Journal of Computers Vol. 29 No. 1, 2018, pp. 175-188
doi:10.3966/199115992018012901016

Locality Preserving Semi-supervised Canonical Correlation Analysis for Localization in Wireless Sensor Network

Su-Wen Zhu[1,2], Xian-Huan Zeng[1,2*]

[1] College of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing, 400065, China

[2] Chongqing Key Laboratory of Computational Inelligence, Chongqing, 400065, China
zhusuwen@foxmail.com, zengxh@cqupt.edu.cn

Received 13 October 2016; Revised 6 March 2017; Accepted 30 March 2017

Abstract. RSSI-based localization technique in Wireless Sensor Network is aimed at building a mapping between signal and physical spaces. The mapping could be overfitting when the number of paired RSSI and location data is small, and the collection of paired data is difficult, so unpaired data could be useful in improving the performance. This paper proposes the Locality Preserving Semi-Supervised Canonical Correlation Analysis (LPSemiCCA) algorithm for localization in Wireless Sensor Network, which combines PCA and CCA smoothly using a tradeoff parameter to overcome problems like sensitivity to data scale of PCA and incapability of utilizing unpaired data of CCA. The algorithm introduces similarity matrices of paired data and whole data to fit the structure of network and employs unpaired data efficiently. Locality Preserving Projection is also applied to construct the objective function in each domain, so the mapping can be calculated in condition of preserving the inner local structure of data. Experimental results in both simulated and realistic data show a higher localization accuracy of the proposed algorithm compared with LapLS, PPLCA and LapSVR.

Keywords: canonical correlation analysis, locality preserving projection, semi-supervised learning, wireless sensor network, localization

1 Introduction

Wireless Sensor Network (WSN) is a distributed network, whose terminals are sensors that can transmit and receive signals. WSN has been widely applied to real life as to the variety of sensors. In many practical applications such as target tracking [1-3], network routing [4-6] and monitoring system [7-8], location is the fundamental base of their realization. Currently, the most widely used Global Position System (GPS) can achieve better performance in places where satellites could reach, yet its equipment cost and energy consumption are relatively high. So, to locate unknown nodes in network using location of partial known nodes is of great value. The goal of localization is to locate physical locations of nodes according to acquired data like Received Signal Strength Indicator (RSSI) etc. Compared with other technologies for localization, RSSI-based technologies consume less power and cost of equipments, so RSSI-based localization algorithms [9-11] gain more attention. The main purpose of these algorithms is to establish a mapping between signal and physical spaces leveraging collected RSSI and corresponding location data, then the mapping can be employed as a prediction model for calculating the coordinates of unknown nodes.

Machine learning methods such as SVR learning [12-13], manifold learning [14-15] and neural network [16] have been applied to WSN node localization. However, to achieve high accuracy, these methods require a large amount of labeled data, whose collection consumes plenty of human effort and hardware equipment. Semi-Supervised learning [17] can take advantage of a small number of labeled data whose collection is relatively difficult and a large number of unlabeled data that is often more

* Corresponding Author

available. Semi-Supervised learning related localization algorithms [18-21] have appeared in recent years. Partially Paired Locality Correlation Analysis (PPLCA) algorithm in [18] introduces local similarity information into Canonical Correlation Analysis (CCA) to learn a mapping that maximizes correlation between signal and physical spaces; it can fit WSN structure and utilize unlabeled data. The Semi-Supervised Laplacian least square (LapLS) algorithm [19] employs Manifold Regularization [22] learning method to address the localization problem with absent physical locations. The Laplacian Support Vector Regression (LapSVR) algorithm [20] extends standard SVR to the Semi-Supervised SVR by using Manifold Regularization [22] to realize the usage of unlabeled data. Semi-Supervised Colocalization [21] combines collaborative filtering with graph-based Semi-Supervised learning to locate unknown nodes. It aims at employing a few labeled data and a large number of unlabeled data to train localization model and improve accuracy while reducing difficulty of data collection and computation complexity. CCA is a classical method to build the mapping between two datasets, but it can only utilize paired data whose number is limited, which could bring about overfitting problem, and is unable to use unpaired data to reduce the workload of data collection. Semi-Supervised CCA (SemiCCA) method proposed in [23] inherits the characters of both CCA and PCA so that the structure in each domain and the co-occurrence information of paired data are smoothly controlled. SemiCCA for automatic image annotation can achieve better effect with less image information.

This paper proposes LPSemiCCA algorithm in the prototype of SemiCCA in [23]; the algorithm introduces similarity matrices of the paired data and the whole data to preserve inner structure of data, and applies Locality Preserving Projection (LPP) [24] to construct the objective function in each domain as its characteristic of being able to transform global nonlinear problem into several local linear problems, so the mapping that maximizes correlation between signal and physical spaces can be calculated in circumstance of preserving the inner local structure of data. The main contributions of this paper are: (1) Construct an LPSemiCCA model combined with inner local structure information of data by introducing similarity matrices of the labeled data and the whole data into SemiCCA; (2) The LPSemiCCA model for WSN localization can fit the topology structure of the network to ensure the localization accuracy as its property of locality preserving, and employ unpaired data efficiently to reduce the localization cost and avoid overfitting for its character of Semi-Supervised.

2 Related Work

Under the same node power and similar transmission mode, if two nodes are neighbors in the sensor network, their RSSI receiving from the same anchor node should be similar, so there exists a bijection between signal and physical spaces [25]. Machine learning methods for WSN localization can be treated as learning a appropriate mapping between signal space consisting of RSSI and physical space comprised of location, then the location of unknown node can be obtained by the mapping. Assume that there exist p access points (APs) in an area $C \subseteq R^2$ that we are interested in. APs periodically send out signals to other non-APs. To establish the mapping, location data $Y = [y_1, y_2, ..., y_n] \in R^{q \times n}$ of n known nodes and corresponding RSSI data $X = [x_1, x_2, ..., x_n] \in R^{p \times n}$ receiving from p APs should be collected. When RSSI data and corresponding location data exist at the same time, the data is called paired data, also called labeled data, that is to say, RSSI and location are mutual label for each other. When only one of RSSI and location data exists, the data is called unpaired data, also called unlabeled data. In some realistic scenarios, to reduce the collection difficulty of data or influenced by environment, unpaired data $X^{(U)} = [x_{n+1}, ..., x_{n+ps}] \in R^{p \times ps}$ and (or) $Y^{(U)} = [y_{n+1}, ..., y_{n+pl}] \in R^{q \times pl}$ are acquired.

The localization process should achieve higher accuracy by low manpower and hardware consumption, i.e., reducing the data collection workload and guaranteeing that the data collected under harsh surroundings (bad weather, signal jamming, etc.) still contributes to improving the localization accuracy. Supervised localization methods could be overfitting when the number of paired data is small. Thus, this paper focuses on localization algorithm that can employ unpaired data efficiently to reduce the localization cost and avoid the overfitting problem.

