An Indoor Localization Method of Image Matching Based on Deep Learning

Guihua Yang[a], Yu Liang
College of Mechanical and Control Engineering, Guilin University of Technology, Guili, 541006, China.
[a]164557418@qq.com

Keywords: indoor localization, deep learning, location algorithm, image matching

Abstract. To overcome the problems of low accuracy and poor stability brought by the complexity of scenarios, an indoor localization method of image matching based on Deep Learning is proposed. The method includes taking images of indoor surroundings with cameras of mobile devices, setting up a dataset of images containing information on position and direction, and training a Convolutional Neural Network (CNN) with the image data. Then use the trained CNN to match the current images taken by the cameras of mobile devices to estimate precise location. The results of experiments show that the accuracy rate of CNN can reach up to 99.2%, positioning accuracy rate is up to 90%, and positioning precision is within 2 metres of diameter. This algorithm can achieve sound robustness, and fairly excellent generalization capabilities.

1. Introduction

With the ever-increasing application of technologies in daily life, people have higher demand for Location Based Services (LBS). Images, due to its attractive qualities such as its capacity for information storage, easy accessibility and easy extraction, are widely utilized on positioning tasks. Literature [1]-[5] propose traditional algorithms for feature extraction to deal with some specific issues. However, these algorithms are easily subject to blur, distortion, noise of images and other disadvantages owing to the complex indoor environment, resulting in poor performance on feature extraction in the light of generalization capabilities, robustness, stability of system and scalability.

Deep learning is a novel algorithm for feature extraction. With its deep multi-layer structure, the model can learn automatically to extract effective features from non-linearity. It has powerful self-learning abilities and efficient feature presentation. Recently, Deep learning has made great breakthroughs in computer vision [6] - [9]. Thanks to the excellent performance of Deep Neural Networks (CNNs) on the task of image processing, this paper introduces an indoor localization method of image matching based on deep learning. Compared with manually improved algorithms, the deep learning-based algorithm is capable of recognizing new features of a scenario, or features of new scenarios, and therefore is more robust against all sorts of disturbance, more capable of generalization and more stable. The proposed method is carried out by first setting up a dataset in which the image space corresponds with physical space, then using the model algorithm to match the extracted feature maps from the dataset with those of the corresponding physical space, and determining the position and direction of users, device according to the matching results. The image space refers to a series of images of an indoor scenario, while the physical space is represented by the geographic coordinates and direction.

2. Algorithm Model of Indoor Positioning Via Image Matching

This paper designs an algorithm for indoor localization that aims at estimating precisely the location and direction of the targeting mobile device. It is designed to overcome the complexity of indoor scenarios and achieve sound generalization capabilities, stability and precise positioning. A deep learning-based image matching algorithm is illustrated in Figure.1.

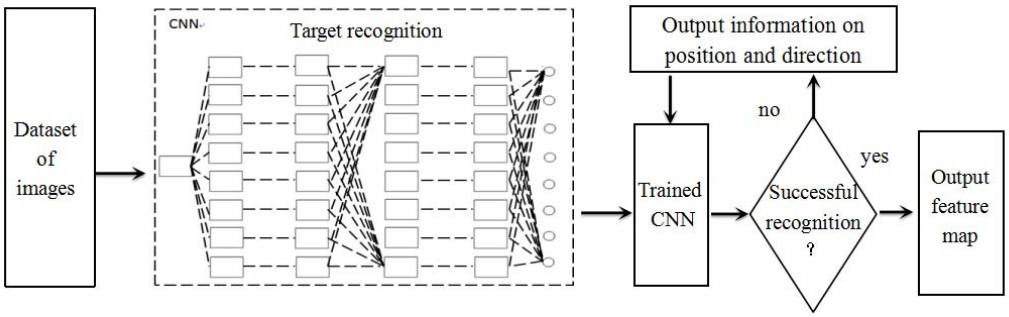

Figure. 1 The architecture of positioning algorithm of image matching

The steps to implement the algorithm are as follows:

Step 1: draw a structure chart and set up the geographic coordinates.

Step 2: choose several points of location inside the mall and take images of the surroundings from different angles at each point.

Step 3: label the images in accordance with the information of location and direction and form a dataset of images after classifying them into categories.

Step 4: obtain a neural network model of image matching by training the Convolutional Neural Network model with those images.

Step 5: the mobile device takes a current image of surroundings, and the network matches it with the images in the dataset.

Step 6: output the information concerning location and direction when the matching succeeded. If not, system goes back to Step 5.

