Feature Relevance Network-Based Transfer Learning for Indoor Location Estimation

Ho-Sik Seok, Kyu-Baek Hwang, *Member, IEEE*, and Byoung-Tak Zhang, *Member, IEEE*

***Abstract*—We present a new machine learning framework for indoor location estimation. In many cases, locations could be easily estimated using various traditional positioning methods and conventional machine learning approaches based on signalling devices, e.g., access points (APs). When there exist environmental changes, however, such traditional methods cannot be employed due to data distribution change. In order to circumvent this difficulty, we introduce feature relevance network-based method, which focuses on interrelatedness among features. Feature relevance networks are connected graphs representing concurrency of the signalling devices such as APs. In the newly created relevance network, a test instance and the prototype of a location are expanded until convergence. The expansion cost corresponds to distance between the test instance and the prototype. Unlike other methods, our model is nonparametric making no assumptions about signal distributions. The proposed method is applied to the 2007 IEEE International Conference on Data Mining Data Mining Contest Task #2 (transfer learning), which is a typical example situation where the training and test datasets have been gathered during different periods. Using the proposed method, we accomplish the estimation accuracy of 0.3238, which is better than the best result of the contest.**

***Index Terms*—Feature relevance networks, indoor location estimation, transfer learning.**

I. Introduction

We INTRODUCE a novel algorithm for indoor location estimation with varying distributions. Unlike most machine learning problems where distributions of training and test data are assumed to be the same, 2007 IEEE International Conference on Data Mining (ICDM) Data Mining Contest (DMC) Task #2 (transfer learning) [1] presents a challenging situation where the training and test instances have been gathered during separate periods. As a result, the estimation framework obtained from traditional positioning methods such as, time difference of arrival, and roundtrip time of flight [2] cannot be used properly. In addition, conventional machine learning methods are not readily applicable due to the same reason [3]. As a result, the accuracy of the location estimation is worsened from 0.8227 (without distribution change) to 0.3223 (with distribution change).

Most previous approaches in these situations have tried to transform the parameters of statistically learned models [4], [5]. In the case of 2007 IEEE ICDM DMC Task #2, radio signal strength (RSS) is not reliable and the number of available data instances from the changed distribution is too small to robustly transform the learned parameters. Therefore, it is nearly impossible to deploy parameter-transfer-based approaches in a smooth manner.

In order to alleviate this difficulty, we developed a novel method built on interrelatedness among features. Intuitively, a good feature representation is crucial for a successful domain adaptation [6]. But distributional changes make it difficult to find a proper representation of features. Our method focuses on interfeature relationship to construct plausible feature representation, which is expected to be resilient to distributional changes of feature values. The core assumption is that the nearer the access points (APs) are located, the more probable they are observed simultaneously. Based on this expectation, we search for the AP pairs, highly adjacent to each other. Such AP pairs comprise edges of a graph structure and then, the problem space is reconstructed using it. When a new test instance is given, it is mapped onto the new problem space and expanded. More precisely, the test instance and the prototype of a class are expanded together until convergence. After that, the most plausible location is chosen. In the demanding task of 2007 IEEE ICDM DMC Task #2 where training and test datasets are obtained from different distributions and there are too few training instances from the test environment, our method shows superior results compared to those from the previous approaches. We achieve the accuracy of 0.5831 (upper bound) and 0.3238 (with the current setting, which is better than the best performance achievement ever).

The structure of the paper is as follows. In Section II, we review the related works. Section III explains the 2007 IEEE ICDM DMC problem and the proposed method. Section IV presents experimental results. In Section V, we discuss the characteristics of the proposed approach. Section VI summarizes the paper.

II. Related Works

Recently, transfer learning is receiving much attention. Transfer learning emphasizes knowledge transfer across domains, tasks, and distributions that are similar but not the same. In a

Manuscript received January 3, 2010; revised June 16, 2010; accepted August 16, 2010. Date of publication October 21, 2010; date of current version August 19, 2011. This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MEST) (No. 2010-0017734), the IT R&D Program of MKE/KEIT (KI002138, MARS), the Industrial Strategic Technology Development Program (10035348) funded by the Korean government (MKE), and the BK21-IT Program. The ICT at Seoul National University provides research facilities for this study. Kyu-Baek Hwang was supported by the Soongsil University Research Fund. This paper was recommended by Associate Editor H. Liu.