2.1 CCA

In machine learning field, CCA [26] is a classical method to build the mapping between two datasets. Here, a group of paired and centered datasets are given, where, $X=[x_1,x_2,...,x_n]\in R^{p\times n}$, $Y=[y_1,y_2,...,y_n]\in R^{q\times n}$, p and q denote different dataset dimension, n denotes dataset number. The purpose of CCA is to find basis vectors w_x and w_y for two groups of datasets X and Y such that the correlation between the projections of the datasets to these bases vectors ${w_x}^T x$ and ${w_y}^T y$ are mutually maximized. Specifically, CCA can be formulated as the issue of computing maximum value of correlation coefficient ρ [26]. When the function ρ is irrelative to scale, CCA can be represented as computing solution of optimization problem in Eq. (1). Where C_{xx} and C_{yy} denote the within-set covariance matrices, and $C_{xy}={C_{yx}}^T$ denotes the between-set covariance matrix.

$$\begin{cases}\max\limits_{w_x,w_y} & {w_x}^T C_{xy} w_y \\ s.t. & {w_x}^T C_{xx} w_x=1 \\ & {w_y}^T C_{yy} w_y=1\end{cases} \tag{1}$$

To solve the optimization problem, Hardoon et al. provide a detailed solution [26]. Usually, a group of basis vectors $(w_{x_i},w_{y_i})(i=1,..,d)$ and corresponding eigenvalues λ_i can be obtained, then combine the basis vectors into two projection matrices $W_x=[w_{x1},w_{x2},...,w_{xd}]$ and $W_y=[w_{y1},w_{y2},...,w_{yd}]$. CCA is the fundamental method to compute the mapping between signal and physical spaces.

2.2 PPLCA

PPLCA in [18] aims at solving problems of incapablity of collecting location data or signal loss due to geographical condition or bad weather such that data acquired cannot be guaranteed in pairs. It introduces local similarity information between the paired data and the whole data into CCA to obtain the mapping between signal and physical spaces. Given two unpaired datasets $\tilde{X}=[X,X^{(U)}]=[x_1,...,x_n,x_{n+1},...,x_{n+ps}]\in R^{p\times(n+ps)}$ and $\tilde{Y}=[Y,Y^{(U)}]=[y_1,...,y_n,y_{n+1},...,y_{n+pl}]\in R^{q\times(n+pl)}$, similarity matrices between the paired data and the whole data are defined as $\tilde{S}_X=\{\tilde{S}_{ij}^X\}_{i,j=1}^{n,n+ps}$, $\tilde{S}_Y=\{\tilde{S}_{ij}^Y\}_{i,j=1}^{n,n+pl}$. The similarity matrices above can be computed by imitating Eq. (2) [27], which can reveal local structure of datasets $\tilde{X}$ and $\tilde{Y}$.

$$\begin{aligned} S_{ij}^X &= \begin{cases}\exp(-\left\|x_i-x_j\right\|^2/t_x), & if\ i\ and\ j\ are\ neighbors\ in\ X \\ 0, & otherwise\end{cases}, \\ S_{ij}^Y &= \begin{cases}\exp(-\left\|y_i-y_j\right\|^2/t_y), & if\ i\ and\ j\ are\ neighbors\ in\ Y \\ 0, & otherwise\end{cases}\end{aligned} \tag{2}$$

t_x and t_y denote the average distance of signal space and physical space respectively, which can fit the structure of WSN better. Here, $t_x=\sum_{i=1}^n\sum_{i=1}^n\left\|x_i-x_j\right\|^2/n(n-1)$, $t_y=\sum_{i=1}^n\sum_{i=1}^n\left\|y_i-y_j\right\|^2/n(n-1)$.

Centralize the training data collected, and list the objective function and constraints of localization in Eq. (3).

$$\begin{cases}\max\limits_{\tilde{w}_x,\tilde{w}_y} & \tilde{w}_x^T\tilde{X}\tilde{G}_{XY}\tilde{Y}^T\tilde{w}_y \\ s.t. & \tilde{w}_x^T\tilde{X}\tilde{G}_{XX}\tilde{X}^T\tilde{w}_x=1 \\ & \tilde{w}_y^T\tilde{Y}\tilde{G}_{YY}\tilde{Y}^T\tilde{w}_y=1\end{cases} \tag{3}$$

Where, $A=[I_n,0]^T_{(n+ps)\times n}$, $B=[I_n,0]^T_{(n+pl)\times n}$, $\tilde{G}_{XY}=AB^T-A\tilde{S}_Y-\tilde{S}_X^TB^T+\tilde{S}_X^T\tilde{S}_Y$, $\tilde{G}_{XX}=AA^T-A\tilde{S}_X-\tilde{S}_X^TA^T+\tilde{S}_X^T\tilde{S}_X$, $\tilde{G}_{YY}=BB^T-B\tilde{S}_Y-\tilde{S}_Y^TB^T+\tilde{S}_Y^T\tilde{S}_Y$.

Optimize the objective function to obtain projection matrices $\tilde{W}_X$ and $\tilde{W}_Y$ along with generalized eigenvalues λ. Project paired data $\{X,Y\}$ to the new two-dimensional space $\{P^X,P^Y\}$ using $\{\tilde{W}_X:\tilde{W}_X^T\times X,\tilde{W}_Y:\tilde{W}_Y^T\times Y\}$, also project unknown data $x_g=(x_{g_1},x_{g_2},...,x_{g_p})^T\in R^{p\times 1}$ to the new space P^g using $\tilde{W}_X:\tilde{W}_X^T\times x_g$. Compute the weighted Euclidean distance of unknown node to all known nodes after transformation by Eq. (4) [25].

$$d_i=\sum_{j=1}^{2}\lambda_j(P_{ji}^X-P_j^g)^2,i=1,...,n \tag{4}$$

Where, d_i denotes the distance of i th known node to unknown node g; λ_j represents the j th value of i th eigenvalue; P_{ji}^X expresses the j th value of i th known node after transformation; P_j^g indicates the j th value of node g after transformation.

Finally, compute K nearest known nodes of unknown node g, that is, K minimum values of $d_i(i=1,...,n)$ and physical coordinates $\{y_1,y_2,...,y_K\}$ of corresponding known nodes. Then, the centroid of K known nodes is the estimated physical coordinate of node g. The innovation idea of this paper is mainly inspired by PPLCA.

