2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

A New Multi-Task Learning Based Wi-Fi Location Approach Using $L_{1/2}$-Norm

Wentao Mao
School of Computer and Information Engineering
Henan Normal University
Xinxiang, CHINA, 453007
Email: maowt.mail@gmail.com

Haicheng Wang
Department of engineering physics
TsingHua University
Beijing, CHINA

Shangwang Liu
School of Computer and Information Engineering
Henan Normal University
Xinxiang, CHINA, 453007

***Abstract*—While many existing multi-task learning based Wi-Fi location approaches pay more attention on the location performance, they generally neglect determining key access points(APs). In order to reduce maintenance cost in complex indoor environment, a new multi-task learning based Wi-Fi location approach is proposed to find the key APs with enough accuracy. First, we introduce extreme learning machine as basic method to establish a new multi-task learning machine. This machine is based on the assumption that the hypotheses learned from a latent feature space, rather than the original high-dimensional feature space, are similar, in which $L_{1/2}$-norm is utilized to construct $L_{2-1/2}$-norm to achieve joint feature selection in multi-task scenario. An alternating optimization method is employed to solve this problem, by iteratively optimizing the latent space and key features. Experiments on real-world indoor localization data are conducted, and the results demonstrate the effectiveness of the proposed approach.**

I. Introduction

Currently, indoor localization using 802.11 wireless LAN has attracted more and more interest from research and industrial communities. Many machine learning methods are widely applied to improve localization performance. Generally speaking, these methods firstly collect wireless signals, often the received signal strength(RSS) values, from various access points(APs), and then establish regression or classification models for localization[1]. When a real-time RSS value is input into this model, the corresponding localization information can be predicted.

At present, multi-task learning(MTL)[2-3] methods have get satisfactory localization results. These methods often assume the distribution of the collected signal data are fixed so that the localization model learned from one device, place or time can be used for another one. Following this assumption, some researches[4-6] apply MTL on multi-device/space/time settings according to their different practical requirements. However, these methods tend to neglect determining the key APs which are important for daily maintenance. This target can be achieved via the joint feature selection in multi-task learning.

Although many traditional MTL methods utilize L_1-norm or LASSO to get satisfactory selection results, the obtained sparsity is not enough in the application of Wi-Fi localization. For example, we need to determine the most key APs, even chosen again from the results of conventional L_1-norm based MTL feature selection. In other words, the results of traditional MTL feature selection are still not enough sparse in this scenario. To find key APs with enough localization precision, a new multi-task learning approach based on $L_{1/2}$-norm is proposed for Wi-Fi localization. The critical step of this approach is utilizing $L_{1/2}$-norm to construct $L_{2-1/2}$-norm regularizer in multi-task learning. Because $L_{1/2}$-norm has many advantages such as more sparse than L_1-norm, easier calculation than L_0-norm regularization[7], etc., the proposed approach can get more key APs than L_1-norm based methods. To our best knowledge, this research serves as a first attempt to establish $L_{2-1/2}$-norm based multi-task learning approach.

The contributions of our work are as follows. For Wi-Fi localization application, we develop a new approach for calibrating a new device by making use of data collected before on other devices, and some key APs can be determined while keeping precision to a certain degree. For machine learning, we develop a new feature selection multi-task learning algorithm based on $L_{2-1/2}$-norm for better exploiting sparsity. In the experimental section, we demonstrate our approachs effectiveness on real data sets.

II. Related works

Learning based Wi-Fi localization methods generally include two steps: in an offline step, a mobile device moving around the wireless environment is used to collect wireless signals from various APs. The received signal strength(RSS) values are then used to learn a

978-1-4799-1484-5/14/$31.00 ©2014 IEEE

statistical model with location information contained. In online localization step, this model is sued to infer the locations according to the real-time RSS values.

Many statistical and machine learning methods were introduced to solve this problem. For example, k-ernel learning[8] is used to achieve location estimation. Gaussian process was also utilized to make localization via constructing a latent variable model. And MTL or TL based localization approaches received more and more attentions in recent years. For example, Zheng[4] extended the general assumption mentioned above, and presented another assumption for multi-device problem: the data distribution for related tasks may not be similar in the high-dimensional feature space. On the contrary, he looks for an appropriate feature mapping to seek a latent low-dimensional feature space, in which new device can get useful information from integrating the data collected before. Many MTL and TL based localization methods[5-6] all employ similar idea.