H.-S. Seok and B.-T. Zhang are with the School of Computer Science and Engineering, Seoul National University, Seoul 151-744, Korea (e-mail: hsseok@bi.snu.ac.kr; btzhang@bi.snu.ac.kr).

K.-B. Hwang is with the School of Computing, Soongsil University, Seoul 156-743, Korea (e-mail: kbwhang@ssu.ac.kr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMCC.2010.2076277

survey paper [5], the authors summarized different settings of transfer learning as inductive, transductive, and unsupervised transfer learning. Inductive transfer learning aims to exploit unlabeled instances in classification tasks. In the self-taught learning framework, e.g., the authors present an approach which learns a succinct, higher-level feature representation of inputs using unlabeled data [9]. It is similar to our approach as it aims at finding a higher level feature representation relying on both labeled and unlabeled instances. However, we assume an underlying environmentally invariant interfeature relatedness and design new conceptual features based on this. In transductive transfer learning setting, a lot of labeled instances in source domain are available while no labeled ones in target domain are available. In [10], both in-domain and out-of-domain data are treated as if they are drawn from a mixture of distributions. In the location-estimation problem, however, it is difficult to obtain such distribution due to the lack of data in target domain. In unsupervised transfer learning setting, researchers try to reduce the dimensionality with the help of related prior knowledge from other classes in the same type of concept [11]. We search for useful features in expanded feature space without explicit prior knowledge.

Contrary to the previous researches, we focus on the representation. The method of explicitly minimizing difference between source and target domains in [12] seems similar to our approach. However, we concentrate on building feature relevance networks in order to infer missing features. If one uses parametric learning, methods in [13] and [14] would be useful. After estimating an initial distribution $\mathcal{D}_l$, the authors revise the model for a different distribution $\mathcal{D}_u$ of test data or produce resampling weights by matching the distributions between training and test sets in a feature space. We do not try to estimate parameters of an assumed distribution. As in [15], our method builds a common representation space for multiple datasets in order to reinforce newly created feature space. In [16], the authors assume that features themselves have metafeatures that are predictive of their relevance to the task, and model their relevance as a function of the metafeatures using hyperparameters. The relationship among features regardless of the relevance to location is of primary importance to our method. Some researchers try to discover shared parameters or priors between source and target domain modes [17], [18]. Because we do not assume a parametric model, these approaches have little relevance to our method.

The proposed method seeks artificial feature combinations suitable for a given problem. In terms of local consistency, this is similar to [19]. The difference is that our method does not assume a globally stable state, rather tries to figure out clusters of neighboring locations. A framework for predictive structures can be obtained from multiple tasks and there might be also a common framework for estimation location [20]. The idea of searching for feature similarity is also interesting, although it is uncertain how the idea could be applied on transfer learning.

Contrary to the traditional common feature representation space [21], we estimate expansion cost based on the assumption that the invariant relationship among features could be used as distance measure. In this sense, the proposed method could be interpreted as employing the manifold assumption [22]. Similar to [23], we derive local manifold structures from feature relevance networks and the distance between two locations is measured using these structures. Because given instances are compared to prototypes, our method is more similar to proximity-based representation spaces method, not to kernel methods.

As well explained in [24], existing methods for manifold learning have several shortcomings such as small sample size problem and information loss problem. Because the proposed method extracts interrelatedness from observed instances regardless of their label, it is less limited by the size of labeled data. Our method is also similar to neighborhood detection in [25]. By focusing on reconstruction of the local state vectors at some specified sites from measured data, the authors try to determine the spatiotemporal neighboring region or sites. We propose a method to determine the necessary dimensions of the reconstruction vector, although we do not consider the temporal characteristics in data. In [26], the idea of using frequency for intracluster similarity and intercluster difference is similar to ours. However, we do not preselect the number of clusters.