2.3 SemiCCA

SemiCCA model proposed in [23] inherits the characteristics of both CCA and PCA so that the global structure in each domain and the co-occurrence information of paired data are smoothly controlled. SemiCCA for automatic image annotation can overcome problems like sensitivity to data scale of PCA and incapability of utilizing unpaired data of CCA. The main idea of SemiCCA is to solve the following objective function.

$$\begin{cases}\max\limits_{\tilde{w}_x,\tilde{w}_y} & \tilde{w}_x{}^T((1-\beta_B)\tilde{S}_{xx})\tilde{w}_x+2\tilde{w}_x{}^T\beta_BS_{xy}\tilde{w}_y+\tilde{w}_y{}^T((1-\beta_B)\tilde{S}_{yy})\tilde{w}_y\\ s.t. & \tilde{w}_x{}^T(\beta_CS_{xx}+(1-\beta_C)I_{Dx})\tilde{w}_x=1\\ & \tilde{w}_y{}^T(\beta_CS_{yy}+(1-\beta_C)I_{Dy})\tilde{w}_y=1\end{cases} \tag{5}$$

Where $X=[x_1,x_2,...,x_n]\in R^{p\times n}$ and $Y=[y_1,y_2,...,y_n]\in R^{q\times n}$ are paired data, $X^{(U)}=[x_{n+1},...,x_{n+ps},0,...,0]\in R^{p\times(ps+pl)}$ and $Y^{(U)}=[0,...,0,y_{n+ps+1},...,y_{n+ps+pl}]\in R^{q\times(ps+pl)}$ are unpaired data. $S_{xy}=\sum_{i=1}^{n}x_iy_i^T/n=S_{yx}{}^T$ is the between-set covariance matrix, $S_{xx}=\sum_{i=1}^{n}x_ix_i^T/n$, $S_{yy}=\sum_{i=1}^{n}y_iy_i^T/n$, $\tilde{S}_{xx}=\sum_{i=1}^{n+ps+pl}x_ix_i^T/(n+ps+pl)$, $\tilde{S}_{yy}=\sum_{i=1}^{n+ps+pl}y_iy_i^T/(n+ps+pl)$ are within-set covariance matrices. The covariance matrices are computed after all the data are centered. I_{Dx} and I_{Dy} are identity matrices; β_B and β_C are tradeoff parameters taking a value in [0,1]. SemiCCA method model is regarded as the prototype of our proposed algorithm.

3 The Proposed LPSemiCCA Algorithm

This paper presents the LPSemiCCA algorithm in the prototype of SemiCCA. First, similarity matrices of the paired data and the whole data are introduced to fit the network topology structure and meanwhile unpaired data are efficiently employed; Second, the characteristic of LPP which can transform global nonlinear problem into several local linear problems is utilized, and then the objective function in each domain is constructed to calculate the mapping that maximizes correlation between signal and physical

spaces in condition of preserving the inner local structure of data; Third, the merits of PCA and CCA are combined using a tradeoff parameter to overcome problems like sensitivity to data scale of PCA and incapability of utilizing unpaired data of CCA. The LPSemiCCA algorithm can employ unpaired data efficiently to reduce the localization cost and avoid the overfitting problem.

3.1 LPSemiCCA Model

Given two groups of unpaired data $\tilde{X}$ and $\tilde{Y}$, where $\tilde{X}=[x_1,x_2,...,x_n,x_{n+1},...,x_{n+ps},0,...,0]\in R^{p\times(n+ps+pl)}$, $\tilde{Y}=[y_1,y_2,...,y_n,0,...,0,y_{n+ps+1},...,y_{n+ps+pl}]\in R^{q\times(n+ps+pl)}$. The first n values in $\tilde{X}$ and $\tilde{Y}$ are paired data X and Y respectively; the ps values from $(n+1)$ to $(n+ps)$ in $\tilde{X}$ and pl values from $(n+ps+1)$ to $(n+ps+pl)$ in $\tilde{Y}$ are unpaired data. The similarity matrices of the paired data $S_X=\{S_{ij}^X\}_{i,j=1}^n$ and $S_Y=\{S_{ij}^Y\}_{i,j=1}^n$ can be computed according to Eq. (2), and the similarity matrices of the whole data $\tilde{S}_X=\{\tilde{S}_{ij}^X\}_{i,j=1}^{n+ps+pl}$ and $\tilde{S}_Y=\{\tilde{S}_{ij}^Y\}_{i,j=1}^{n+ps+pl}$ can also be obtained by imitating Eq. (2). The purpose of calculating similarity matrices is to learn the inner manifold structure of data, and then a mapping that maximizes correlation of two datasets and maintains inner structure of dataset can be established. For simplification, this paper uses one tradeoff parameter $\beta\in[0,1]$ to combine PCA and CCA that introduced into local structure information. In conclusion, the objective function and constraints of computing the mapping is shown as follows:

$$\begin{cases}\max\limits_{\tilde{w}_x,\tilde{w}_y} \ \tilde{w}_x{}^T((1-\beta)\tilde{X}\tilde{S}_{XX}\tilde{X}^T)\tilde{w}_x+2\tilde{w}_x{}^T\beta XS_{XY}Y^T\tilde{w}_y+\tilde{w}_y{}^T((1-\beta)\tilde{Y}\tilde{S}_{YY}\tilde{Y}^T)\tilde{w}_y \\ s.t. \quad \tilde{w}_x{}^T(\beta XS_{XX}X^T+(1-\beta)\tilde{X}I_{Dx}\tilde{X}^T)\tilde{w}_x=1 \\ \qquad \tilde{w}_y{}^T(\beta YS_{YY}Y^T+(1-\beta)\tilde{Y}I_{Dy}\tilde{Y}^T)\tilde{w}_y=1\end{cases} \tag{6}$$

The corresponding Lagrangian is(where λ_x and λ_y are Lagrangian coefficients):

$$\begin{aligned}L(\lambda,\tilde{w}_x,\tilde{w}_y)=&\tilde{w}_x{}^T((1-\beta)\tilde{X}\tilde{S}_{XX}\tilde{X}^T)\tilde{w}_x+2\tilde{w}_x{}^T\beta XS_{XY}Y^T\tilde{w}_y+\tilde{w}_y{}^T((1-\beta)\tilde{Y}\tilde{S}_{YY}\tilde{Y}^T)\tilde{w}_y \\ &-\lambda_x(\tilde{w}_x{}^T(\beta XS_{XX}X^T+(1-\beta)\tilde{X}I_{Dx}\tilde{X}^T)\tilde{w}_x-1) \\ &-\lambda_y(\tilde{w}_y{}^T(\beta YS_{YY}Y^T+(1-\beta)\tilde{Y}I_{Dy}\tilde{Y}^T)\tilde{w}_y-1)\end{aligned} \tag{7}$$

Taking derivatives with respect to $\tilde{w}_x$ and $\tilde{w}_y$ we obtain:

$$\begin{cases}\dfrac{\partial f}{\partial\tilde{w}_x}=((1-\beta)\tilde{X}\tilde{S}_{XX}\tilde{X}^T)\tilde{w}_x+\beta XS_{XY}Y^T\tilde{w}_y \\ \qquad -\lambda_x(\beta XS_{XX}X^T+(1-\beta)\tilde{X}I_{Dx}\tilde{X}^T)\tilde{w}_x=0, \ (8.1) \\ \dfrac{\partial f}{\partial\tilde{w}_y}=\beta YS_{YX}X^T\tilde{w}_x+((1-\beta)\tilde{Y}\tilde{S}_{YY}\tilde{Y}^T)\tilde{w}_y \\ \qquad -\lambda_y(\beta YS_{YY}Y^T+(1-\beta)\tilde{Y}I_{Dy}\tilde{Y}^T)\tilde{w}_y \quad =0. \ (8.2)\end{cases} \tag{8}$$

Subtracting $\tilde{w}_y{}^T$ times the second equation from $\tilde{w}_x{}^T$ times the first equation in (8) we have:

$$\lambda_y\beta\tilde{w}_y{}^TYS_{YX}X^T\tilde{w}_x-\lambda_x\beta\tilde{w}_x{}^TXS_{XY}Y^T\tilde{w}_y=0. \tag{9}$$

Then we can get that $\lambda_x=\lambda_y$, let $\lambda=\lambda_x=\lambda_y$.