Multi-task learning make each task get benefit from others via learning a set of related tasks jointly. An general assumption for MTL is that the hypotheses learned from the original high-dimensional feature space for related tasks are similar[2]. Another further assumption is that the hypotheses learned in a latent feature space are similar[3]. Based on these assumptions, multi-task feature selection is a key issue in real applications. Paredes[9] utilizes an irrelative task to learn main tasks via using its prior information to construct a sparse and efficient information expression. Argyriou[3] introduces L_1-norm regularization to extend LASSO for single task into the setting of multi-task, and selects important features while improving learning performance. Obozinski[10] presents L_{2-1}-norm for joint feature selection in multi-task learning. This method firstly calculates L_2-norm of one feature across tasks, and then computes L_1-norm across features. Kumar[11] utilizes a group of basic latent task to express each learning task, which is based on the assumption that related tasks have similar linear weights of these latent tasks. Totally speaking, these methods generally fail to seek stronger sparsity in multi-task learning.

III. Brief introduction of ELM

As studied by [12], the theoretical foundations of ELM is that SLFN with at most N hidden neurons can learn N distinct samples with zero error by adopting any bounded nonlinear activation function. Following this concept, Huang[13] proposed ELM algorithm whose main procedure is determining the output weights by a matrix pseudo-inversion computation after initializing the input weights and hidden layer biases randomly. As proved empirically by many researchers[14], ELM has very high learning speed, simple network structure and good generalization performance. Here a brief summary of ELM is provided.

Given a set of $i.i.d$ training samples $\{(\mathbf{x}_1,\mathbf{y}_1),\cdots,(\mathbf{x}_N,\mathbf{y}_N)\} \subset \mathbb{R}^d \times \mathbb{R}^m$, standard SLFNs with $\tilde{N}$ hidden nodes are mathematically formulated as:

$$\sum_{i=1}^{\tilde{N}} \beta_i g_i(\mathbf{x}_j) = \sum_{i=1}^{\tilde{N}} \beta_i g_i(\mathbf{w}_i \cdot \mathbf{x}_j + b_i) = \mathbf{o}_j,\ \ j=1,...,N \tag{1}$$

where $g(x)$ is activation function, $\mathbf{w}_i = [w_{i1}, w_{i2}, ..., w_{id}]^T$ is input weight vector connecting input nodes and the ith hidden node, $\boldsymbol{\beta}_i = [\beta_{i1}, \beta_{i2}, ..., \beta_{im}]^T$ is the output weight vector connecting output nodes and the ith hidden node, b_i is bias of the ith hidden node. Huang[13] has rigorously proved that then for N arbitrary distinct samples and any $(\mathbf{w}_i, b_i)$ randomly chosen from $\mathbb{R}^d \times \mathbb{R}$ according to any continuous probability distribution, the hidden layer output matrix $\mathbf{H}$ of a standard SLFN with N hidden nodes and is invertible and $\|\mathbf{H}\boldsymbol{\beta} - \mathbf{T}\| = 0$ with probability one if the activation function $g : \mathbb{R} \mapsto \mathbb{R}$ is infinitely differentiable in any interval. Then given $(\mathbf{w}_i, b_i)$, training a SLFN equals finding a least-squares solution of the following equation[13]:

$$\mathbf{H}\boldsymbol{\beta} = \mathbf{Y} \tag{2}$$

where:

$$\mathbf{H}(\mathbf{w}_1, ..., \mathbf{w}_{\tilde{N}}, b_1, ..., b_{\tilde{N}}, \mathbf{x}_1, ..., \mathbf{x}_{\tilde{N}})$$
$$= \begin{bmatrix} g(\mathbf{w}_1 \cdot \mathbf{x}_1 + b_1) & \cdots & g(\mathbf{w}_{\tilde{N}} \cdot \mathbf{x}_1 + b_{\tilde{N}}) \\ \vdots & \cdots & \vdots \\ g(\mathbf{w}_1 \cdot \mathbf{x}_N + b_1) & \cdots & g(\mathbf{w}_{\tilde{N}} \cdot \mathbf{x}_N + b_{\tilde{N}}) \end{bmatrix}_{N \times \tilde{N}}$$
$$\boldsymbol{\beta} = [\beta_1, ..., \beta_{\tilde{N}}]^T$$
$$\mathbf{Y} = [\mathbf{y}_1, ..., \mathbf{y}_N]^T$$