The indoor location estimation has been researched based on various technologies. In [27], the idea of forming the region of confidence (RoC) is similar to our method but the authors obtain the RoC using geometric properties. Our method does not consider the geometric properties and is solely based on feature relatedness. In [28], the authors address the location estimation in an 802.11 wireless LAN environment. The authors perform a kernel-based transformation of signal and physical spaces to capture the nonlinear relationship between signals and locations. This approach does not provide explicit solutions for transfer learning. In [29], the authors address a problem similar to the task of this paper. A radio map is constructed by calibrating signal-strength values in offline phase. However, this method uses the luxury of neighboring reference points not available in our problem setting.

III. Indoor Location Estimation and the Proposed Method

A. Indoor Location Estimation Problem

The 2007 DMC, sponsored by the IEEE ICDM, provides the first realistic public benchmark dataset for indoor location estimation using RSS that a client device receives from a set of Wi-Fi APs. The dataset was collected from an 145.5 × 37.5 m academic building at the Hong Kong University of Science and Technology where the location is divided into a grid of 247 units. The contest focused on two tasks: indoor location estimation and transferring knowledge for indoor location estimation. We focus on transferring knowledge for the indoor location estimation task.

Let $\mathcal{Z}$ ($|\mathcal{Z}| = 3128$) be the set of test instances. Each point $\mathbf{z}_i$ ($\mathbf{z}_i \in \mathcal{Z}$) is expected to belong to one of the M locations ($M = 247$). We will use c_i to denote the location associated to $\mathbf{z}_i$. Our goal is to estimate c_i for $\mathbf{z}_i$. Basically, the contest is to predict locations on the basis of the RSS values received from the Wi-Fi APs. In this paper, training instances are denoted as $\mathcal{T}$ ($\mathcal{T} = \mathcal{T}_L \cup \mathcal{T}_U$). $\mathcal{T}_L$ denotes labeled instances

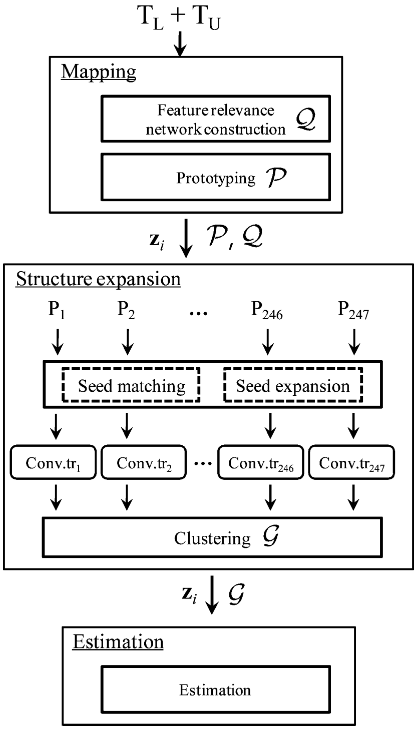

Fig. 1. Overview of the proposed method. The proposed method is composed of mapping, structure expansion, and estimation steps. At the mapping step, environmentally invariant properties are obtained. The feature relevance network $\mathcal{Q}$ is constructed from $\mathcal{T}$. P_r ($P_r \in P$), the prototype of location r is computed from $\mathcal{T}_L$. At the structure expansion step, a test instance and P_r are expanded until converging to the same structure based on $\mathcal{Q}$. $Conv.tr_i$ denotes the ith converged tree. Based on the accumulated cost of the expansion, some locations are selected as a member of the neighboring group, $\mathcal{G}$. At the estimation step, the location for the given test instance is estimated from $\mathcal{G}$.