So, the eigenvalue equations below are obtained.

$$\begin{cases}((1-\beta)\tilde{X}\tilde{S}_{XX}\tilde{X}^T)\tilde{w}_x+(\beta XS_{XY}Y^T)\tilde{w}_y=\lambda(\beta XS_{XX}X^T+(1-\beta)\tilde{X}I_{Dx}\tilde{X}^T)\tilde{w}_x \\ (\beta YS_{YX}X^T)\tilde{w}_x+((1-\beta)\tilde{Y}\tilde{S}_{YY}\tilde{Y}^T)\tilde{w}_y=\lambda(\beta YS_{YY}Y^T+(1-\beta)\tilde{Y}I_{Dy}\tilde{Y}^T)\tilde{w}_y\end{cases} \tag{10}$$

To simplify the eigenvalue equations, the following definitions are given:

$$\begin{cases} B = \beta \begin{pmatrix} 0 & XS_{XY}Y^T \\ YS_{YX}X^T & 0 \end{pmatrix} + (1-\beta) \begin{pmatrix} \tilde{X}\tilde{S}_{XX}\tilde{X}^T & 0 \\ 0 & \tilde{Y}\tilde{S}_{YY}\tilde{Y}^T \end{pmatrix}, \\ C = \beta \begin{pmatrix} XS_{XX}X^T & 0 \\ 0 & YS_{YY}Y^T \end{pmatrix} + (1-\beta) \begin{pmatrix} \tilde{X}I_{Dx}\tilde{X}^T & 0 \\ 0 & \tilde{Y}I_{Dy}\tilde{Y}^T \end{pmatrix}. \end{cases} \tag{11}$$

where $S_{XY} = S_X \circ S_Y = S_{YX}^T, S_{XX} = S_X \circ S_X, S_{YY} = S_Y \circ S_Y, \tilde{S}_{XX} = \tilde{S}_X \circ \tilde{S}_X, \tilde{S}_{YY} = \tilde{S}_Y \circ \tilde{S}_Y$, the operator '$\circ$' is element-by-element product in two matrices, I_{Dx} and I_{Dy} are identity matrices.

So, the solution of the mapping is converted to solve the following generalized eigenvalue problem.

$$B\begin{pmatrix} \tilde{w}_x \\ \tilde{w}_y \end{pmatrix} = \lambda C \begin{pmatrix} \tilde{w}_x \\ \tilde{w}_y \end{pmatrix}, \tag{12}$$

Obviously, the solution in (14) can be solved by eigenvalue decomposition. When C is invertible, Eq. (12) can be transformed into (13.1), otherwise into (13.2).

$$\begin{cases} C^{-1}B\begin{pmatrix} \tilde{w}_x \\ \tilde{w}_y \end{pmatrix} = \lambda \begin{pmatrix} \tilde{w}_x \\ \tilde{w}_y \end{pmatrix}, (13.1) \\ C^{+}B\begin{pmatrix} \tilde{w}_x \\ \tilde{w}_y \end{pmatrix} = \lambda \begin{pmatrix} \tilde{w}_x \\ \tilde{w}_y \end{pmatrix}. (13.2) \end{cases} \tag{13}$$

where $C^+ = C^T(CC^T)^{-1}$ is the Moore-Penrose pseudo-inverse of C. Then the correlation vectors $(\tilde{w}_x, \tilde{w}_y)$ and generalized eigenvalues λ are obtained. The projection matrices $\tilde{W}_X$ and $\tilde{W}_Y$ of data $\tilde{X}$ and $\tilde{Y}$ after the same transformation can also be achieved by combining correlation vectors $(\tilde{w}_x, \tilde{w}_y)$ into matrices.

LPSemiCCA model focuses on obtaining a mapping that maximizes correlation between two datasets, i.e. getting parameters of projection matrices $\tilde{W}_X, \tilde{W}_Y$ and eigenvalues λ.

3.2 LPSemiCCA Localization Algorithm

The goal of LPSemiCCA model for WSN node localization is to build a mapping that maximizes the correlation between signal space and physical spaces on the premise of fitting network topology structure as possible. The inputs of computing the physical location coordinates of unknown node are $\tilde{X} = [x_1, x_2, ..., x_n, x_{n+1}, ..., x_{n+ps},\ 0, ..., 0] \in R^{p\times(n+ps+pl)}$, $\tilde{Y} = [y_1, y_2, ..., y_n, 0, ..., 0, y_{n+ps+1}, ..., y_{n+ps+pl}] \in R^{q\times(n+ps+pl)}$, $x_g = (x_{g_1}, x_{g_2}, ..., x_{g_p})^T \in R^{p\times 1}$. $\tilde{X}$ and $\tilde{Y}$ are datasets consisting of RSSI and physical location respectively, in which the first n RSSI vectors in $\tilde{X}$ have corresponding location coordinates in $\tilde{Y}$, the ps RSSI vectors in $\tilde{X}$ from $(n+1)$ to $(n+ps)$ do not have corresponding location coordinates in $\tilde{Y}$, and the pl location coordinates in $\tilde{Y}$ from $(n+ps+1)$ to $(n+ps+pl)$ do not have corresponding RSSI vectors in $\tilde{X}$, and x_g is the RSSI vector of unknown node g in a certain moment. Solve eigen equation in Eq. (12), and then eigenvalues λ and projection vectors $(\tilde{w}_x, \tilde{w}_y)$ are obtained. After that, combine the projection vectors into two projection matrices $(\tilde{W}_x, \tilde{W}_y)$.

Project paired data $\{X = [x_1, x_2, ..., x_n], Y = [y_1, y_2, ..., y_n]\}$ and RSSI vector of unknown node $x_g = (x_{g_1}, x_{g_2}, ..., x_{g_p})^T$ into new two-dimensional spaces $\{P^X, P^Y, P^g\} = \{\tilde{W}_X^T \times X, \tilde{W}_Y^T \times Y, \tilde{W}_X^T \times x_g\}$. Compute the weighted Euclidean distance of unknown node to all known nodes after transformation using Eq. (4) [25].

Lastly, compute K nearest known nodes of unknown node g, that is, K minimum values of $d_i(i=1,...,n)$ and physical coordinates $\{y_1, y_2, ..., y_K\}$ of corresponding known nodes. Then, the centroid of K known nodes is the estimated physical coordinate of node g.

To sum up, node localization algorithm based on LPSemiCCA model includes two phrases, (1) training phrase: establish localization model by collected data, that is, learn the mapping between signal space and physical spaces; (2) localization phrase: estimate location of unknown node by the mapping learned from training phrase and the RSSI vector of unknown node in a certain moment. The localization algorithm is concluded as follows.

