3ZD-07

Estimation of Arbitrary Locations by Using Infrared Sensor Array

Ryo Ota[†1,a)] Shoichi Ichimura[†1,b)] Qiangfu Zhao[†1,c)] Yong Liu[†1,d)]

Abstract: In this paper, we consider the possibility of estimating arbitrary locations of a resident by using infrared sensor array. The goal of our research is to build a smart home system that is smart enough to help the resident, but not too smart to observe the privacy of the resident. In order to achieve this purpose, we attempt to estimate the coordinate of the resident using a sparse infrared sensor array. Experimental results show that an array containing fifteen sensors is relatively good for estimating arbitrary locations for privacy protecting.

Keywords: Senior care, Sensor Array, PIR Sensor, Artificial Intelligence

1. Experimental Settings

1.1 Sensor Array

The sensor is an Arduino-Nano-compatible device (see Fig. 1). From the time stamps of the data we can see that the data rate is 60 Hz. That is, the sensor outputs 60 data per second. The sensors are connected to a local server (i.e. a computer for collecting and processing the data) via USB cables. The sensor is Passive Infrared Sensor (PIR sensor), which sends 1 if somebody is around the sensor and sends 0 if nobody is around the sensor or somebody is not moving. The cross signs in figure 3 illustrate the exact locations of fifteen sensors. The dot signs in Fig. 3 illustrate the sixteen human locations for training a neural network estimator. The locations are labeled as L0 to L15. Fig. 3 also shows an estimation result, which will be explained later.

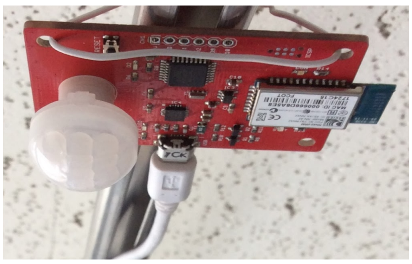

Fig. 1 Infrared Sensor

1.2 Data collection

Data collection was conducted in a room of our laboratory. The size of the room was about 4.0 x 8.5 m^2 and its height was about 2.7 m. The span of x direction was 50 cm and the span of y direction was 100 cm. In my experiment, the subject was one person, and the activity was fixed to "clapping". The duration of standing in each location was three minutes. Clapping was conducted by using metronome to get the stable data in 75 BPM. The direction of the subject was always the same.

2. Method

2.1 Moving Average

The Moving Average (MA) was used for normalizing and visualizing binary data from sensors in this study. The inputs of feature vectors were MA (Eq. (1)) of sensor data. A decision to use MA is that MA performed very well in our previous studies.

$$MA(n) = \frac{1}{W}\sum_{k=0}^{W-1} d(n-k) \tag{1}$$

W : Window size

$n : W+0, W+1, W+2, \cdots$

$d(n)$: Sensor signal, 0 or 1

2.2 Location Regression

In order to estimate arbitrary locations, regression analysis will be conducted between subject locations and sensors' signals. The regression will give us the opportunity to estimate xy-coordinates of the room from sensors' signals. Thus, this regression will allow us to estimate arbitrary locations. Figure 2 describes the structure of MLP Regression. The dimension of feature vectors is fifteen due to the number of sensors being fifteen.

In order to estimate arbitrary locations, it is ordinal to use xy-coordinates of the locations as outputs of machine learning model. For example, Global Positioning System (GPS) tells us longitude and latitude, which are the coordinates of the Earth. The number of subject is only one because we consider senior care of lonely elderly person in this study. This study is the first step to estimate arbitrary locations. The justification for choosing the MLP Regression is that MLP Classification has given us

[†1] Presently with University of Aizu, Aizu-Wakamatsu, Fukushima, Japan
[a)] s1220150@u-aizu.ac.jp
[b)] s1220066@u-aizu.ac.jp
[c)] qf-zhao@u-aizu.ac.jp
[d)] yliu@u-aizu.ac.jp

a great result in previous our studies. Thus, in the first step, using MLP Regression is a suitable choice to estimate locations by infrared sensors.

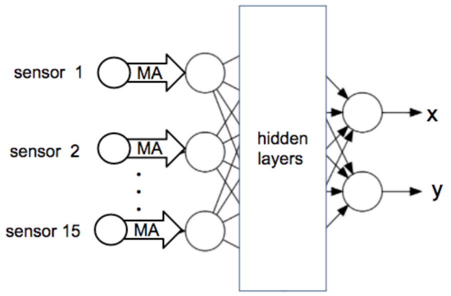

Fig. 2 MLP structure for xy-coordinate regression

3. Result

The result of subsection 3.1 used MLPRegressor in scikit-learn in Python where *max_iter* = 2000 and the number of hidden layers is one. All parameters were default values found in `http://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPRegressor.html` unless that was not described.