($|\mathcal{T}_L| = 621$), and $\mathcal{T}_U$ denotes unlabeled instances ($|\mathcal{T}_U| = 1701$). Each $\mathbf{t}_l$ ($\mathbf{t}_l \in \mathcal{T}_L$) is denoted as $\mathbf{t}_l = \langle \mathbf{a}_l, c_l \rangle$ and $\mathbf{a}_l = \langle a_{l,1}, \ldots, a_{l,99} \rangle$ ($a_{l,i}$ means the RSS value from the ith AP of the lth instance). Each $\mathbf{t}_u (\mathbf{t}_u \in \mathcal{T}_U)$ is represented as $\mathbf{t}_u = \langle \mathbf{a}_u \rangle$ and $\mathbf{a}_u = \langle a_{u,1}, \ldots, a_{u,99} \rangle$. The distributions of $\mathcal{Z}$ and $\mathcal{T}$ are different, so the traditional assumption that training and test examples are sampled from the same distribution does not hold in this task. A few reference points with location label from the test environment are provided as landmarks, i.e., $\mathcal{D}$ ($|\mathcal{D}| = 52$), to assist the estimation procedure. Difficulties of the estimation problem come from the following two facts. The first is that properties of some locations are too similar. The second is that the given landmark instances are scarce. Due to the lack in landmark data, one cannot build a reliable parameter transfer model for each location.

B. Proposed Method

Fig. 1 illustrates the overall algorithm. The proposed method is composed of mapping, structure expansion, and estimation steps. In the mapping step, the feature relevance network $\mathcal{Q}$ is obtained from $\mathcal{T}$. $\mathcal{Q}$ consists of AP pairs supposed to be adjacent. The adjacency of APs is inferred based on their co-occurrence. In addition, the prototype for each location ($\mathcal{P} = \{P_1, \ldots, P_{247}\}$) is obtained using $\mathcal{T}_L$. In the structure expansion step, each prototype and a test instance are expanded until they are converged to the same structure. At the estimation step, c_i is estimated from a cluster of neighboring locations ($\mathcal{G}$).

TABLE I
MAPPING INTO A NEW PROBLEM SPACE

```
Step 1 Generate all possible AP pairs from the 99 APs.
       ($E_{j,k}$: $j$=1,...,98, $k$=$j+1$,...,99)
For each $E_{j,k}$
Step 2 $Freq_{j,k} = 0$.
Step 3 Count frequency $Freq_{j,k}$ from $\mathcal{T}$.
       $Freq_{j,k} = Freq_{j,k} + 1$,
       if $AP_j$ and $AP_k$ are observed concurrently.
Step 4 Calculate association value,
       $AValue_{j,k}$ for all AP pairs.
End of loop
For each location
Step 5 Make the prototype, $P_r$, for each location $r$ using $\mathcal{T}_L$.
End of loop
```

1) Mapping: The aim of this step is to find highly related AP pairs using the procedure in Table I. At first, the feature relevance network $\mathcal{Q}$ is obtained from $\mathcal{T}$. $\mathcal{Q}$ is defined as a triple $\mathcal{Q} = (\mathbf{X}, \mathbf{E}, \mathbf{W})$, where $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_{|X|}\}$, $\mathbf{E} = \{\mathbf{E}_1, \ldots, \mathbf{E}_{|E|}\}$, $\mathbf{W} = \{\mathbf{w}_1, \ldots, \mathbf{w}_{|E|}\}$ are the sets of vertices, edges, and weights, respectively ($\mathbf{x}_i$ means the ith AP, $|X|$ is the number of APs, and $|E|$ denotes the number of AP pairs, i.e., 4851). $\mathbf{W}$ denotes the relevance between two APs. $\mathbf{W}$ is defined using (1). If two APs are not observed concurrently, then $AValue$ is a predefined NULL value. Otherwise, $AValue$ is defined as follows.

Definition 1: Association value, $AValue_{j,k}$ of an edge, $E_{j,k}$ between APs *j* and *k* is defined as

$$AValue_{j,k} = 1 - \frac{\mathrm{Freq}_{j,k}}{\mathrm{MAX}_{\mathrm{Freq}} + 1} \tag{1}$$

where $\mathrm{Freq}_{j,k}$: frequency of the cases that AP j and k are observed concurrently, $\mathrm{MAX}_{\mathrm{Freq}}$: maximum frequency. $AValue$ is computed from $\mathcal{T}$.

