The Application of Manhattan Tangent Distance in Outdoor Fingerprint Localization

Zhihe Li†, Xiaofeng Zhong†, Jie Wei‡, Han Shi†
†Dept. of Electronic Engineering, Tsinghua University, Beijing, China
‡ School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing, China
Email: {lizhihe16, shih14}@mails.tsinghua.edu.cn, zhongxf@mail.tsinghua.edu.cn, jwei@bjtu.edu.cn

***Abstract*—Received signal strength (RSS) is a typical measurement used for indoor and outdoor localization because of its low power consumption and no requirement of additional mode. Fingerprint localization is known to have higher accuracy than traditional methods and the most frequently used algorithm is *K Nearest Neighbors (KNN)*, where common metrics have some disadvantages. In this paper, exponential transformation and tangent distance will be utilized to improve accuracy of weighted-KNN based fingerprint localization, where Manhattan tangent distance (MTD) and approximate Manhattan tangent distance (AMTD) are proposed. In addition, data are collected by a field tester from three different outdoor area for test. Experiments demonstrate that MTD achieves the lowest root mean square error (RMSE), AMTD achieves the second lowest RMSE with much lower computation complexity compared to MTD and both of them outperform common metrics.**

***Index Terms*—Received signal strength, fingerprint localization, weighted-KNN, exponential transformation, Manhattan tangent distance, approximate Manhattan tangent distance.**

I. Introduction

Localization of smart devices has always been a hot research area because of it is the basis of many applications. GPS is the most common choice for outdoor localization. However, the accuracy of GPS is not ideal in crowed cities and it is sensitive to severe weather because satellite signal cannot overcome multipath and block well. In addition, GPS needs devices to be GPS enabled.

Received signal strength (RSS) based fingerprint localization only uses RSS from different wireless access points (WAP), such as base stations or Wi-Fi, to localize devices [1]. Fingerprint localization can outperforms traditional ones because the latter require line-of-sight measurement [2], which can be strict condition in cities. Fingerprint localization consists of two parts: offline and online stages. At offline stage, database is established, which contains pairs of coordinates in physical space and RSSs in signal space. At online stage, location will be computed based on input RSSs.

"*K Nearest Neighbors (KNN)* and its variants have been widely used in indoor positioning for its low-cost and high performance" [3]. *Weighted-KNN* is one of the variants, whose performance is better than classic KNN [4].

Except for coefficient k, metrics of distance is also an important coefficient in KNN. Only some metrics are frequently used and compared, such as Euclidean distance, Manhattan distance, and cosine similarity. There are already some comparisons between these common metrics. When Manhattan distance is compared to Euclidean distance and Tanimoto distance, it is the best choice [5]. And cosine similarity can also achieve lower error compared to Euclidean distance [6]. However, all of the metrics above give the same weight to big RSS and small RSS, which may not be an ideal choice. An exponential transformation is proposed to solve this problem [7]. But coefficient setting in transformation was not analyzed there and related experiment will be carried out in this paper.

Manifold is a classic assumption in machine learning that high dimensional inputs typically live on or near a low dimensional manifold [8] and slight variance in sample will form a manifold. In that case, the actual distance between two points is not a simple distance between them but distance between two manifolds that they live on. Tangent distance was firstly proposed to solve hand-written digital recognition problem [9]. It is a local linear approximation of distance between manifolds. However, the original tangent distance is based on Euclidean distance and it cannot outperform Manhattan distance in our experiments. Hence, Manhattan tangent distance should be proposed and it may improve accuracy.

The rest of this paper is organized as follows. Section II introduces fingerprint localization, KNN, common metrics used in KNN, and analyzes the disadvantage of common metrics. Section III illustrates the idea behind tangent distance, explains method to form tangent lines, derives Euclidean tangent distance, Manhattan tangent distance, and approximate Manhattan tangent distance. Section IV firstly introduces how the data are collected and features of data collection area. Secondly, it demonstrates the relationship between RMSE and exponential transformation coefficient based on four common metrics in KNN. Thirdly, it demonstrates the relationship between RMSE and neighbor number in computing derivative of tangent distance. Finally, it compares the results of Euclidean distance, Manhattan distance and tangent distance based on them. Section V concludes the paper.