Considering most cases that $\tilde{N} \ll N$, $\boldsymbol{\beta}$ cannot be computed through the direct matrix inversion. Therefore, Huang[13] calculated the $smallest\ norm$ least-squares solution of equation (2):

$$\hat{\boldsymbol{\beta}} = \mathbf{H}^{\dagger}\mathbf{T} \tag{3}$$

where $\mathbf{H}^{\dagger}$ is the Moore-Penrose generalized inverse of matrix $\mathbf{H}$. Based the above analysis, Huang[13] proposed ELM whose framework can be stated as follows:

Step 1. Randomly generate input weight and bias $(\mathbf{w}_i, b_i)$, $i = 1, \cdots, \tilde{N}$.
Step 2. Compute the hidden layer output matrix $\mathbf{H}$.
Step 3. Compute the output weight $\hat{\boldsymbol{\beta}} = \mathbf{H}^{\dagger}\mathbf{T}$.

Therefore, the output of SLFN can be calculated by $(\mathbf{w}_i, b_i)$ and $\hat{\boldsymbol{\beta}}$:

$$f(\mathbf{x}_j) = \sum_{i=1}^{\tilde{N}} \hat{\beta}_i g_i(\mathbf{w}_i \cdot \mathbf{x}_j + b_i) = \hat{\boldsymbol{\beta}} \cdot h(\mathbf{x}_j)$$

IV. Multi-task learning for Wi-Fi localization

Different from the methods above, we attempt to establish a general multi-task learning framework which can improve generalization performance and select most important features. In this target, the input dimension is not high. Considering extreme learning machine(ELM)[13] has many advantages such as high learning speed, good generalization, etc, we use ELM as basic architecture to construct new multi-task learning framework. Due to the limitation of space, the introduction of ELM is omitted here.

A. Optimization target

Consider a two-dimensional Wi-Fi localization problem. Assume there are *m* APs, which periodically broadcast wireless signals. A device can measure the RSS values from *m* APs. Each RSS vector $\mathbf{x}^i = \left(x_1^i, x_2^i, \cdots, x_d^i\right)'$ is a data instance, and its label $y^i = (y_1^{(i)}, y_2^{(i)})$ is a 2-D location coordinate vector. In real application, we need to collect lots of labeled signal data $D_{src} = \left\{\left(\mathbf{x}_{src}^i, \mathbf{y}_{src,j}^i\right)\right\}$, $i = 1, \cdots, n_s$, $j = 1, 2$. Here our objective is to use D_{src} to help predict the labels for a target device, by which we will only collect a small number of labeled signal data.

Specifically speaking, we focus on the learning task on different devices. To share the information between devices, we need to find proper feature mapping functions $\varphi_t, t = 1, ..., T$, which can map the data collected from source and target devices into a common latent feature space. Like [2], we define the task relatedness by exploiting the shared structure by the hypotheses. More specifically, we consider ELM regression problem: $f(\mathbf{x}) = \boldsymbol{\beta} \cdot h(\mathbf{x})$, where $\boldsymbol{\beta}$ is output weight vector and $h(\cdot)$ is activation function. Each task t, $t = 1, \cdots, T$, has a hypothesis parameterized as β_t, then β_t share a common structure β_0 by:

$$\beta_t = \beta_0 + \nu_t, \quad t = 1, \cdots, T$$

where ν_t denotes the difference for each task *t*. We follow the idea of latent multi-task learning proposed in [4], and formulate our multi-task learning method under ELM setting as follows:

$$\begin{aligned} \min J(\boldsymbol{\beta}_0, \boldsymbol{\nu}_t, e_{it}, \Phi) = & \sum_{t=1}^{T}\sum_{i=1}^{m} e_{it}^2 + \tfrac{\lambda_1}{T}\sum_{t=1}^{T}\|\boldsymbol{\nu}_t\|^2 \\ & + \lambda_2\|\boldsymbol{\beta}_0\|^2 + \lambda_3\|\Phi\|^{\frac{1}{2}}{}_{2-1/2} \\ s.t. \quad & y_{it} - (\boldsymbol{\beta}_0 + \boldsymbol{\nu}_t)\cdot h(\phi_t(\mathbf{x}_{it})) = e_{it}, \\ & \lambda_1, \lambda_2, \lambda_3 > 0 \end{aligned} \tag{4}$$

From left to right in equation (1), we explain each term:

• The first term $\sum_{t=1}^{T}\sum_{i=1}^{m} e_{it}^2$ indicates the regression loss across total T tasks. Minimization this term equals to minimize the overall localization error over T devices.

• In the second term, minimizing $\|\boldsymbol{\nu}_t\|$ make the tradeoff between the relatedness of tasks in the latent feature space $\phi_t(x)$.