In the mapping step, the set of prototypes $\mathcal{P}$ representing each location is also constructed from $\mathcal{T}_L$ as follows:

For all $\mathbf{t}_l$ whose location is r

$$P_r = \langle \bar{\mathbf{a}}_r, r \rangle \tag{2}$$

where $\bar{\mathbf{a}}_r = \langle \bar{a}_{r,1}, \ldots, \bar{a}_{r,99} \rangle$ ($\bar{a}_{r,h}$ is the averaged RSS value of the hth AP of the instances whose location is r).

2) Structure Expansion: The aim of this step is to form the candidate location set $\mathcal{G}$ for a given test instance. $\mathcal{G}$ corresponds to a group of locations adjacent to the given test instance. In order to select the candidate locations, the test instance, and the *i*th location's prototype are compared and expanded until convergence. The expansion cost is interpreted as distance. The structure expansion step is composed of *seed matching* and *seed expansion*.

Seed matching procedure is summarized in Table II. For a test instance $\mathbf{z}_i$ and a prototype P_r, the commonly observed APs in $\mathbf{z}_i$ and P_r constitute the set of common APs (S_{COM}). A structure connecting the APs in S_{COM} is constructed using a minimum spanning tree (MST) algorithm [7]. $EValue$ for this

TABLE II
SEED MATCHING AND EXPANSION

Step 1 Form S_{COM} (set of commonly observed APs),
S_{PL} (set of APs existing only in P_r and not in $\mathbf{z}_i$),
S_{TL} (set of APs existing only in $\mathbf{z}_i$ and not in P_r)
from test instance $\mathbf{z}_i$ and location prototype P_r.

Step 2 (**Seed matching**) Obtain set of $E_{j,k}$'s (E_{COM})
connecting the elements of S_{COM} based on $AValue_{j,k}$
using an MST algorithm.

Step 3 (**Seed expansion**) Obtain set of $E_{j,k}$'s (E_{EXP})
connecting S_{COM}, S_{TL}, S_{PL} based on $AValue_{j,k}$
using an MST algorithm.

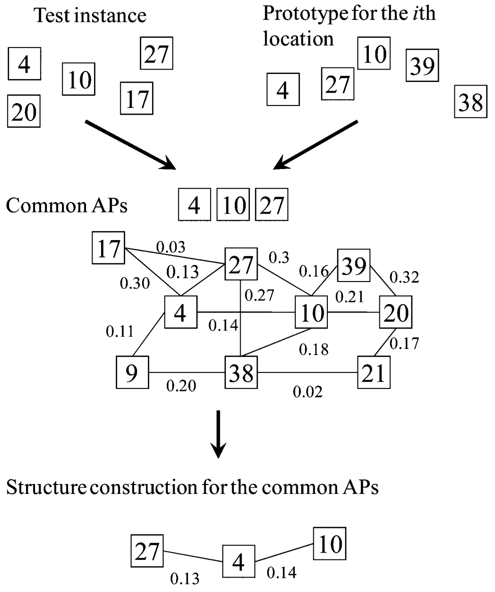

Fig. 2. Illustration of seed matching. At this step, the common APs observed in a test instance and the prototype of the *i*th location are processed. In the figure, AP_4, AP_{10}, and AP_{27} compose of S_{COM}. Based on the $AValue$s, the edges with the lowest $EValue$, $E_{4,27}$ and $E_{4,10}$ are selected and comprise the connecting structure S_{COM}.

structure is computed as follows:

$$EValue_r = \sum_{E_{j,k} \in E_{\mathrm{COM}}} AValue_{j,k} \tag{3}$$

where E_{COM} is the structure for S_{COM} obtained by an MST algorithm.

In the example illustrated in Fig. 2, the observed APs of a test instance are $\{4, 10, 17, 20, 27\}$, the observed APs of a prototype are $\{4, 10, 27, 38, 39\}$, and S_{COM} is $\{4, 10, 27\}$. Then, edges with lower $AValue$ are selected. Therefore, $E_{\mathrm{COM}} = \{E_{4,10}, E_{4,27}\}$.