Notation. Throughout the paper, the following notations are used. Lowercase and uppercase letters denote scalar values. Bold uppercase letters and bold lowercase letters denote matrices and vectors, respectively. $\|\cdot\|_2$ is Euclidean distance and $\|\cdot\|_1$ is Manhattan distance.

II. Common Distance Metrics in Fingerprint Localization

A. Fingerprint Localization and KNN

Fingerprint localization is based on the belief that two points whose distance in physical space is short will also have short distance in signal space and vice versa, since two near points should have similar channel information and multipath effect. Hence, if RSSs from different WAPs are known, the location can be estimated by combining the physical space coordinate of K nearest neighbors in signal space, which is a common machine learning algorithm. The combination method used here is weighted-KNN, whose weights are reciprocal of distance in signal space. k denotes the number of selected nearest neighbors, which is also the most obvious coefficient can be tuned. However, since nearest neighbors are defined by distance, distance metrics also should be selected carefully.

B. Common Metrics in KNN

First of all, let us define a physical-signal space pair: $(\mathbf{x}, \mathbf{s})$, where $\mathbf{x}$ is the physical space coordinate and $\mathbf{s}$ is RSS vector from different WAPs. And D denotes the dimension of signal vector $\mathbf{s}$.

As introduction said, Euclidean, Manhattan distance and Cosine similarity are some of the most common metrics.

Both Euclidean and Manhattan distance belong to Minkowski distance, which is defined by

$$d_{minkowski}(\mathbf{s}, \mathbf{r}) = \left(\sum_{i=1}^{D} |s_i - r_i|^p \right)^{\frac{1}{p}}, \tag{1}$$

where $\mathbf{s}, \mathbf{r}$ are RSS vectors at physical location $\mathbf{x}$ and $\mathbf{y}$, respectively. $p = 2$ in Euclidean distance and $p = 1$ in Manhattan distance.

The classic cosine similarity is defined by

$$Cosine(\mathbf{s}, \mathbf{r}) = \frac{\mathbf{s}^T \mathbf{r}}{\|\mathbf{s}\| \, \|\mathbf{r}\|}, \tag{2}$$

which takes value in the interval of $[-1, 1]$ and the closer is the value to 1, the more similar is two vectors. However, it is opposite to distance, so $1 - Cosine(\mathbf{s}, \mathbf{r})$ will be taken as the metric.

The four metrics above will be taken into account to compare their root mean square error (RMSE) of localization. The results will be demonstrated in section IV, where Manhattan distance will achieve the lowest RMSE between these metrics.

C. Disadvantages of Common Metrics

Although the metrics above are widely applied in fingerprint localization, they still have some disadvantages:

1. Signal with different RSS share the same weights, which is not a rational choice. According to log-distance path loss model, when RSS value is big, the disturbance caused by noise will affect the distance from WAP to mobile device less than the case when RSS value is small.

