p. In our scenario, distribution l is considered to be a binary vector which represents the correct label for a given training data row, while p is the output probability distribution of our neural network. In this perspective, the cross entropy provides a measure for difference between correct answer (class label) and the output of neural network for a particular BoW descriptor of a training image.

In order to minimize the cost value between true class label vector and estimated probability distribution, we used backpropagation in conjunction with Adam algorithm [24] at the training stage. Adam is an algorithm which is introduced recently that is capable of optimizing objective functions efficiently with less memory requirements. It is based on first and second order moments of the gradient and makes use of individual adaptive learning rates for different parameters.

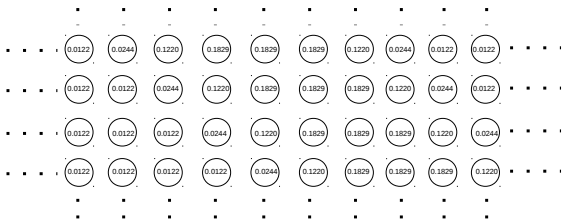

Fig. 5. Part of the transition matrix for shopping center floor map

Once we complete the training of our neural network, we test it using test data collected on a completely different day. In the testing phase, we extract BoW descriptors from test images using the same method we followed for training images. Once this is done, we feed these descriptors into our trained neural network and obtain the probability distribution from the output layer. In order to improve the accuracy of answer, a 20x20 transition matrix has been introduced to this implementation for representing the motion model of the moving subject (Fig. 5). Rows of this matrix represent current region while columns represent next region, and each cell has a probability which represents the likelihood of transiting from current region to next region. These probabilities depend on our prior knowledge on the layout of shopping mall. The final probability vector for test image is calculated by, multiplying the retrieved probability vector from output layer of our neural network with the appropriate row of this transition matrix.

IV. Experiments And Results

A. Data Collection

All our experiments are performed based on images collected from Broadway shopping mall in Sydney. In the training stage, video streams of 20 different shop fronts and similar regions were captured in different days and in different times. Reasons for capturing video streams are, collecting as many images as possible for training and ensuring the presence of motion blur in some images which can be experienced in a practical implementation. We used Galaxy 'Tab A' Android tablet to collect the video streams and made sure the data collection happened under varying lighting and shop front setups. In order to ensure these characteristics, we collected data before and after Christmas time, which introduced a considerable variety for the training images. Training video streams were collected in three separate days and once these are broken down into images, they are cropped in order to remove the bottom portions. Altogether, there were 128 images per region for training our neural network.

In order to test our technique, we collected a video stream in the same manner as in training data collection, but we performed the test data collection in a completely different day. The reason for this is, if we have either extracted

test images from the training images or collected even a completely separate test image set on one of the training data collection days, there exist a high possibility of test images displaying same environmental characteristics as in training images. Therefore, the higher results shown might be misleading in such a testing, as different days and different conditions could introduce various features into the images as well as remove some features that were not there before. This factor has been taken into consideration, specially when images are collected from a highly dynamic environment such as a shopping mall, hence we use test images from a completely different day for performing a fair valuation of our approach.

B. Setup

We used C++ OpenCV 2.4.8, tensorflow implementation in Python 2.7 and MATLAB 8.6.0.267246 (R2015b) on an Intel Core i5 machine with a memory of 8GB, for this implementation and data analysis.

We performed the manual classification of collected images by renaming JPEG file names. The reason for manual classification is to label the collected data in order to utilize them at the training and testing stages. Once the manual classification is completed, corresponding BoW descriptors are generated using an application developed in C++. These BoW descriptors are normalized and loaded into Python tensorflow program for training and testing the neural network model. We have used Matlab for various data analysis tasks in order to obtain a good understanding on the BoW descriptors related to training and testing images.

In order to develop our feedforward neural network, we employed tensorflow python library. In this implementation, we load the previously saved descriptors into the application for training/testing our neural network. We performed many experiments for understanding the neural network characteristics by adjusting parameters such as epoch count and batch size, and this task is elaborated in the next sub section. Our network is designed with softmax activation function at the output layer such that it produces a probability vector for image classification task. Once this probability vector is generated at the output layer, it is multiplied by a previously created transition matrix; i.e., Rows represent current region while columns represent next region. In this setup transition matrix is created such that the current region and nearest regions have higher transition probabilities while other regions have lower values. The assumption here is, a person cannot skip to a far away region suddenly without passing nearby regions. The final probability vector is retrieved after this multiplication and it demonstrates the likelihood of test image being captured from any given geographical region in the shopping mall. The region with maximum probability could be considered as the final answer for the place recognition task.

C. Characteristics of The Neural Network

Studying the neural network output accuracy by assigning different values to parameters such as epoch count, batch size, number of neurons in the hidden layer, etc., enables us to experimentally analyse the network characteristics. Applying the suitable activation function for network layers as well as using the most effective optimization algorithm are also key factors which effect the final result and efficiency of network.