Algorithm 1. LPSemiCCA Localization Algorithm.

Input: RSSI matrix $\tilde{X} = [X, X^{(U)}] \in R^{p\times(n+ps+pl)}$, location matrix $\tilde{Y} = [Y, Y^{(U)}] \in R^{q\times(n+ps+pl)}$, RSSI matrix $X_g = (x_{g1}, x_{g2}, ..., x_{gm})$ of m unknown nodes and change interval $\Delta\beta$ of tradeoff parameter β.

Output: Location coordinates matrix Y_{gl} of m unknown nodes.

Step1: Compute space similarity matrices $S_X, S_Y, \tilde{S}_X, \tilde{S}_Y$ by Eq. (2).

for $j=1;\ \beta \in [0:\Delta\beta:1];\ j <= (1/\Delta\beta+1);$ **do**

(1) Train the collected data to obtain projection matrices $(\tilde{W}_X, \tilde{W}_Y)$ and two eigenvalues λ_1 and λ_2 according to Eq. (12);

(2) Project RSSI matrix X into new two-dimensional space using $P^X = \tilde{W}_X^T \times X$;

for $i=1;\ i <= m;$ **do**

(1) Project RSSI vector x_{gi} of unknown node gi into two-dimensional space using $P^{gi} = \tilde{W}_X^T \times x_{gi}$, and calculate the weighted Euclidean distance of unknown node to all known nodes after transformation by Eq. (4);

(2) Compute K nearest known nodes of unknown node gi and corresponding physical coordinates, the centroid of K known nodes is the estimated location l_{gi} of node gi;

(3) Obtain the location matrix $L_g = (l_{g1}, l_{g2}, ..., l_{gm})$ of all the unknown nodes;

end for

(3) Achieve $(1/\Delta\beta+1)$ estimated location coordinates matrix Y_g;

end for

Step 2: Output one coordinates matrix Y_{gl} with the least average localization error and corresponding optimal parameters $W_X, \lambda_1, \lambda_2, \beta$.

To demonstrate the LPSemiCCA localization algorithm more clearly, Fig. 1. gives an explicit description. Fig. 1. is the schematic diagram of LPSemiCCA localization algorithm, in which the gray part (rounded rectangles) is the training phrase and the blue part (rectangles) is the localization phrase. The gray part aims at finding appropriate projection matrices between signal and physical spaces; the blue part uses the projection to obtain two-dimensional space representation of paired data and unknown data, and the weighted Euclidean distance of unknown node to all known nodes after projection can be calculated, then the centroid of *K* nearest known nodes of unknown node is its estimated location.

4 Experiments Analysis

To verify the effectiveness of our algorithm, experiments on simulated data and realistic data are conducted. We also carry out PPLCA [18], LapLS [19] and LapSVR [20] as comparative algorithms. PPLCA can be applied to the scenario of loss of both RSSI data and location data; LapLS and LapSVR only use in such a case that the number of RSSI data is greater than or equal to the number of location data. To perform the experiments more elaborately, the change interval $\Delta\beta$ used is 0.001, so the iteration is 1001 times.

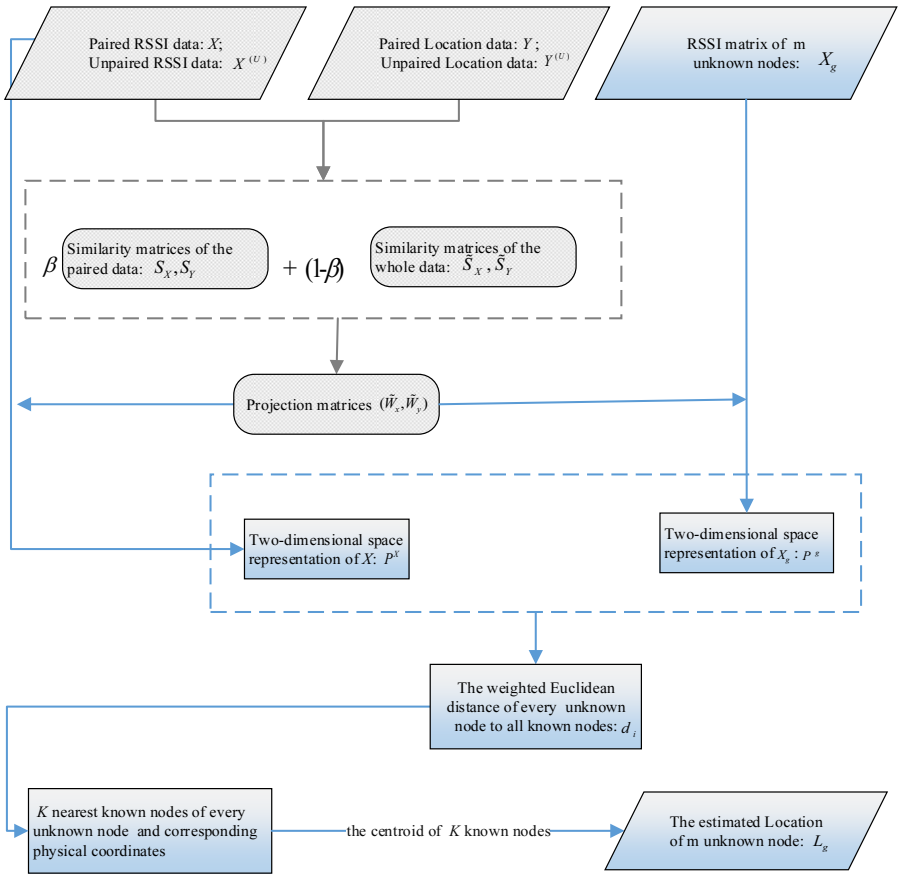


Fig. 1 The schematic diagram of LPSemiCCA localization algorithm

4.1 Experimental Setting

Simulated data. The simulated data is to simulate the random WSN in a square area C; the radio propagation model is regular network model; the radio range is circle; and the communication radius is 200m. Compared with log-normal shadowing model (Eq. (14)) [28], the regular network model do not have shadowing effect. The parameters $P_r(dist), P_r(d_0), \eta, dist, X_\sigma$ in Eq. (14) denote the received power at the distance $dist$ to the transmitter, the received power at the reference distance d_0 to the transmitter, the distance attenuation factor, the distance between receiver and transmitter, and the shadowing effect, respectively. The received power of node in the simulated network can be represented by Eq. (15). Where $P_t, P_l(d_0)$ are the transmitted power (0), path loss power (55) at the distance d_0 (1m), and the value of η is 4. The number of APs and all the nodes in the simulated network are 9 and 900 respectively. The network deployment graph is illustrated in Fig. 2, where red triangles represent APs and blue circles indicate non-APs.