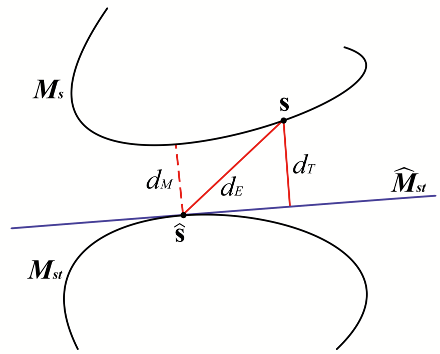

Fig. 1. Illustration of the relationship between Euclidean distance, manifold distance, and tangent distance. In signal space, $\mathbf{s}$ is test point and $\widehat{\mathbf{s}}$ is a training point. $\boldsymbol{M_s}$ and $\boldsymbol{M_{st}}$ are the manifolds that s and $\widehat{\mathbf{s}}$ live on, respectively. $\widehat{\boldsymbol{M}}_{\boldsymbol{st}}$ is the tangent line of $\boldsymbol{M_{st}}$ at $\widehat{\mathbf{s}}$. d_E denotes Euclidean distance between $\mathbf{s}$ and $\mathbf{s}_t$. d_M denotes the distance between manifold $\boldsymbol{M_s}$ and $\boldsymbol{M_{st}}$. d_T denotes tangent distance from $\mathbf{s}$ to manifold $\boldsymbol{M_{st}}$. Obviously, d_T is a better approximation of d_M than d_E.

   So it is rational to set bigger weight to big RSS. Note that signal in dBm is logarithm of nature signal and we can reverse the process to set bigger weight for big RSS:

$$f(\mathbf{s}) = 10^{\frac{\mathbf{s}}{10\lambda}}, \tag{3}$$

   where λ is a coefficient to control weights of RSS. The smaller is λ, the bigger is the relative weight of big RSS, which may cause small RSS ignored in distance calculation.
   Coincidentally, this idea has been approved by experiment [7]. However, the influence of λ was not analyzed there and it will be demonstrated in section IV.

2. Slight variance in sample will form a manifold. As to fingerprint localization, RSSs of a point in physical space and its nearby area should live on a manifold, because a local small area should share similar channel information and only varies slightly. In addition, in our dataset, the number of base station varies from 80 to 200, which means input dimension varies from 80 to 200. They are high dimensional inputs compared to 2d physical space. When manifolds exist in high dimensional space, simple metrics will not measure distance between points in signal space accurately.
   If a test point want to be classified or regressed accurately, a simple but inefficient even impossible method is to collect all of the possible data, which will form manifolds automatically, and simple metrics (such as Euclidean distance) will success to find minimum distance between manifolds. However, the situation is that size of dataset is limited, so a method to approximate distance between manifolds with limited data is needed.

III. Tangent Distance in Fingerprint Localization

Tangent distance is a local linear approximation of distance from test point to training point manifold. The distance from test point to the tangent line of training point manifold will be taken as metric, which is illustrated in Fig.1. Computing tangent distance only need to know the gradient at $(\widehat{\mathbf{x}}, \widehat{\mathbf{s}})$ instead of the whole manifold. Therefore, tangent distance needs much less data than actually computing distance between manifolds.

In theory, direction of tangent line is the same as gradient of manifold. However, gradient is still difficult to compute and difference is taken as direction of tangent line. Manifold is a function of small variance in physical space, which is distance here. For example, if $(\mathbf{x}, \mathbf{s})$ is a physical-signal space pair, manifold $\boldsymbol{M_s}$ at physical point $\mathbf{x} + a\boldsymbol{\delta}$ can be written as $\boldsymbol{M_s}(a)$. $\boldsymbol{\delta}$ is a unit vector denotes direction of physical space variance and a is a scalar denotes distance caused by variance. It seems not straightforward to take derivative with respect to (w.r.t.) distance instead of physical coordinate. The reason is that physical coordinate $\mathbf{x}$ is a 2d or 3d vector, which means partial derivative is needed. First of all, calculating partial derivative will take more time. What's worse, approximating partial derivative w.r.t. $x_i (i = 1, 2, 3)$ by difference needs variance in x_i but invariance in the other dimensions, which is a strict condition for data collection, especially for large-scale outdoor data collection. Actually, if data collection condition is satisfied, partial derivative or derivative makes no difference in the following analysis, except that optimization variable will become vector. In brief, partial derivative is also suitable for tangent distance.