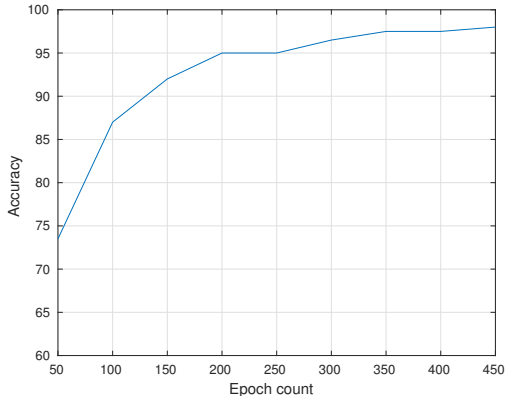

Fig. 6. Accuracy vs epoch count

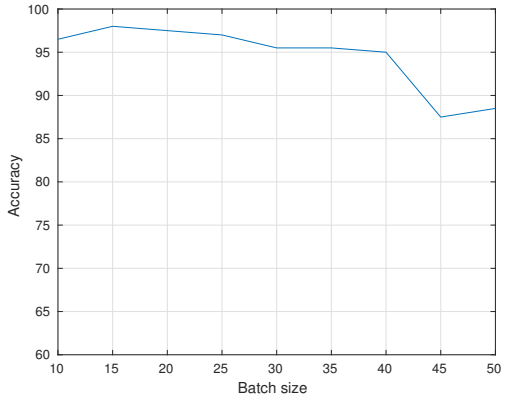

Fig. 7. Accuracy vs batch size

In our experiments, we could observe a clear trend in accuracy when number of training epochs are adjusted. As shown in Fig. 6, it is evident that the accuracy improves and reaches to a stable value when the number of epochs are increased. Throughout this experiment we kept the batch size at a constant value of 15. The change in accuracy with different batch sizes are shown in Fig. 7. Here the epoch count is kept at a constant value of 450. Increment in epoch count as well as reduction in batch size resulted in a higher training time as expected. However, we did not observe a significant change in testing time in any scenario.

We also could observe considerable accuracy variations with the change of hidden layer size. According to our results, the neural network produced an accuracy of 98.0% for 150 hidden layer neurons, 96.5% for 300 neurons and 77.5% for 450 neurons, when epoch count and batch size are kept at 450 and 15 respectively. We could observe a growth in training time with the increase of hidden layer neuron count. We tested our neural network with sigmoid function at the hidden layer instead of rectifier for evaluating its performance. We did not observe a significant change in

results when sigmoid activation is applied as it generated 97.5% of accuracy for the same test data.

It should be noted that all the experiments were performed with the presence of transition matrix.

D. Results

Accuracy of results in our approach before applying transition matrix was 96.0% and this has been improved up to 98.0% once the transition matrix is applied. For comparison purpose, we tested Bag of Words with Support Vector Machines for the same data as in our approach, and it produced an accuracy of 93.5% (Table I).

Our feedforward neural network took approximately 6.5 seconds to classify 200 normalized BoW descriptors from test images, hence nearly 32.5 milliseconds for a single image.

TABLE I

BoW-FNN accuracy comparison with SVM

Method	Accuracy
Normalized BoW with FNN	98.0%
Support Vector Machines	93.5%

V. Conclusion

This paper presents an image based place classification method for a crowded indoor environment utilizing a feedforward neural network as the core classification technique. We use Bag of Words method for generating image descriptors and feed them into our feedforward neural network as inputs. The network produces a vector of probabilities, where each element represents the likelihood of corresponding region being the correct region that the test image belongs. Moreover, the accuracy of place recognition task is enhanced by incorporating a motion model that is based on the building layout. Our approach is evaluated using data collected from Sydney Broadway shopping center.

As a future work, we aim to enhance our model by incorporating the capability to recognize and avoid processing images with high ambiguity caused by extreme occlusions and motion blur, in the context of localizing a moving subject. Furthermore, we anticipate to develop a localization framework, where it is capable of fusing the final classification probabilities with other sensory outputs. Our final goal is to design a robust localization platform, equipped with image based place recognition method and other sensor based localization techniques, which can provide reasonably accurate location estimates for a moving subject in a crowded indoor environment.

We thank Illawara Retirement Trust and Centre for Autonomous Systems in University of Technology, Sydney for support provided throughout this work.

References

[1] J. Torres-Sospedra, R. Montoliu, S. Trilles, . Belmonte, and J. Huerta, "Comprehensive analysis of distance and similarity measures for Wi-Fi fingerprinting indoor positioning systems," *Expert Systems with Applications*, vol. 42, pp. 9263–9278, 12 2015.

[2] E. Kaplan and C. Hegarty, *Understanding GPS: Principles and Applications*. Artech House, 2005.