2.1 Convolutional Neural Network Architecture

(1) Convolutional Neural Network uses convolution kernels to extract features, and by performing pooling operation, the downsampled data is then converted into a feature map. The Architecture usually consists of five layers: input, convolutional, activation, pooling and fully connected layers.

(2) The input layer takes images with 3 color channel or gray-level images with 1 channel as inputs, which are pre-processed through averaging, normalization, and PAC/whitening to improve computation speed and recognition.

(3) The subsequent layer is the convolutional layer. Convolution is an operation that gives the integral of the pointwise multiplication of functions such that it can better represent feature and denoise. The convolution expression is f=wx+b , where f is the output, x is the input signal, w is weight vector, and b is the bias term. The convolutional operation, as in Figure. 2, first adds pixel 0 at the margin of input image to intensify margin feature and denoise, then multiplies the local window matrix of input image by convolution filter matrix. The product is the output of that pixel. Small window matrix slide into input image with set stride, with each slippage forming a new pixel until all the pixels of input image convert into outputs. The sliding window is locally connected to the convolutional filter so that filters are shared in the same layer, reducing a large number of parameters.

The activation layer usually follows the convolutional layer, but it is not always necessary for the networks. The layer applies activation function to obtain non-linear mappings of the outputs generated in the convolutional layer for avoiding gradient loss.

(4) The pooling layer, as shown in Figure. 3, further extracts features of images from the previous layer, down samples feature map and the number of parameters, and thus reduces overfitting. There are two common pooling operations: max pooling and average pooling. The former operates with filters of size $n \times n$ applied with a stride of n, down sampling every depth slice in the input by n both width and height, while the later performs down-sampling by dividing the input into rectangular pooling regions and computing the average values of each region. So far some new CNNs have no pooling layer.

(5) The fully-connected layer is the last of CNNs, which works as a classifier. Input layers obtain output feature maps by implementing pooling, and the output layer outputs results, which are usually the highest scoring classifier predictions among the top 5 accuracy rates.

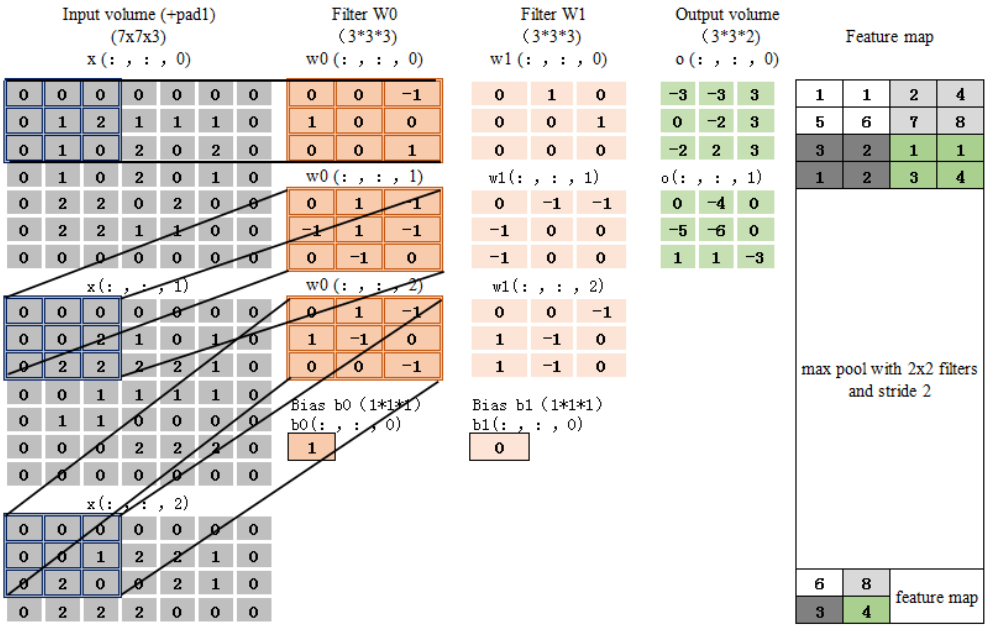

Figure. 2 Convolutional operation

Figure. 3 Pool operation

2.2 A Novel Deep Convolutional Neural Network Model

With the rapid development of both the computer software and hardware and deep CNNs, there is much better chance improving network performance. Traditional CNNs increase depth by adding more layers for better performance and results. However, it also increases complexity of the network and a large number of parameters, which makes the network easier to get overfitting and time-consuming. In this paper, we use GoogLeNet model, a novel deep CNN model, as shown in Figure.4.