A. Euclidean Tangent Distance

Tangent distance was proposed to be based on Euclidean distance [9], which is defined as

$$d_T = \min_a \left\| \mathbf{s} - \mathbf{t}a - \widehat{\mathbf{s}} \right\|_2, \tag{4}$$

where $\mathbf{s}$ and $\widehat{\mathbf{s}}$ are signal of training point and test point, respectively. $\mathbf{t}$ is the derivative near training point:

$$\mathbf{t} = \left(\frac{\partial s_1}{\partial d}, ..., \frac{\partial s_D}{\partial d} \right)^T, i = 1, 2, ..., D. \tag{5}$$

Since derivative should be approximated by difference, it is important to select points in training dataset for difference. Our method is to find N nearest neighbors from training data in physical space (different from fingerprint in signal space), and then N derivative will be computed by

$$\mathbf{t}_j = \frac{\mathbf{s} - \mathbf{s}_j}{d_j}, j = 1, 2, ..., N. \tag{6}$$

Note that $\mathbf{t}_j$ denotes jth derivative of $\mathbf{s}$ instead of jth element of vector $\mathbf{t}$. Metric of d_j is indifferent since it only affects derivative in scale instead of direction, whose influence will be eliminated during optimizing a in (4). Numerical experiment about N will be demonstrated in Section IV.

d_T is easy to compute, which is solving a linear least square problem:

$$a_j = \left({\mathbf{t}_j}^T \mathbf{t}_j \right)^{-1} {\mathbf{t}_j}^T \left(\mathbf{s} - \widehat{\mathbf{s}} \right), j = 1, 2, ..., N. \tag{7}$$

After all a_j have been computed, average of these tangent distance d_{Tj} in (4) will be considered as distance metric in KNN.

We call the distance above *Euclidean Tangent Distance (ETD)*. Although ETD is easy to solve, numerical experiment shows that Manhattan distance performs better than all of the common Metrics mentioned above, even better than ETD. Since ETD performs better than Euclidean distance, it may be induced that a Manhattan distance based tangent distance performs better than Manhattan distance. Hence, Manhattan tangent distance will be proposed.

B. Manhattan Tangent Distance

For simplicity, derivative $\mathbf{t}$ and Manhattan tangent distance d_{MT} in the following part will not be attached with subscript j. Similar to (4), Manhattan tangent distance can be defined by:

$$d_{MT} = \min_a \left\| \mathbf{s} - \mathbf{t}a - \widehat{\mathbf{s}} \right\|_1. \tag{8}$$

However, the optimization of a is not as easy as (4) because the derivative of L1-norm is sign function. Define $\mathcal{L} = \left\| \mathbf{s} - \mathbf{t}a - \widehat{\mathbf{s}} \right\|_1$, then

$$\frac{\partial \mathcal{L}}{\partial a} = -\mathbf{t}^T \mathrm{sgn} \left(\mathbf{s} - \mathbf{t}a - \widehat{\mathbf{s}} \right). \tag{9}$$

Solving (9) is difficult and even there is no solution. In that case, $\tanh(\beta x)$ will take place of $\mathrm{sgn}(x)$ here. When $\beta \to \infty$, $\tanh(\beta x) = \mathrm{sgn}(x)$. β is set as 10000 in our experiment.

Let $\mathbf{h} = \mathbf{s} - \mathbf{t}a - \widehat{\mathbf{s}}$, then

$$\frac{\partial \mathcal{L}}{\partial a} = -\mathbf{t}^T \tanh \left(\beta \mathbf{h} \right), \tag{10}$$

$$\frac{\partial^2 \mathcal{L}}{\partial a^2} = \beta \left(\mathbf{t} \odot \mathbf{t} \right)^T \left[\mathbf{1} - \tanh \left(\beta \mathbf{h} \right) \odot \tanh \left(\beta \mathbf{h} \right) \right], \tag{11}$$

where $\odot$ is Hadamard product (element-wise product) and $\mathbf{1}$ is an all one vector. Since derivative w.r.t. a has been changed from (9) to (10), primitive function also become $\frac{1}{\beta} \mathbf{1}^T \ln \left[\cosh \left(\beta \mathbf{h} \right) \right]$, which is also a convex function (it can be approved by $\frac{\partial^2 \mathcal{L}}{\partial a^2} > 0$ in (11)). Still, Manhattan distance will be considered as objective function because of computation complexity and slight difference between two function value. If $\mathbf{t}$ is partial derivative, it will become a matrix and a will become a vector whose dimension is the same with dimension of physical space. In addition, second-order derivative will become Hessian matrix. This will not change convexity of optimization.