3.1 XY-coordinate Regression

XY-coordinate regression was conducted to estimate human xy-coordinates. The number of the output neurons was two (see Fig. 2). The test set contained eight locations - L0, L5, L10, L15, L3, L6, L9 and L12, and the other eight locations were used as training set. This meant that the trained model did not know about locations in the test set.

Fig. 3 visualized the predictions in some of the test sets. The window size was 300 and the number of hidden neurons was 20. Each stain-like pattern in the figures consisted of 1000 very small pixels which were sampled randomly from the predictions. Prediction of L10 in Fig. 3 is relatively good, however, prediction of L0 in x-coordinate shifted from the correct locations. Thus, the system of x-coordinate regression is worse than y-coordinate one.

Table 1 shows average absolute errors in case of different window size in 5-fold cross validation where the number of hidden neurons is 20. As seen from the table, the errors of x-coordinates tended to be larger than the errors of y-coordinates. However, there were no remarkable differences between the errors in different window sizes.

Table 2 shows average absolute errors in case of different numbers of hidden neurons in 5-fold cross validation where window size is 300. As seen from the prediction of L0 in Fig. 3, the result of xy-coordinate regression seemed to be overfitting. Thus, to decrease overfitting, the tables shows errors in different numbers of hidden neurons. Unfortunately, there were no remarkable differences between the errors in different numbers of hidden neurons. Thus, less hidden neurons did not contribute less error.

3.2 Discussion

The numbers of sensors in each direction might give negative impact to the result of xy-coordinate regression. The number of sensors in y direction was larger than x direction. Actually, y direction has five sensors, but x direction has three sensors. In addition, the spans between sensors in x direction were narrower than y direction. Thus, y direction is more informative than x direction. Therefore, if an experimental room is square-shaped and the numbers of sensors in both directions are the same, x-coordinate regression may work well like y-coordinate regression.

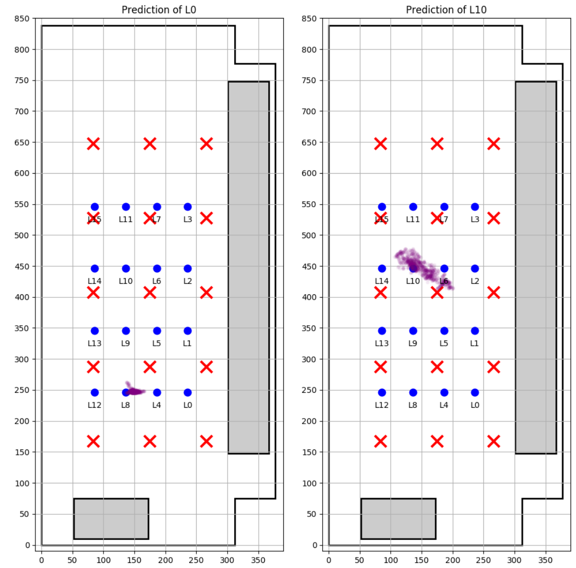

Fig. 3 XY-coordinate predictions of L0 and L10 in test set

Table 1 Average absolute errors of xy-coordinate regression in different window size [cm]

Window size	x-coordinate error	y-coordinate error
50	54.97	21.63
100	51.09	30.37
500	46.25	23.74
1000	47.03	23.88
1500	52.41	16.69
2000	47.21	29.43

Table 2 Average absolute errors of xy-coordinate regression in different # of hidden neurons [cm]

# of hidden neurons	x-coordinate error	y-coordinate error
1	50.67	20.57
5	58.07	19.84
10	57.93	26.89
15	55.30	16.38
20	43.05	29.56

4. Conclusion

This study sought to estimate human locations protecting residents' privacies using infrared sensor array. The location estimations in y-coordinates could be possible in relatively less errors. Thus, the sensor array system understands where human is roughly. In contrast, location estimations in x-coordinates were worse than estimations in y-coordinates. However, as mention in subsection 3.2, the system has a possibility to estimate x-coordinates if the numbers of sensors in both directions are enough, and the distances between sensors are also enough.