• In the third term, minimizing $\|\boldsymbol{\beta}_0\|^2$ means improving the generalization ability of ELM, which is shown in ELM theory[13]. Note regularization parameters λ_1 is forced to be larger than λ_2 in order to make the task related to each other.

• In the fourth term, $\|\Phi\|^{\frac{1}{2}}{}_{2-1/2}$ denotes the complexity and sparsity of mapping functions $\Phi = [\phi_1, \phi_2, \cdots, \phi_t, \cdots \phi_T]$. To make our problem tractable, we force ϕ_t as a linear transformation, as $\phi_t(\mathbf{x}) = \phi_t\mathbf{x}$. So Φ is $d \times (k \times T)$ matrix, $k < d$ is the dimension of the latent space. This term plays as a $L_{2-1/2}$-norm regularizer of mapping function, which makes the latent feature be sparse.

Note that because $L_{1/2}$-norm is not convex, then equation (1) is not jointly convex. We use an alternating optimization strategy to solve this problem. First, fixing Φ, optimize $(\boldsymbol{\beta}_0, \boldsymbol{v}_t, e_{it})$, and second, optimize mapping function Φ with fixed $(\boldsymbol{\beta}_0, \boldsymbol{v}_t, e_{it})$. The whole algorithm works iteratively until convergence.

1) First step: learning the regression model: Given mapping function matrix Φ, we need to construct he regression functions from equation (1) to predict the location labels $\{y_{it}\}$ for signal data $\{\mathbf{x_{it}}\}$. We follow the idea of [2] and [4], and re-formulate a feature mapping function as $F : X \times \{1, \cdots, T\} \rightarrow \Re$, which means a function across all tasks. So, we can re-define a special mapping function as the following uniform setting:

$$f(\mathbf{x}, t) = \boldsymbol{\beta} \cdot \varphi(\phi(\mathbf{x}, t))$$

where:

$$\left.\begin{aligned} \varphi(\phi(\mathbf{x}, t)) = (\tfrac{h(\phi(\mathbf{x}))}{\sqrt{\mu}}, \underbrace{\mathbf{0}, \mathbf{0}...\mathbf{0}}_{t-1}, h(\phi(\mathbf{x})), \underbrace{\mathbf{0}, ...\mathbf{0}}_{T-t}) \\ \boldsymbol{\beta} = (\sqrt{\mu}\boldsymbol{\beta}_0, \boldsymbol{\nu}_1, ...\boldsymbol{\nu}_T), \mathbf{0} \in R^d, \mu = \tfrac{T\lambda_2}{\lambda_1} \end{aligned}\right\} \tag{5}$$

Then we have the following theorem:

Theorem IV.1. *According to the definition in equation (2), the multi-task learning in equation (1) can be reformulated as a standard ELM problem:*

$$\begin{aligned} \min \tilde{J}(\boldsymbol{\beta}, e_{it}) = & \tfrac{C}{2}\sum_{t=1}^{T}\sum_{i=1}^{m} e_{it}^2 + \tfrac{1}{2}\|\boldsymbol{\beta}\|^2 \\ s.t. \quad & y_{it} - \boldsymbol{\beta}\cdot\varphi(\phi_t(\mathbf{x}_{it})) = e_{it} \end{aligned} \tag{6}$$

where $C = \frac{T}{\lambda_1}$

Proof. Because $\boldsymbol{\beta} = (\sqrt{\mu}\boldsymbol{\beta}_0, \boldsymbol{\nu}_1, ...\boldsymbol{\nu}_T)$, we have

$$\|\boldsymbol{\beta}\|^2 = u\|\boldsymbol{\beta}_0\|^2 + \sum_{t=1}^{T} \|\boldsymbol{v}_t\|^2$$

Given Φ, the term $\lambda_3 \|\Phi\|^{\frac{1}{2}}{}_{2-\frac{1}{2}}$ is fixed. Therefore, we have:

$$\begin{aligned}\min J &= \sum_{t=1}^{T}\sum_{i=1}^{m} e_{it}^2 + \tfrac{\lambda_1}{T}\sum_{t=1}^{T}\|\boldsymbol{\nu}_t\|^2 + \lambda_2\|\boldsymbol{\beta}_0\|^2 \\ &= \tfrac{T}{2\lambda_1}\sum_{t=1}^{T}\sum_{i=1}^{m} e_{it}^2 + \tfrac{1}{2}(\sum_{t=1}^{T}\|\boldsymbol{\nu}_t\|^2 + \tfrac{T\lambda_2}{\lambda_1}\|\boldsymbol{\beta}_0\|^2) \\ &= C\sum_{t=1}^{T}\sum_{i=1}^{m} e_{it}^2 + \tfrac{1}{2}\|\boldsymbol{\beta}\|^2\end{aligned} \tag{7}$$