Now that first and second order derivative are derived in (10) and (11), Newton's method can be applied to minimize $\mathcal{L}$ so that tangent distance is computed. We call this distance *Manhattan Tangent Distance (MTD)*. Its physical meaning is the linear approximate Manhattan distance (sum of absolute difference along all coordinates) between test point and the manifold of a training point, which is to find a point on tangent line to minimize Manhattan distance between it and test point.

C. Low Computation Complexity Approximate Solution

Although a method to solve MTD is proposed above, its disadvantage is obvious: high computation complexity. It is $\mathcal{O}(NDS)$ times of multiplication more than using Manhattan distance, where N is the neighbor numbers in computing gradient, D is the dimensionality of RSS and S is the step number that Newton's method needs to converge. Because each time when a tangent distance is computed, a convex optimization must be solved, which is not acceptable in a online system.

Since tangent distance is a local linear approximation of signal space manifold near physical space $\mathbf{x}$, tangent distance in (8) only approximate the manifold well with a closed to 0. In addition, the Taylor series of $\tanh(x)$ is $\tanh(x) = x + \mathcal{O}\left(x^3\right)$. Replace $\tanh(x)$ with its first-order Taylor series x and then $\frac{\partial \mathcal{L}}{\partial a}$ in (10) will become:

$$\frac{\partial \mathcal{L}}{\partial a} = -\beta \mathbf{t}^T \left(\mathbf{s} - \mathbf{t}a - \widehat{\mathbf{s}}\right). \tag{12}$$

The solution of (12) equals 0 is the same as (7). In other words, a will be estimated based on Euclidean distance but tangent distance will be computed based on Manhattan distance when a is taken back to objective function. We call it *Approximate Manhattan Tangent Distance (AMTD)*.

Computation complexity of AMTD is $\mathcal{O}(ND)$ times of multiplication more than using Manhattan distance, where N is the neighbor numbers in computing gradient and D is the dimensionality of RSS.

IV. Experiment and Results

A. Data Collection and Preprocessing

Data used in the following parts were collected by ourself in Tsinghua University, Beijing, China. Data were collected by a field tester, which collects RSS from nearby base stations and records GPS coordinate at the same time. Data were collected in three different area: lotus pond, teaching area, and dormitory area. Features of these area and data collected there are:

1. Lotus pond with an island: There are many trees but no high building. The total area is 0.097km^2. Data was collected when collectors were walking and sample frequency was 0.2Hz. Average distance between two temporal sequences is 5.0m. There are 621 samples and 85 base stations in this dataset.
2. Teaching area: There are some high buildings and trees but roads are wide and distance between buildings is long. The total area is 0.416km^2. Data was collected when collectors were riding bicycle and sample frequency was 0.5Hz. Average distance between two temporal sequences is 4.8m. There are 2273 samples and 204 base stations in this dataset.
3. Dormitory area: There are many high buildings with narrow roads. The total area is 0.143km^2. Data was collected when collectors were riding bicycle and sample frequency was 0.5Hz. Average distance between two temporal sequences is 4.4m. There are 1035 samples and 149 base stations in this dataset.