$$P_r(dist) = P_r(d_0) - 10\eta \log \frac{dist}{d_0} + X_\sigma \tag{14}$$

$$P_r(dist) = P_t - P_l(d_0) - 10\eta \log \frac{dist}{d_0} \tag{15}$$

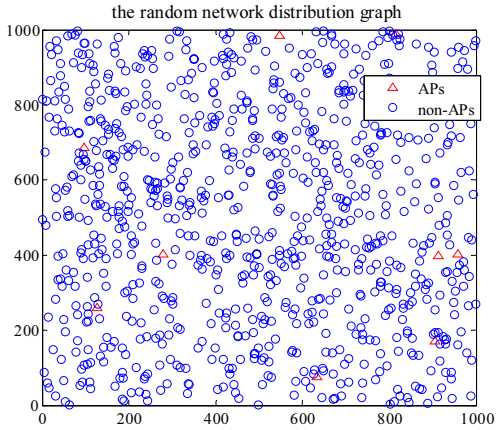


Fig. 2 The random network deployment graph in a square area C of simulated data, where red triangles represent APs and blue circles indicate non-APs

Realistic data. The realistic data [29] comes from School of Computer Science and Communication of the Royal Institute of Technology (KTH). An office environment ($400m^2$), including a hallway and a set of rooms, is the experimental place. A commercial Wi-Fi AP with a detachable external antenna is used as the radio signal source (transmitter). Five small USB wireless adapters of the same model (TP-Link TL-WN722N) with detachable external antennas are attached to the robot. These wireless adapters (Wi-Fi stations) act as the radio receivers and are connected to the AP using the IEEE 802.11n 2.4GHz channels. The communication radius of wireless adapters is 50m. When collecting data, the robot is teleoperated (at a velocity ≤ 0.2 m/s) to record physical coordinates in some locations and corresponding RSSI (dBm) [30].

The average location error of all the unknown nodes is adopted as error criterion to verify the performance of our algorithm. The smaller the *average location error* (ALE) is, the better the performance of the algorithm is. To ensure the veracity of experimental results, every group of experiments has been conducted at least twenty times.

4.2 Experimental Results

The experiments are divided into two parts, that is, to verify the influence of unlabeled data and labeled data to ALE (average location error) respectively.

No matter in simulated data or realistic data, the neighbor value k related to computing similarity matrices in PPLCA, LapLS, LapSVR and LPSemiCCA, and the value K related to the number of the nearest known nodes to unknown nodes gi in PPLCA and LPSemiCCA are determined by the optimal values of double iteration in PPLCA. The specific allocation is described in Table 1 to Table 4. The regularization coefficients in LapLS and LapSVR are defined as 0.0001 and 5. For simplification, *average error value* is abbreviated as AEV.

Table 1. Parameter setting and error comparison table in simulated data when labeled data remains unchanged

The neighbor value k	The value K	The smallest AEV	Iteration time of the smallest AEV	β value of the smallest AEV
3	6	0.3053	608	0.607
6	3	0.2865	625	0.624
5	3	0.2628	607	0.606
6	3	0.2673	679	0.678
6	4	0.2639	727	0.726
18	3	0.2346	85	0.084
6	3	0.2260	988	0.987

Table 2. Parameter setting and error comparison table in realistic data when labeled data remains unchanged

The neighbor value k	The value K	The smallest AEV	Iteration time of the smallest AEV	β value of the smallest AEV
29	4	0.4495	13	0.012
27	38	0.4460	111	0.110
32	4	0.4380	78	0.077
28	4	0.4350	938	0.937
27	45	0.4205	941	0.940
26	46	0.4180	943	0.942
25	41	0.3935	100	0.099
26	51	0.3535	178	0.177

Table 3. Parameter setting and error comparison table in simulated data when unlabeled data remains unchanged

The neighbor value k	The value K	The smallest AEV	Iteration time of the smallest AEV	β value of the smallest AEV
22	3	0.3776	319	0.318
8	3	0.3299	723	0.722
9	3	0.2622	962	0.961
3	4	0.2338	954	0.953
3	3	0.2129	985	0.984
3	3	0.2157	851	0.850
3	3	0.2107	813	0.812

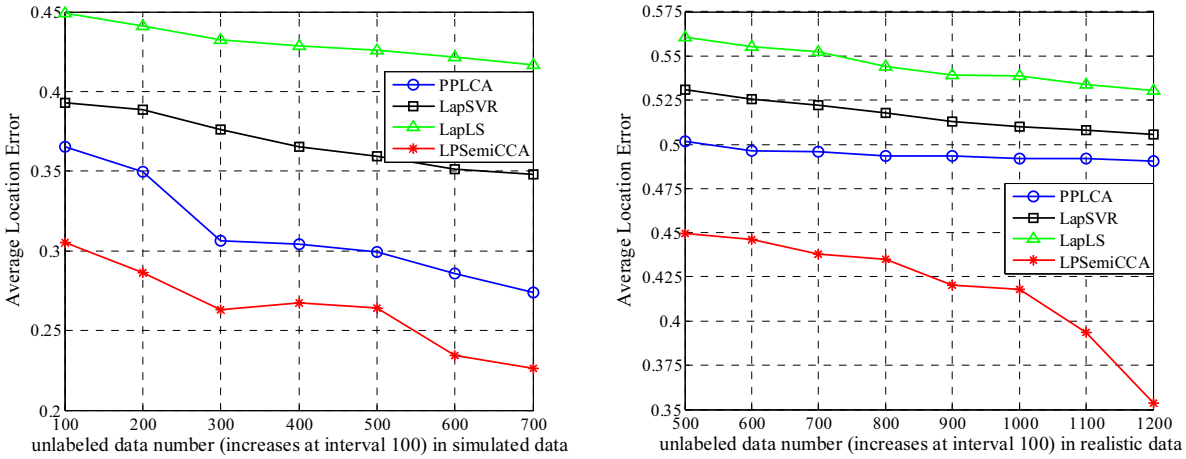
Table 4. Parameter setting and error comparison table in realistic data when unlabeled data remains unchanged

The neighbor value k	The value K	The smallest AEV	Iteration time of the smallest AEV	β value of the smallest AEV
8	4	0.2282	3	0.002
3	3	0.2217	2	0.001
25	3	0.2248	494	0.493
25	3	0.2137	875	0.874
3	3	0.2093	4	0.003
20	3	0.2010	970	0.969
9	3	0.2050	994	0.993
48	4	0.1648	844	0.843

The influence of unlabeled data to ALE. To test the influence of unlabeled data to ALE, simulated data are split into 7 groups, where the number of labeled data 200 remains unchanged, the number of unlabeled RSSI and location increase at interval 50 from 50 to 350 respectively, and the data of unknown nodes number 50 maintains unchanged. Realistic data are split into 8 groups, where the labeled data number 489 remains unchanged, unlabeled RSSI number increases at interval 50 from 300 to 650, unlabeled location number increases at interval 50 from 200 to 550, and data of unknown nodes number 300 maintains unchanged. Table 1 and Table 2 are parameter setting and error comparison tables when labeled data remains unchanged in simulated data and realistic data respectively.