By means of equation (2), the constraint can be rewritten as:

$$y_{it} - (\boldsymbol{\beta}_0 + \boldsymbol{\nu}_t)\cdot h(\phi_t(\mathbf{x}_{it})) = y_{it} - \boldsymbol{\beta}\cdot\varphi(\phi_t(\mathbf{x}_{it}, t)) = e_{it}$$

Applying Lagrange method, the dual of equation (3) is:

$$\begin{aligned} lL(\beta, \alpha_{it}) = &\frac{1}{2}\|\beta\|^2 + \frac{C}{2}\sum_{t=1}^{T}\sum_{i=1}^{m} e_{it}^2 \\ &- \sum_{t=1}^{T}\sum_{i=1}^{m}\alpha_{it}(\beta\cdot\varphi(\phi_t(\mathbf{x}_{it})) - y_{it} + e_{it})\end{aligned} \tag{8}$$

Let $\frac{\partial L}{\partial \boldsymbol{\beta}} = 0$, we have

$$\boldsymbol{\beta} = \sum_{t=1}^{T}\sum_{i=1}^{m}\alpha_{it}\varphi(\phi(\mathbf{x}_{it}))$$

Let $\frac{\partial L}{\partial e_{it}} = 0$, we have

$$C\cdot e_{it} = \alpha_{it}$$

Substituting these two equations into equation (5), we have:

$$\sum_{s=1}^{T}\sum_{j=1}^{m}\alpha_{sj}\varphi(\phi(\mathbf{x}_j), s)\varphi(\phi(\mathbf{x}_i), t) + \frac{\alpha_{it}}{C} = y_{it} \tag{9}$$

After calculating the system of linear equations (6), we obtain the regression model:

$$f_t(\mathbf{x})^* = \sum_{s=1}^{T}\sum_{i=1}^{m}\alpha_{is}^* k_{st}(\mathbf{x}_i, \mathbf{x})$$

where

$$K_{st}(\mathbf{x}_i, \mathbf{x}) = \varphi(\phi(\mathbf{x}_i, s))\cdot\varphi(\phi(\mathbf{x}, t))s, t = 1, 2..., T$$

2) Second step: learning the latent feature space: After calculating $(\boldsymbol{\beta}_0, \boldsymbol{v}_t, e_{it})$ from equation (6), we can rewritten equation (1) as:

$$\begin{aligned} l\min\tilde{J}(\Phi) = &\lambda_3\|\Phi\|^{\frac{1}{2}}{}_{2-1/2} \\ s.t\quad &\boldsymbol{\beta}_t\cdot h(\phi_t(\mathbf{x}_{it})) - y_{it} = e_{it}, \\ &\lambda_3 > 0\end{aligned} \tag{10}$$

Equation (7) equals to:

$$\min\tilde{J} = \sum_{i=1}^{m}\|\mathbf{Y}_i - \boldsymbol{\beta}_t\cdot h(\mathbf{X}_i\otimes\Phi)\|_F^2 + \lambda_3\|\Phi\|_{2-1/2}^{\frac{1}{2}} \tag{11}$$

Here $\mathbf{X}_i$ and $\mathbf{Y}_i$ denotes the $i-\mathrm{th}$ input and output of all T tasks. $\|\cdot\|_F$ is Frobenius norm of a matrix. Equation (8) contains two parts. One is squared loss function, and the other is $L_{2-1/2}$ regularizer which means calculating L_2-norm of each feature across tasks firstly and then calculating $L_{1/2}$-norm across features. Therefore, the key issue is summarized to calculate $L_{1/2}$-norm. Because $L_{1/2}$-norm regularizer is not convex and tractable easily, optimizing $L_{1/2}$-norm regularizer is generally transformed into a series of weighted L_1-norm regularization iterations[7], in which L_1-norm regularizer is convex. Here in the first iteration, it needs to solve a L_1-norm regularization problem, and in the second iteration, the weighted L_1-norm regularization problem is solved after a simple linear transformation.