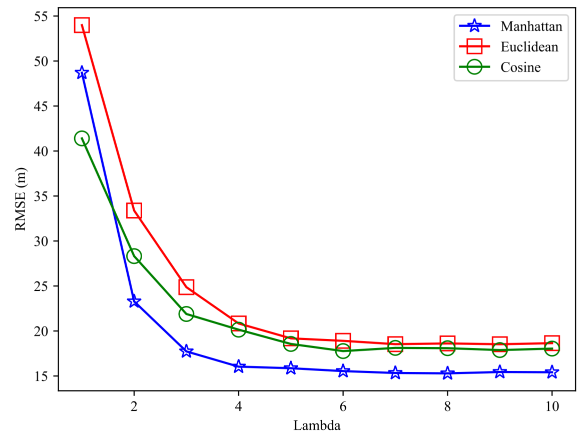

Fig. 2. Relationship between λ in exponential transform and RMSE. RMSE is stable when $\lambda \geq 7$.

TABLE I
RMSE Comparison of Exponential Transformation

RMSE/m	**Without Exp**	**With Exp** ($\lambda = 10$)
Manhattan	15.85	15.41
Euclidean	20.97	18.65
Cosine	20.28	18.04

Although there are many base stations, only some of these base station signal can be received in one sample. Minimum RSS of particular base station in dataset minus 1 will be considered as the RSS if this base station does not appear in sample.

B. Setting Value for λ in Exponential Transformation

Experiment was carried out to discover relationship between λ and RMSE based on different common metrics in KNN. Only data from lotus pond were used here because of time complexity. In order to eliminate influence of random partition and make curves smooth, 1000 times of experiment were carried out with each λ and 10% of data were chosen randomly for test each time. Results are demonstrated in Fig.2 and RMSE is stable when $\lambda \geq 7$. In order to make it more robust, $\lambda = 10$ will be applied in the following experiment.

As a contrast, localization without exponential transformation was also carried out and results are in Table I.

Obviously, Manhattan distance achieves the lowest RMSE in both of the experiment. In addition, the most common choice, Euclidean distance, is not the best choice. As to cosine similarity, it performs better than Euclidean distance but still much worse than Manhattan distance.

C. Setting Value for N in Tangent Distance Derivative

Still, data from lotus pond were used here. N is the number of neighbors used to estimate tangent distance. K-fold method

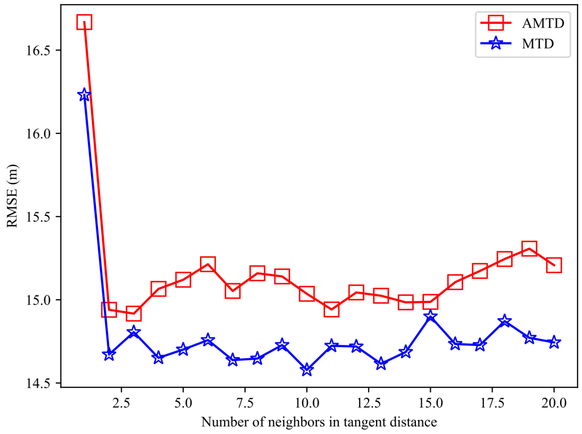

Fig. 3. Relationship between N in MTD and RMSE. With the increasing of N, RMSE decreases firstly and then increases.

TABLE II
RMSE COMPARISON OF TANGENT DISTANCE

RMSE/m	Lotus Pond	Teaching Area	Dormitory Area
Euclidean	19.20	13.56	10.29
ETD	16.69	11.47	9.54
Manhattan	15.78	9.96	7.95
MTD	**14.75**	**9.14**	**7.45**
AMTD	**14.90**	**9.43**	**7.63**

was utilized here with 10% of data divided for test in each fold and MTD, AMTD will be applied at the same time. The stopping criterion of MTD in Newton's method is 10^{-4}.

Fig.3 demonstrates the results. Although it is fluctuant, the trend is that RMSE decreases firstly and then increases. After consider accuracy and computation complexity together, $N = 5$ will be applied in the following experiment.

D. Performance of Tangent Distance

Before localizing, exponential transformation with $\lambda = 10$ are carried out to all of the following metrics, Euclidean distance, ETD, Manhattan distance, MTD, and AMTD. N in tangent distance are all set as 5, and k in KNN of fingerprint localization are all set as 3 (It is the optimal value but experiment results will not be shown here because of space limitations).

