Accepted Manuscript

Adaptive K-nearest neighbour algorithm for wifi fingerprint positioning

Jongtaek Oh, Jisoo Kim

PII: S2405-9595(18)30050-X
DOI: https://doi.org/10.1016/j.icte.2018.04.004
Reference: ICTE 150

To appear in: *ICT Express*

Received date : 6 February 2018
Accepted date : 10 April 2018

Please cite this article as: J. Oh, J. Kim, Adaptive K-nearest neighbour algorithm for wifi fingerprint positioning, *ICT Express* (2018), https://doi.org/10.1016/j.icte.2018.04.004

ICT Express
Information & Communications Technology

Adaptive K-Nearest Neighbour Algorithm for WiFi Fingerprint Positioning

Jongtaek Oh*, Jisoo Kim

Department of Electronics, Hansung University, 116 16gil, Samsungyo-ro, Sungbuk-gu, Seoul 02876, Republic of Korea (e-mail: jtoh@hansung.ac.kr)

ABSTRACT

K-nearest neighbour is one of the most widely used algorithms for indoor positioning systems. However, the error for each estimated position notably varies depending on the *K* value used for the algorithm. Therefore, if *K* is a fixed value, the estimation error for the positions cannot be further reduced. In this Letter, I propose an algorithm that adapts the *K* value for each position by analysing the correlation between the *K* value and the received WiFi signal strength. The proposed algorithm provides an improvement above 30% on the positioning accuracy compared to the algorithm with fixed *K* value.

Index Terms: KNN; positioning; fingerprint

I. INTRODUCTION

Various indoor position estimation techniques have been proposed. Specifically, WiFi fingerprinting is widely used to estimate indoor coordinates of connected terminals with no additional infrastructure required [1, 2]. This method measures the received signal strength (RSS) of the WiFi routers around several reference points (RPs) within an area, and stores the measurements as WiFi fingerprint data. Then, the method estimates the terminal position at any test point (TP) in the area by measuring the corresponding RSS, which is compared against the stored WiFi fingerprint data. The position estimation is generally based on the Euclidean distance (ED) between compared data points. Hence, algorithms such as the k-nearest neighbour (KNN) can be used to estimate the TP position by considering the average of its closest *K* data points.

Likewise, more complex algorithms than the KNN have been used for WiFi fingerprinting, such as principal component analysis and support vector machines [3, 4]. However, the estimation accuracy of such algorithms is comparable to that of the KNN, which is relatively simple and hence more widely used. The estimation accuracy of the KNN algorithm can be further improved by using a weighted ED for each data point in the averaging process [5, 6]. On the other hand, algorithms for grouping closest data points have also been studied to improve KNN's position estimation performance. However, there is only minor performance improvement in Cluster filtered KNN [7] and Fuzzy C-means clustering [8], and these methods have a disadvantage that pre-processing time for RPs data is much consumed.

For all cases, the accuracy is susceptible to the *K* value selected for the KNN algorithm, and it varies according to the terminal position [9, 10]. Therefore, a fixed *K* value does not guarantee accurate estimation at every position.

In this letter, I propose an adaptive KNN algorithm to prevent the shortcomings of a fixed *K* value. The improved estimation accuracy is confirmed through experimental results.

II. K VALUE EFFECT AT DIFFERENT TPS

First, to analyse the applications of the KNN algorithm to WiFi fingerprint positioning in a real environment, I collected the RSS of WiFi routers signal around at 124 RPs in a 1 m interval using a smartphone application. After 3 hours, 9 TPs were measured and their estimation analysed. To minimize the interference from the human body, I installed the smartphone on a tripod pointing to the North.

In addition, I used (1) to compare the RSS similarity between the *i*-th RP and TP, where *N* is the number of WiFi routers measured at the TP, and RSS_j^r, RSS_j^t are the average RSS of WiFi router j at the *i*-th RP and TP, respectively. Moreover, I used the ED inverse as weight to estimate the TP position, $\hat{P}_t$, by averaging the coordinates of the *K* closest RPs, P_l, as expressed in (2), where ϵ is a small positive real number [5, 6].

$$ED_i = \frac{1}{N}\sqrt{\sum_{j=1}^{N}\left(RSS_j^r - RSS_j^t\right)^2} \tag{1}$$

$$\hat{P}_t = \left(\sum_{l=0}^{K}\frac{1}{ED_l+\epsilon}P_l\right)/\left(\sum_{l=0}^{K}\frac{1}{ED_l+\epsilon}\right) \tag{2}$$

Table 1 lists the position estimation error according to the *K* values for two TPs located at coordinates (6, 3) m and (15, 4) m with respect to a reference position. The table represents two typical estimation error patterns. For the TP at (6, 3) m, the position estimation error considering only the smallest ED (i.e., *K* = 1) is large, and it can be confirmed that the error reduces when more RPs are considered. Still, if *K* is above a specific value, the positions of distant RPs (i.e. large EDs) are also