Because solving $L_{1/2}$-norm regularization problem can be transferred to a series of L_1-norm regularization problems[7], $L_{2-1/2}$-norm regularization problem is essentially equal to a series of L_{2-1}-norm regularization problems. Different from the conventional joint feature selection based on L_{2-1}-norm which makes the upper limit of L_2-norm of each row in block matrix as constraint, $L_{2-1/2}$-norm uses the multiplier of upper limit of L_2-norm of each row with L_2-norm in the latest iteration as constraint. Therefore, the constraint in $L_{2-1/2}$-norm regularization merely adds a constant on the conventional L_{2-1}-norm regularizer, so the constraint set is also a closed convex set. Therefore, we can use the solving framework of L_{2-1}-norm regularization[9] to optimize $L_{2-1/2}$-norm regularizer. The solution procedure for equation (8) is as follows:

Step 1. Initialize the max iteration number K, and $\Phi^0 = [\mathbf{1}, \mathbf{1}, \cdots, \mathbf{1}]$
Step 2. Solving:

$$\begin{aligned}\Phi^{k+1} = \arg\min &\sum_{i=1}^{m}\|\beta\cdot h(\mathbf{X}_i\otimes\Phi) - \mathbf{Y}_i\|_F^2 \\ &+ \lambda_3\sum_{j=1}^{d}\frac{\|\Phi_j\|}{\sqrt{\|\Phi_j^k\|}}\end{aligned} \tag{12}$$

and set $k = k + 1$

Step 3. Solving equation (9) equals to solve the following optimization problem under constraint condition:

$$\begin{aligned} \min \quad & \sum_{i=1}^{m} \|\boldsymbol{\beta} \cdot h(\mathbf{X}_i \otimes \Phi) - \mathbf{Y}_i\|_F^2 + \lambda_3 \sum_{j=1}^{d} \mathbf{t}_j \\ s.t. \quad & \|\Phi^j\| \leq t_j \cdot \sqrt{\|\Phi_j^k\|} \\ & \mathbf{t} = [t_1, t_2, ...t_d]^T \end{aligned} \tag{13}$$

Step 4. Go to step 2 until $k = K$. Output Φ^K if reaching convergence.

V. Experiments

In this section, we test the benefits of the proposed multi-task learning approach in the application of Wi-Fi localization. We firstly use AutoCAD to partition the whole floor, as shown in Fig.1. In Fig.1, each grid

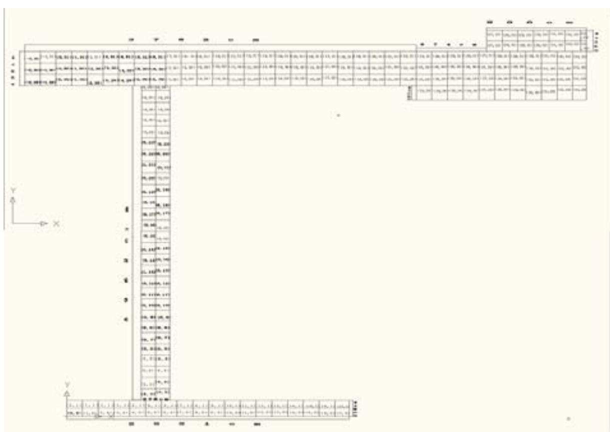

Fig. 1: Physical partition of collection area

denotes 1.2 square meters. We establish a coordinate axis according to these girds, and then get the label of each grid. Then we collect RSS values in our office buildings using two Android mobile phones. At each grid, we collected 50 samples for each device. The collected RSS vector has 48 dimensions, and the latent space dimension *k* is set to be 20.

For better comparison, two multi-task learning algorithms are introduced. The first is multi-task feature learning algorithm, called FS-MTL, proposed in [9] which employs L_{2-1}-norm regularizer. Another is Wi-Fi localization algorithm based on multi-task support vector machine[4]. This algorithm, called L-SVM in this article, uses L_2-norm to construct MTL model and also needs to find latent feature space. The proposed method is called L-MTL. Moreover, we also run the classical ELM as single-task algorithm for comparison. Each of input and output variables are rescaled linearly to the range $[-1, +1]$. All programs are carried out

TABLE I: Localization error of different methods

	Max	Min	Average	Std
ELM	4.382	2.921	3.164	0.921
FS-MTL	2.366	1.781	2.136	0.788
L-SVM	2.527	1.522	1.934	0.728
L-MTL	2.123	1.556	2.041	0.734

in MATLAB2010a environment running in a Core 2, 2.66GHz CPU and 3.37GB RAM. All results are the mean of 30 trials. Since the localization problem is a regression problem, we report the average error distance and the standard deviation. A method with lower error values is better.