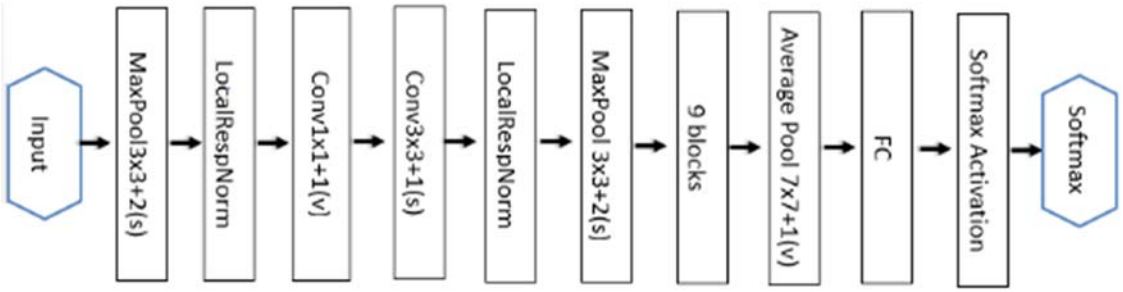

Figure. 4 GoogLeNet model

Contrary to traditional CNNs, the design of GoogleLeNet allows for increasing the width of the model by adding multiple sparse networks, whose block structure, as is shown in Figure. 5, uses smaller convolutional layers, making it more complex but with fewer parameters. The novel network has better nonlinear representation ability and faster computing speed.

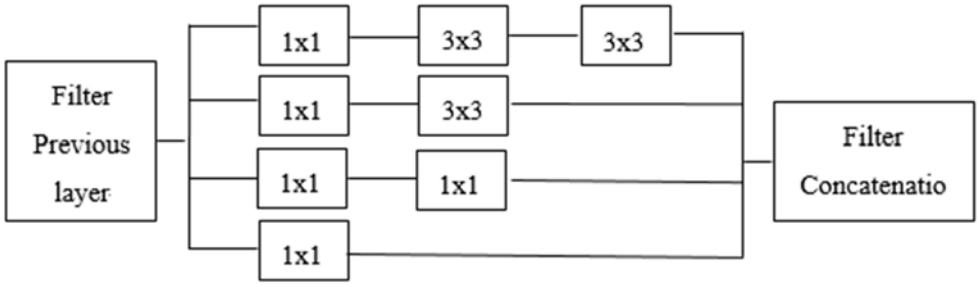

Figure. 5 Block structures

2.3 Learning Method of Deep CNN Googlenet Model

The training of deep CNN GoogLeNet model makes use of Stochastic Gradient Descent (SGD) to obtain optimal weights and bias value by achieving the minimum loss function. Take the sum of square errors of output node as the loss function, as in expression (1) (2)

$$E_d\left(w_{m,n},b;y^*_{ij},t_{ij}\right)=\frac{1}{2}\sum\left(t_{i,j}-y_{i,j}\right)^2 \tag{1}$$

Where E_d is the sample error,$w_{m,n}$ is the weight vector at location (m, n) of the filter, b is the bias term,y^*_{ij} is the output node at location(i, j), t_{ij} is the output ofy^*_{ij} , which corresponds to sample$d$. The activation function of neurons$f$ is function$Relu$（2）.

$$f(x)=\max(0,x) \tag{2}$$

$$y^*_{ij}=f\left(conv\left(w_{m,n},x_{ij}\right)+w_b\right) \tag{3}$$

$$w_{ij}\leftarrow w_{ij}-\mu\frac{\partial E_d}{\partial w_{ij}} \tag{4}$$

$$y^*_{ij}=\max\left(y^*_{11},y^*_{12}\cdots\cdots\cdots y^*_{ij}\right) \tag{5}$$

Where μ is the learning rate. Equation(3) calculates the convolutional node output, Equation(4) corrects weights and bias term, and Equation(5) optimizes the pooling operation.