Fig. 3(a) and Fig. 3(b) illustrate average location error comparison of four algorithms in simulated data and realistic data when labeled data number 200 and 489 remain unchanged respectively, where x-axis denotes unlabeled data number (increase at a certain interval), and y-axis denotes ALE.

The blue, black, green and red curve represent the change tends of PPLCA, LapSVR, LapLS and LPSemiCCA respectively when unlabeled data number increases. In Fig. 3(a), the ALE change ranges of PPLCA, LapSVR, LapLS and LPSemiCCA when unlabeled data number increases from 100 to 700 are [0.3652~0.2737], [0.3927~0.3471], [0.4492~0.4168] and [0.3053~0.2260]; In Fig. 3(b), the ALE change ranges of PPLCA, LapSVR, LapLS and LPSemiCCA when unlabeled data number increases from 500 to 1200 are [0.5015~0.4904], [0.5309~0.5053], [0.5605~0.5306], and [0.4495~0.3535].



(a) the number of labeled data (200) remains unchanged and unlabeled data number increases from 100 to 700 at interval 100 in simulated data

(b) the number of labeled data (489) remains unchanged and unlabeled data number increases from 500 to 1200 at interval 100 in realistic data

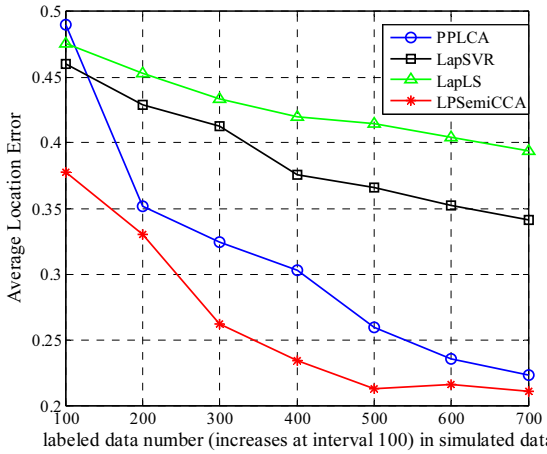
Fig. 3 The average localization error comparison graph of four algorithms

Through analyzing the change of ALE, we find that the four curves all decreases as the increasing of unlabeled data number, which means that the ALE of four algorithms are all gradually decreasing. It can be seen that unlabeled data has a positive influence on the training accuracy of localization model, and the usage of it can improve localization accuracy and meanwhile reduce workload of data collection. Moreover, the blue, black and green curves are always above red curve under the same parameter setting in Fig. 3(a) and Fig. 3(b), which indicates that the ALE of LPSemiCCA is lower than PPLCA, LapSVR and LapLS. It can be observed that the decrease trends of black and green curve are relatively gentle compared with other two curves in Fig. 3(a), which means the positive influence of unlabeled data to LapSVR and LapLS is not as strong as to PPLCA and LPSemiCCA. That is because LapSVR and LapLS can only utilize unlabeled RSSI rather than both unlabeled RSSI and location data, so data used for model training is relatively less. The decrease trend of red curve in Fig. 3(b) is the steepest, which means the positive influence of unlabeled data to LPSemiCCA is the most obvious.

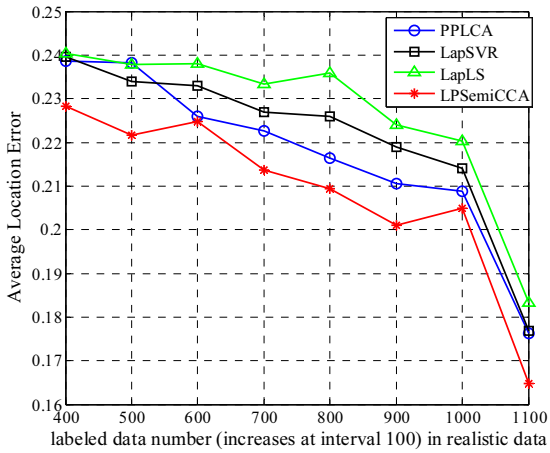
The influence of labeled data to ALE. To test the influence of labeled data to ALE, simulated data are split into 7 groups, where unlabeled RSSI number 100 and unlabeled location number 100 remain unchanged, labeled data number increases at interval 100 from 100 to 700, and data of unknown nodes number 100 maintains unchanged. Realistic data are split into 8 groups, where unlabeled RSSI number 400 and unlabeled location number 189 remain unchanged, labeled data number increases at interval 100 from 400 to 1100, data of unknown nodes number 400 maintains unchanged. Table 3 and Table 4 are parameter setting and error comparison tables when unlabeled data remains unchanged in simulated data and realistic data respectively.

Fig. 4(a) is the ALE comparison graph of four algorithms in simulated data when unlabeled data number 200 maintains unchanged, and Fig. 4(b) is the ALE comparison graph of four algorithms in realistic data when unlabeled data number 589 remains unchanged, where x-axis denotes labeled data number (increase at a certain interval), and y-axis denotes ALE.

The blue, black, green and red curve represent the change tends of PPLCA, LapSVR, LapLS and LPSemiCCA respectively when labeled data number increases. In Fig. 4(a), the ALE change ranges of PPLCA, LapSVR, LapLS and LPSemiCCA when labeled data number increases from 100 to 700 are [0.4899~0.2230], [0.4601~0.3411], [0.4759~0.3936] and [0.3776~0.2107]; In Fig. 4(b), the ALE change ranges of PPLCA, LapSVR, LapLS and LPSemiCCA when labeled data number increases from 400 to 1100 are [0.2387~0.1762], [0.2395~0.1768], [0.2404~0.1832], and [0.2282~0.1648].



(a) the number of unlabeled data (200) remains unchanged and labeled data number increases from 100 to 700 at interval 100 in simulated data



(b) the number of unlabeled data (589) remains unchanged and labeled data number increases from 400 to 1100 at interval 100 in realistic data

Fig. 4 The average localization error comparison graph of four algorithms

The change tendencies of four curves all declines as the growing of labeled data number in Fig. 4(a) and Fig. 4(b), which denotes that the ALE of four algorithms are gradually dropping. Training data for learning projection matrices is becoming more adequate as the growing number of labeled data, so the mapping between signal and physical spaces can be built more precise. Moreover, the red curve is always below the other three curves in the same parameter setting in Fig. 4(a) and Fig. 4(b), which means the ALE of LPSemiCCA is lower than PPLCA, LapSVR and LapLS. The growing number of labeled data indeed has a positive influence to improve the localization accuracy, but the workload of labeled data collection is rather difficult, so the labeled data number should be controlled considering application in realistic scenario. The best situation is to employ less labeled data and more unlabeled data to achieve higher localization accuracy.

To sum up, the average localization error (ALE) is an intuitive and reliable indicator to evaluate the performance of a localization algorithm. The smaller the ALE is, the better the performance of the algorithm is. The localization results are effective, if the error distance is within 1.5, i.e., the estimated coordinates of unknown nodes can relatively represent their absolute location [35]. The ALE of four algorithms including our algorithm and three comparative algorithms in every experiment is less than 1.5, and the ALE of our algorithm is basically smaller than the comparative algorithms', so the proposed algorithm is of great value.