First, we evaluate the influence of three regularization parameters on localization performance. In equation (1), λ_1, λ_2 and λ_3 control the tradeoff between empirical error, model complexity and latent feature space. Fig.2 provides the localization error of different parameters values.

From Fig.2, three parameters play quite different

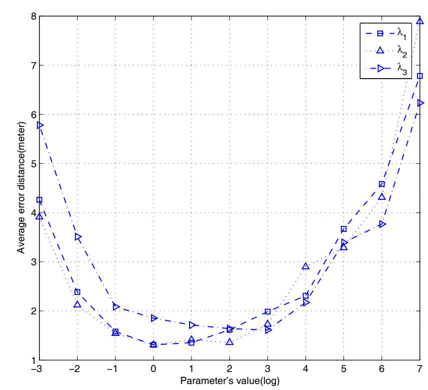

Fig. 2: Average error distance of different values of three parameters

role. Specifically for λ_3, the error is lowest at the value exp(3), which indicates the latent space is not very important. According to Fig.2, we set these three parameters sequentially as 1, 1 and 50 in the next experiments.

We also check the localization precision of four methods. The regularization parameters in FS-MTL and L-SVM are chosen via cross validation. Table 1 provides the comparative results.

Obviously, the error of L-MTL is not lowest. L-SVM get lowest error, but theres very few difference between L-MTL and L-SVM. Moreover, FS-MTL also get satisfactory results. Table 1 shows, for Wi-Fi localization, the conventional multi-task learning methods could reach enough good performance, but single-task learning method is hard to decrease localization error due to its few training sample.

Second, we evaluate the convergence rate. Because FS-MTL, L-SVM and L-MTL all employ iterative solution, the convergence rate and speed are two important issues in practical applications. Fig.3 illustrates the convergence speed of three multi-task learning methods. Here the sample size is 80.

Obviously, L-MTL reach convergence quickly. FS-MTL also get similar speed. Although L-SVM obtains lowest localization error, its convergence speed is more slow than two other methods. According to theoretical analysis, the reason is that L-SVM utilizes L_2-norm rather than L_1-norm used in L-MTL and FS-MTL. Despite the error is not lowest, L-MTL gets satisfactory results in selecting key features. Fig.4 shows the key features with different iteration number.

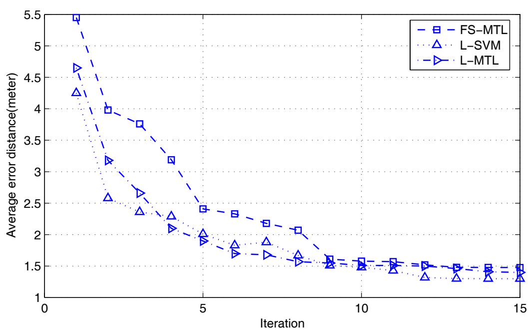

Fig. 3: Convergence speed of three methods

In Fig.4, L-SVM are not be considered, because it uses L_2-norm which has no sparsity. Compared with FS-MTL which uses L_{2-1}-norm, L-MTL gets better sparsity. Taking Fig.3 and Fig.4 into account together, although L-MTL cannot get lowest localization error, it exactly uses few key features to obtain similar localization error, which is just our algorithms value.

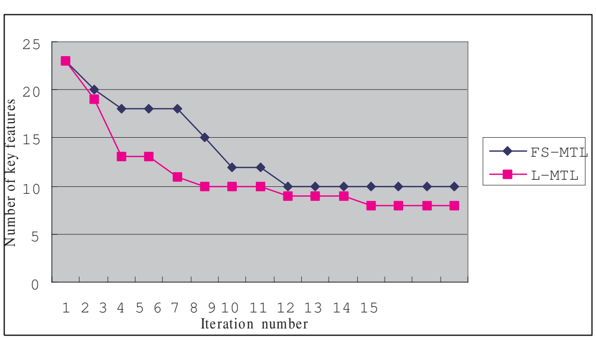

Fig. 4: Quantity of key features with different iteration number

Finally, we test the running time of these four methods. We run experiments 30 times, and get the average running time as final result, as shown in Fig.5. The settings of experiment are same to the above one.

Obviously, ELM needs least time, and the slowest one is L-SVM. According to section 3.2, L-MTL employs $L_{2-1/2}$-norm regularization which is more simple than L_2-norm based L-SVM and doesnt need to solve quadratic programming. Meanwhile, FS-MTL focuses on feature selection which increases algorithms complexity.