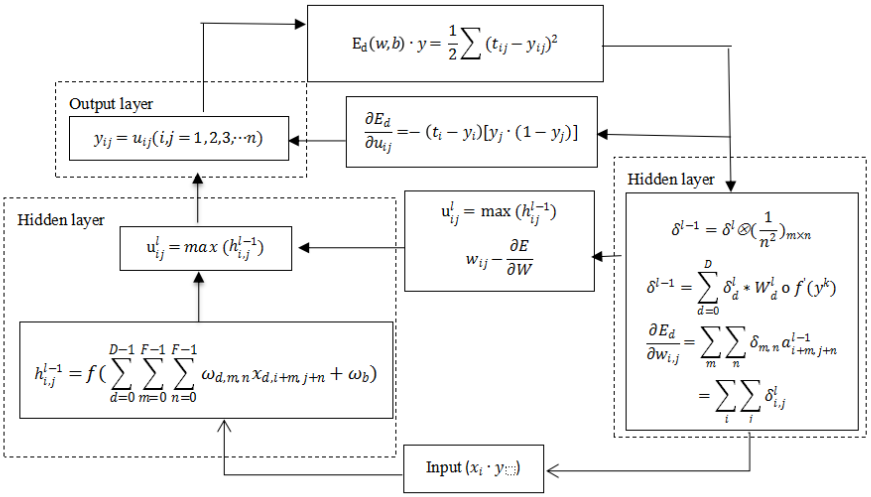

Figure. 6 Flow chart of model training

Figure. 6 presents the process of SGD based Error BackPropagation Algorithm training model. In the designed algorithm, GoogLeNet is a deep architecture consisting of the input layer, the convolutional layer (7x7+2(s)), the max pooling layer (7x7+1(v)), the fully connected layer and 3 activation function classification layers. Figure. 7 illustrates the curve of accuracy rate by operating 200 iterations during CNN training, and Figure. 8 shows the changes of loss function.

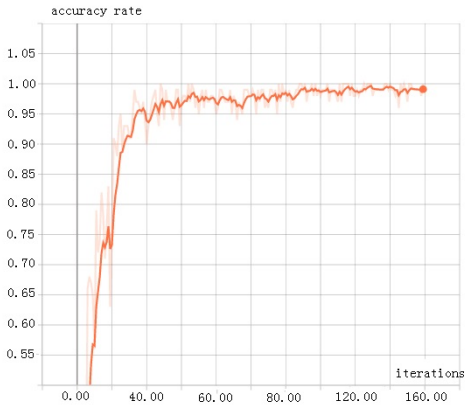

Figure. 7 Graph of accuracy rate

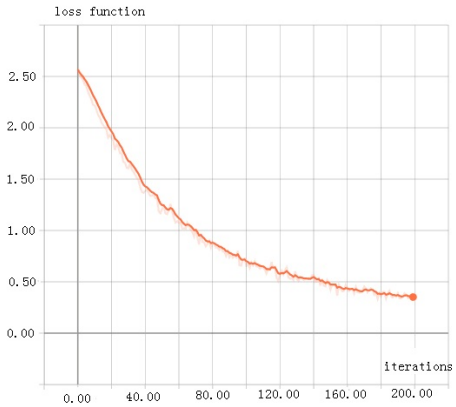

Figure. 8 Graph of loss function

3. Experiment and Analysis

3.1 Experiment

To validate the feasibility and effectiveness of the algorithm, we use images of a mall inside No.3 canteen of Guilin University of Technology for the implementation the localization method. The algorithm is implemented on Python. We first take 10 images of the indoor environment of the mall with the smartphone, Xiaomi 5C, and input the images into the computer. Then we run the algorithm to extract and match features from the testing images with those in the dataset. The system outputs the accuracy rate of location to determine position and direction. The implementation process is shown in Figure. 9.