5 Conclusion

This paper proposes the LPSemiCCA algorithm in prototype of SemiCCA model. It inherits merits of both PCA and CCA flexibly, and overcomes disadvantages of sensitivity to data scale of PCA and incapability of utilizing unpaired data of CCA. The algorithm also introduces similarity matrices of paired data and whole data to fit the structure of network and employs unpaired data efficiently. Moreover, LPP is applied to construct the objective function in each domain, so the mapping can be computed in condition of preserving the inner local structure of data. Experiments on simulated and realistic data are conducted, and experimental results indicate that unlabeled data has a positive influence on improving localization accuracy. Furthermore, the performance of our algorithm is better than other four comparison algorithms in the same parameter setting. Our algorithm can achieve a better performance, but the tradeoff parameter could not be calculated in a robust way, and the algorithm is only suitable for two-dimensional space localization, which is not extended to three-dimensional space. These aspects will be the subject of our future work.

Acknowledgements

The author would like to thank the anonymous reviewers for their help. This work was supported by the National Natural Science Foundation of China (Grant No. 61672120), the Chongqing Natural Science Foundation Program (Grant No. cstc2015jcyjA40036) and Postgraduate Scientific Research and Innovation Project of Chongqing (Grant No. CYS15173).

References

[1] S.C. Oh, Using an adaptive search tree to predict user location, Journal of Information Processing Systems 8(3)(2012) 437-444.

[2] M.Z.A. Bhuiyan, G. Wang, A.V. Vasilakos, Local area prediction-based mobile target tracking in wireless sensor networks, IEEE Transactions on Computers 64(7)(2015) 1968-1982.

[3] X. Zhou, Y. Ge, X. Chen, A distributed cache based reliable service execution and recovery approach in MANETs, Journal of Convergence 3(1)(2012) 298-305.

[4] J. Chen, K. He, R. Du, Dominating set and network coding-based routing in wireless mesh networks, IEEE Transactions on Parallel and Distributed Systems 26(2)(2015) 423-433.

[5] A. Milan, K. Schindler, S. Roth, Multi-target tracking by discrete-continuous energy minimization, IEEE Transactions on Pattern Analysis and Machine Intelligence 38(10)(2016) 2054-2068.

[6] A.K. Yadav, S.K. Das, S. Tripathi, EFMMRP: Design of efficient fuzzy based multi-constraint multicast routing protocol for wireless ad-hoc network, Computer Networks 118(11)(2017) 15-23.

[7] T. Teraoka, Organization and exploration of heterogeneous personal data collected in daily life, Human-Centric Computing and Information Sciences 2(1)(2012) 1-12.

[8] J. Yang, J. Zhou, Z. Lv, A real-time monitoring system of industry carbon monoxide based on wireless sensor networks, Sensors 15(11)(2015) 29535-29546.

[9] Q. Luo, Y. Peng, J. Li, RSSI-based localization through uncertain data mapping for wireless sensor networks, IEEE Sensors Journal 16(9)(2016) 3155-3162.

[10] F. Yaghoubi, A.A. Abbasfar, B. Maham, Energy-efficient RSSI-based localization for wireless sensor networks, IEEE Communications Letters 18(6)(2014) 973-976.

[11] P.K. Sahu, E.H.K. Wu, J. Sahoo, DuRT: Dual RSSI trend based localization for wireless sensor networks, IEEE Sensors Journal 13(8)(2013) 3115-3123.

[12] J. Lee, B. Choi, E. Kim, Novel range-free localization based on multidimensional support vector regression trained in the primal space, IEEE Transactions on Neural Networks and Learning Systems 24(7)(2013) 1099-1113.

[13] J. Yoo, H.J. Kim, Target localization in wireless sensor networks using online semi-supervised support vector regression, Sensors 15(6)(2015) 12539-12559.

[14] N. Patwari, A.O. Hero, Manifold learning algorithms for localization in wireless sensor networks, in: Proc. 2004 ICASSP International Conference, 2004.

[15] J. Kashniyal, S. Verma, K.P. Singh, Wireless sensor networks localization using progressive Isomap, Wireless Personal Communications 92(3)(2017) 1281-1302.

[16] A. Payal, C.S. Rai, B.V.R. Reddy, Analysis of some feedforward artificial neural network training algorithms for developing localization framework in wireless sensor networks, Wireless Personal Communications 82(4)(2015) 2519-2536.

[17] O. Chapelle, B. Scholkopf, A. Zien, Semi-Supervised Learning, IEEE Transactions on Neural Networks 20(3)(2009) 542-542.

[18] J. Gu, S. Chen, T. Sun, Localization with incompletely paired data in complex wireless sensor network, IEEE Transactions on Wireless Communications 10(9)(2011) 2841-2849.

[19] B. Yang, J. Xu, J. Yang, Localization algorithm in wireless sensor networks based on semi-supervised manifold learning and its application, Cluster Computing 13(4)(2010) 435-446.

[20] J. Yoo, H.J. Kim, Semi-Supervised location awareness in wireless sensor networks using Laplacian support vector regression, International Journal of Distributed Sensor Networks 2014(1)(2014) 1-7.

[21] J.J. Pan, S.J. Pan, J. Yin, Tracking mobile users in wireless networks via semi-supervised colocalization, IEEE Transactions on Pattern Analysis and Machine Intelligence 34(3)(2012) 587-600.

[22] M. Belkin, P. Niyogi, V. Sindhwani, Manifold regularization: A geometric framework for learning from labeled and unlabeled examples, Machine Learning Research 7(11)(2006) 2399-2434.

[23] A. Kimura, H. Kameoka, M. Sugiyama, SemiCCA: Efficient semi-supervised learning of canonical correlations, in: Proc. 2010 ICPR International Conference, 2010.

[24] X.F. He, P. Niyogi, Locality preserving projections, in: Proc. 2004 ICONIP International Conference, 2004.

[25] J.J. Pan, J.T. Kwok, Q. Yang, Multidimensional vector regression for accurate and low-cost location estimation in pervasive computing, IEEE Transactions on Knowledge and Data Engineering 18(9)(2006) 1181-1193.

[26] D.R. Hardoon, S. Szedmak, J. Shawe-Taylor, Canonical correlation analysis: An overview with application to learning methods, Neural Computation 16(12)(2004) 2639-2664.

[27] T.K. Sun, S.C. Chen, Locality preserving CCA with applications to data visualization and pose estimation, Image and Vision Computing 25(5)(2007) 531-543.

[28] T.S. Rappaport, Wireless Communications: Principles and Practice, Prentice Hall PTR, Upper Saddle River, NJ, 1996.

[29] CRAWDAD dataset kth/rss (v. 2016-01-05), traceset: indoor. <http://crawdad.org/kth/rss/20160105/indoor>, 2016.

[30] S. Caccamo, R. Parasuraman, F. Baberg, Extending a UGV teleoperation FLC interface with wireless network connectivity information, in: Proc. 2015 IROS International Conference, 2015.