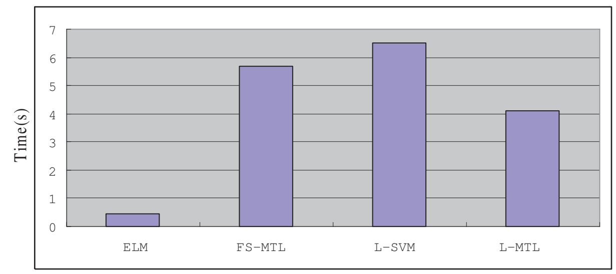

Fig. 5: Running time of four methods

VI. Conclusion

In this paper, a new Wi-Fi localization approach based on latent multi-task learning is proposed. To seek a common latent feature space, this approach constructs a regularization framework including mapping function as regularizer. To select key APs, the proposed approach employs $L_{1/2}$-norm rather than L_1-norm to achieve joint feature selection in multi-task scenario. A key step of this approach is to solve $L_{2-1/2}$-norm regularization problem. The experimental results on Wi-Fi data demonstrate the effectiveness of the proposed approach. There are two problems to be studied in our future research. First is a matter of choosing the appropriate value of latent spaces dimension, which will be determined by cross-validation in practice. Another problem is how to identify the inner structure in the tasks.

Acknowledgements

This work was supported by the National Natural Science Foundation of China(No. U1204609,U1304607) and Foundation and Advanced Technology Research Program of Henan Province, China(NO.132300410430).

References

[1] B. Ferris, D. Fox, and N. Lawrence. *Wifi-slam using Gaussian process latent variable models*. In: International Joint Conference on Artificial Intelligence(2007), pp. 2480-2485.

[2] T. Evgeniou, M. Pontil. *Regularized multi-task learning*. In KDD04: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, New York, USA: ACM, pp. 109-117, 2004.

[3] A. Argyriou, T. Evgeniou, and M. Pontil. *Multi-task feature learning*. In: Advances in Neural Information Processing Systems 19, Cambridge, MA: MIT Press, pp. 41-48, 2007.

[4] V.W. Zheng, S.J. Pan, Q. Yang and J.J. Pan. *Transferring Multi-device Localization Models using Latent Multi-task Learning*. In Proceedings of the 23rd AAAI Conference on Artificial Intelligence (AAAI 08), Chicago, USA. July 2008, pp. 1427-1432.

[5] S.J. Pan, D. Shen, Q. Yang and J.T. Kwok. *Transferring Localization Models Across Space*. In Proceedings of the 23rd AAAI Conference on Artificial Intelligence (AAAI 08), Chicago, USA, July 2008, pp. 1383-1388.

[6] V.W. Zheng, E.W. Xiang, Q. Yang and D. Shen. *Transferring Localization Models Over Time*. In Proceedings of the 23rd AAAI Conference on Artificial Intelligence (AAAI 08), Chicago, USA. July 2008, pp. 1421-1426.

[7] Z.B. Xu, H. Zhang, and Y. Wang, et al.*L(1/2) regularization*. SCIENCE CHINA-INFORMATION SCIENCES, 2010, 53(6): 1159-1169.

[8] X. Nguyen, M.I. Jordan, and B. Sinopoli. *A kernel-based learning approach to ad hoc sensor network localization*. ACM Trans. Sen. Netw. 2005, 1(1): 134-152.

[9] B.R. Paredes, A. Argyriou, N. Berthouze and M. Pontil. *Exploiting Unrelated Tasks In Multi-Task Learning*[J]. Journal of Machine Learning Research, 2012, 22: 951-959.

[10] G. Obozinski, B. Taskar, M.I. Jordan. *Mmulti-Task Feature Selection*. Statistics Department, UC Berkeley, 2006.

[11] A. Kumar, H. DauméIII. *Learning Task Grouping and Overlap in Multi-task Learning*. in Proceedings of the 29th International Conference on Machine Learning(ICML), 2012.

[12] G.-B. Huang, L. Chen and C.-K. Siew, "Universal Approximation Using Incremental Constructive Feedforward Networks with Random Hidden Nodes", *IEEE Transactions on Neural Networks*, vol. 17, no. 4, pp. 879-892, 2006.

[13] G.-B. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme Learning Machine for Regression and Multiclass Classification," *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics*, vol. 42, no. 2, pp. 513-529, 2012.

[14] G.CB. Huang, Q. Zhu, and C. Siew, et al. *Extreme learning machine: Theory and applications*. Neurocomputing, 2006, 70: 489-501.