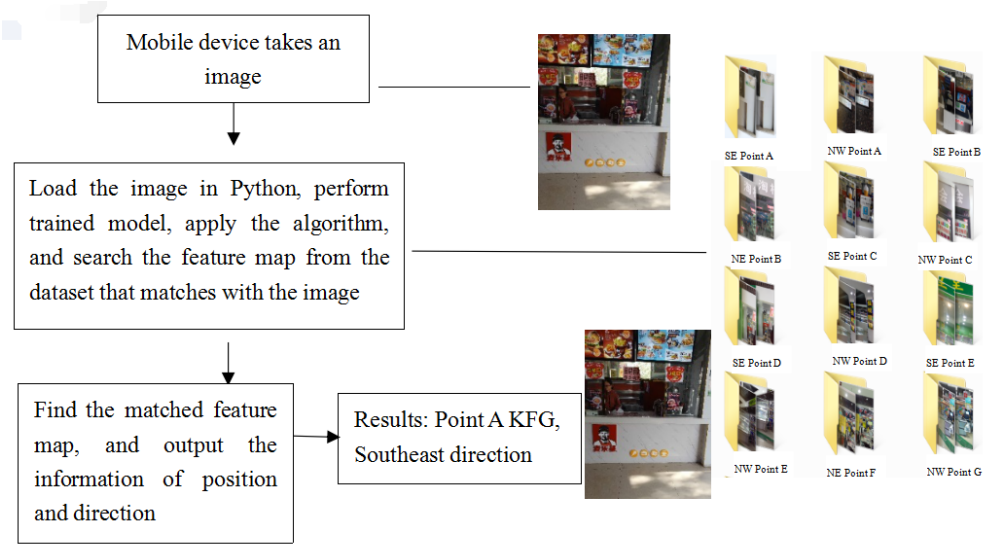

Figure. 9 Implementation of positioning

3.2 Experimental Results and Analysis

In the experiment, the algorithm achieves fast localization, high accuracy and good real-time response: with 200 iterations, the model obtains the testing accuracy rate up to 99.2%, localization accuracy rate exceeds 90%, and estimation error is within 2 metres of diameter, and the computing speed reaches up to 10 locations per 10 seconds. Moreover, the operation is simple, the algorithm is scalable, and various terminals are suited to take images. However, the position estimation can be somewhat affected by the brightness of color images. The testing results are shown in Table. 1.

Table. 1 The result of positioning test

No.	Testing images	Positioning results, top5,		Accuracy rate %
1	Apoint-1-se	KFG Point A	Southeast	95.55%
2	Apoint-2-nw	Milkshake shop Point A	Northwest	90.35%
3	Apoint-4-se	KFG Point A	Southeast	90.66%
4	Cpoint-1-se	Ice cream shop Point C	Southeast	90.33%
5	Dpoint-1-nw	Drug store Point D	Southeast	90.71%
6	Fpoint-1-ne	Market Exit Point F	Northeast	91.54%
7	Fpoint-2-ne	Market Exit Point F	Northeast	92.17%
8	Gpoint-1-nw	Market Exit Point G	Northwest	91.36%
9	Gpoint-2-sw	Fun Catch Point G	Southwest	90.46%
10	Epoint-2-se	Drug store Point E	Southeast	92.60%

4. Conclusion

The paper presents a positioning algorithm via image matching based on deep Convolutional Neural Network. It overcomes challenges brought by the complexity of indoor scenarios. Using the deep learning method, the algorithm is applicable to new features of a scenario or new scenarios. The experiment demonstrates the advantages of the algorithm, such as fast positioning, high accuracy, good generalization capabilities, and satisfying real-time response. Compared with localization using wireless signals, this method is more robust and stable. Moreover, thanks to the widespread application of smartphones, users can obtain the information of position and direction just by taking a picture of the surroundings with their smartphones, without deploying any other extra devices. The algorithm is scalable and can be applied to various scenarios, since the dataset can be updated instantly by users loading new data concerning the information of location, or by setting up a new dataset of images of new scenarios to achieve localization in new environment. The algorithm is also promising in its application to intelligent indoor navigation.

Acknowledgments

[Fund Program] Guangxi science and technology project (Number: 2015GXNSFAA139272); Bowen College of Management, GUT, University-level research project (Number: KY201701)

References

[1]. E. Wirbel, B. Steux, S. Bonnabel, and A. de La Fortelle. Humanoid Robot Navigation: From a Visual SLAM To a Visual Compass [C]. 2013 10th IEEE international conference on networking, sensing and control (ICNSC), Apr-2013:678-683.

[2]. N. Ravi, P. Shankar, A. Frankel, A. Elgammal, and L. Iftode, Indoor Localization Using Camera Phones [J]. Mobile Computing Systems and Applications, 2006: 1-7.

[3]. Liang J Z, Corso N, Turner E, et al. Image Based Localization in Indoor Environments[C] Computing for Geospatial Research and Application (COM.Geo), 2013 Fourth International Conference on. IEEE, 2013:70 -75.

[4]. Lowe D G. Distinctive Image Features from Scale-Invariant Keypoints[J]. International Journal of Computer Vision, 2004, 60(60):91 -110.

[5]. Calonder M, Lepetit V, Strecha C, et al. BRIEF: Binary Robust Independent Elementary Features [C] European Conference on Computer Vision. Springer-Verlag, 2010:778-792.

[6]. A. Krizhevsky, L. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In Proc.Neural Information Processing Systems, 2012.

[7]. Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied todocument recognition. Proceedings of the IEEE, 86:2278– 2324, 1998.

[8]. Le Cun Y, Boser B, Denker J S, et al. Backpropagation applied to handwritten zip code recognition [J]. Neural Computation, 1989, 1(4): 541-551.

[9]. Tian Yuan-dong, A Simple Analysis of AlphaGo [J], Acta Automatica Sinica, 2016, 42(5).