[3] Z. Yan, X. Xiaodong, P. Xuejun, and W. Wei, "Mobile robot indoor navigation using laser range finder and monocular vision," in *Robotics, Intelligent Systems and Signal Processing, 2003. Proceedings. 2003 IEEE International Conference on*, vol. 1, pp. 77–82, 10 2003.

[4] K. Demirli and . Türkşen, "Sonar based mobile robot localization by using fuzzy triangulation," *Robotics and Autonomous Systems*, vol. 33, pp. 109–123, 11 2000.

[5] B.-S. Cho, W.-s. Moon, W.-J. Seo, and K.-R. Baek, "A dead reckoning localization system for mobile robots using inertial sensors and wheel revolution encoding," *Journal of Mechanical Science and Technology*, vol. 25, pp. 2907–2917, 11 2011.

[6] M. Kotaru, K. Joshi, D. Bharadia, and S. Katti, "SpotFi: Decimeter Level Localization Using WiFi," in *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication - SIGCOMM '15*, vol. 45, (New York, New York, USA), pp. 269–282, ACM Press, 8 2015.

[7] J. V. Marti, J. Sales, R. Marin, and E. Jimenez-Ruiz, "Localization of Mobile Sensors and Actuators for Intervention in Low-Visibility Conditions: The ZigBee Fingerprinting Approach," *International Journal of Distributed Sensor Networks*, vol. 2012, pp. 1–10, 8 2012.

[8] Y. M. Mustafah, A. W. Azman, and F. Akbar, "Indoor UAV Positioning Using Stereo Vision Sensor," *Procedia Engineering*, vol. 41, pp. 575–579, 2012.

[9] W. Winterhalter, F. Fleckenstein, B. Steder, L. Spinello, and W. Burgard, "Accurate indoor localization for RGB-D smartphones and tablets given 2D floor plans," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3138–3143, IEEE, 9 2015.

[10] S. George K and S. Joseph, "Text Classification by Augmenting Bag of Words (BOW) Representation with Co-occurrence Feature," *IOSR Journal of Computer Engineering*, vol. 16, pp. 34–38, 1 2014.

[11] Y. Yoshikawa, T. Iwata, and H. Sawada, "Latent Support Measure Machines for Bag-of-Words Data Classification," in *Advances in Neural Information Processing Systems*, pp. 1961–1969, 2014.

[12] C. McManus, B. Upcroft, and P. Newman, "Learning place-dependant features for long-term vision-based localisation," *Autonomous Robots*, vol. 39, pp. 363–387, 7 2015.

[13] N. Sunderhauf, S. Shirazi, A. Jacobson, F. Dayoub, E. Pepperell, B. Upcroft, and M. Milford, "Place recognition with ConvNet landmarks: Viewpoint-robust, condition-robust, training-free," *ARC Centre of Excellence for Robotic Vision; School of Electrical Engineering & Computer Science; Science & Engineering Faculty*, 2015.

[14] C. Piciarelli, "Visual Indoor Localization in Known Environments," *IEEE Signal Processing Letters*, vol. 23, pp. 1330–1334, 10 2016.

[15] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-Up Robust Features (SURF)," *Computer Vision and Image Understanding*, vol. 110, pp. 346–359, 6 2008.

[16] Jianbo Shi and Tomasi, "Good features to track," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition CVPR-94*, pp. 593–600, IEEE Comput. Soc. Press, 1994.

[17] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, pp. 381–395, 6 1981.

[18] E. Sizikova, V. K. Singh, B. Georgescu, M. Halber, K. Ma, and T. Chen, "Enhancing Place Recognition Using Joint Intensity - Depth Analysis and Synthetic Data.," in *ECCV Workshops (3)* (G. Hua and H. Jégou, eds.), vol. 9915 of *Lecture Notes in Computer Science*, pp. 901–908, 2016.

[19] S. Romberg, L. G. Pueyo, R. Lienhart, and R. van Zwol, "Scalable logo recognition in real-world images," in *Proceedings of the 1st ACM International Conference on Multimedia Retrieval - ICMR '11*, (New York, New York, USA), pp. 1–8, ACM Press, 4 2011.

[20] S. Romberg and R. Lienhart, "Bundle min-hashing for logo recognition," in *Proceedings of the 3rd ACM conference on International conference on multimedia retrieval - ICMR '13*, (New York, New York, USA), p. 113, ACM Press, 4 2013.

[21] F. N. Iandola, A. Shen, P. Gao, and K. Keutzer, "DeepLogo: Hitting Logo Recognition with the Deep Neural Network Hammer," 10 2015.

[22] Z. Xu, M. Pang, G. Zhou, and L. Fang, "Feature Fusion for Storefront Recognition and Indoor Navigation," 10 2016.

[23] M. Jaderberg, A. Vedaldi, and A. Zisserman, "Deep Features for Text Spotting," pp. 512–528, Springer International Publishing, 2014.

[24] D. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," 2014.