Results are demonstrated in Table II. Firstly, mention that ETD achieves lower RMSE than Euclidean but still higher RMSE than Manhattan and it is the motivation for proposing MTD. Secondly, it shows that MTD achieves the lowest RMSE on all three dataset. In addition, although RMSE of AMTD is a bit higher than MTD, it still performs better than Manhattan distance, which achieves the lowest RMSE among common metrics. It is obvious that data collected in dormitory area are improved the lest with MTD and AMTD. Note that average distance between two temporal sequences in dormitory area is the lest. Therefore, simple metrics like Manhattan distance can approximate distance between manifolds better than in the other datasets, which may be the reason. However, accuracy is affected by many other factors, such as channel information. Compared to the other area, buildings in dormitory area are intensive and channel varies greatly w.r.t. distance, which may limit the improvement of MTD and AMTD.

V. CONCLUSION

This paper discusses the application of tangent distance in fingerprint localization based on RSS with weighted-KNN algorithm. Firstly, the common metrics in KNN are introduced and their disadvantages are analyzed. For these disadvantages, exponential transformation and tangent distance are utilized. Then, the idea behind tangent distance is illustrated, Euclidean tangent distance, Manhattan tangent distance, and approximate Manhattan tangent distance are derived. Data are collected in three different area for analysis. Experiments for discovering relationship between RMSE and λ in exponential transformation, RMSE and N in tangent distance are carried out. Finally, results of comparison between Euclidean, ETD, Manhattan, MTD, and AMTD are demonstrated. In all three datasets, MTD achieves the lowest RMSE and AMTD achieves the second lowest RMSE with much lower computation complexity compared to MTD.

ACKNOWLEDGMENT

This work is supported by National Natural Science Foundation of China (No. 61631013), and the Fundamental Research Funds for the Central Universities (No. 2016JBM010).

REFERENCES

[1] P. Bahl and V. N. Padmanabhan, "Radar: An in-building rf-based user location and tracking system," in *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 2. Ieee, 2000, pp. 775–784.

[2] C. Gentile, N. Alsindi, R. Raulefs, and C. Teolis, *Geolocation Techniques: Principles and Applications*. Springer Science & Business Media, 2012.

[3] S. Xia, Y. Liu, G. Yuan, M. Zhu, and Z. Wang, "Indoor fingerprint positioning based on wi-fi: an overview," *ISPRS International Journal of Geo-Information*, vol. 6, no. 5, p. 135, 2017.

[4] L. Yen, C.-H. Yan, S. Renu, A. Belay, H.-P. Lin, and Y.-S. Ye, "A modified wknn indoor wi-fi localization method with differential coordinates," in *Applied System Innovation (ICASI), 2017 International Conference on*. IEEE, 2017, pp. 1822–1824.

[5] N. Marques, F. Meneses, and A. Moreira, "Combining similarity functions and majority rules for multi-building, multi-floor, wifi positioning," in *Indoor Positioning and Indoor Navigation (IPIN), 2012 International Conference on*. IEEE, 2012, pp. 1–9.

[6] S. Han, C. Zhao, W. Meng, and C. Li, "Cosine similarity based fingerprinting algorithm in wlan indoor positioning against device diversity," in *Communications (ICC), 2015 IEEE International Conference on*. IEEE, 2015, pp. 2710–2714.

[7] J. Torres-Sospedra, R. Montoliu, S. Trilles, Ó. Belmonte, and J. Huerta, "Comprehensive analysis of distance and similarity measures for wi-fi fingerprinting indoor positioning systems," *Expert Systems with Applications*, vol. 42, no. 23, pp. 9263–9278, 2015.

[8] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[9] P. Y. Simard, Y. A. LeCun, J. S. Denker, and B. Victorri, "Transformation invariance in pattern recognitiontangent distance and tangent propagation," in *Neural networks: tricks of the trade*. Springer, 1998, pp. 239–274.