At the seed expansion step, a test instance and the location's prototype are expanded by adding unobserved APs until convergence. The aim of this step is to find a set of edges, E_{EXP} connecting APs in S_{TL}, S_{PL}, and S_{COM} (where S_{TL} is the set of APs observed only in a test instance. S_{PL} is the set of APs observed only in the ith prototype) using the procedure summarized in Table II. This step is also based on an MST algorithm. The edges in E_{EXP} are used to compute $LValue$. This is based on the assumption that if an AP is not observed due to an environmental change, it could be guessed from nearby APs using the feature relevance network.

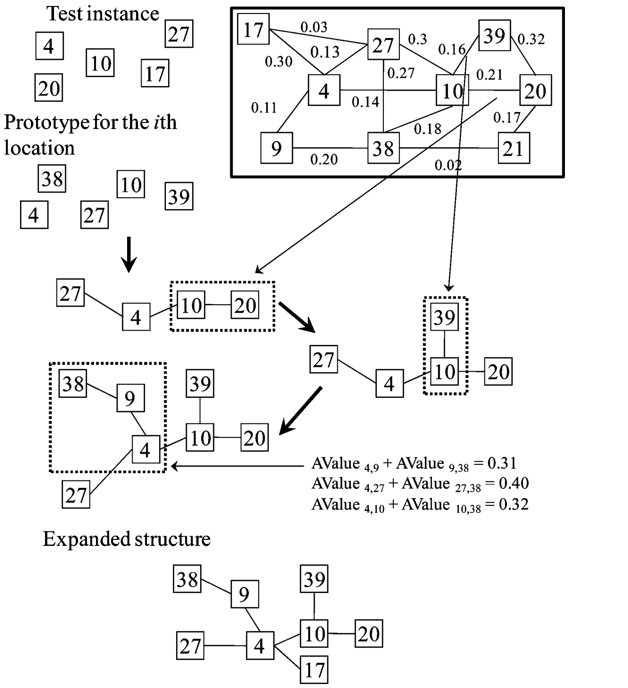

Fig. 3. Exemplary seed expansion procedure for missing APs. E_{EXP} connecting APs in a test instance and the *r*th prototype is constructed by adding a new AP "9".

$$LValue_r = \sum_{E_{j,k} \in E_{\mathrm{EXP}}} AValue_{j,k}. \tag{4}$$

In Fig. 3, the seed expansion step is illustrated using the same example in Fig. 2, where $S_{\mathrm{COM}} = \{4, 10, 27\}$, $S_{\mathrm{TL}} = \{17, 20\}$, and $S_{\mathrm{PL}} = \{38, 39\}$. At the seed expansion step, a structure connecting APs $\{4, 10, 17, 20, 27, 38, 39\}$ is constructed by adding a new AP "9". After the execution of the MST algorithm, $E_{\mathrm{EXP}} = \{E_{9,38}, E_{4,9}, E_{4,27}, E_{4,10}, E_{4,17}, E_{10,39}, E_{10,20}\}$.

After the structure expansion step, $EValue$ and $LValue$ are obtained for each pair of the test instance and the *i*th location's prototype. $EValue$ represents the degree of relatedness between the test instance and the prototype. $LValue$ corresponds to the accumulated expansion cost. Because two observations obtained at different periods at the same location would have more common APs and lower expansion cost than observations from different locations, the ratio of these two values can be used as a distance. This idea is represented as follows:

$$RScore_r = \frac{LValue_r}{EValue_r} (r = 1, \ldots, 247). \tag{5}$$

Now, the locations supposed to be located more adjacent to the given test instance can be recommended based on $RScore$ values (5). These locations will be denoted as $\mathcal{G}$. The other locations will be represented as $\mathcal{G}'$. $\mathcal{G}$ and $\mathcal{G}'$ can be obtained after sorting the 247 prototypes according to the their $RScore$s, then dividing them by a proper criterion.

In order to select the proper criterion, we devise two operators γ and ψ as follows:

$$\gamma_r = RScore_r - RScore_{r-1}. \tag{6}$$

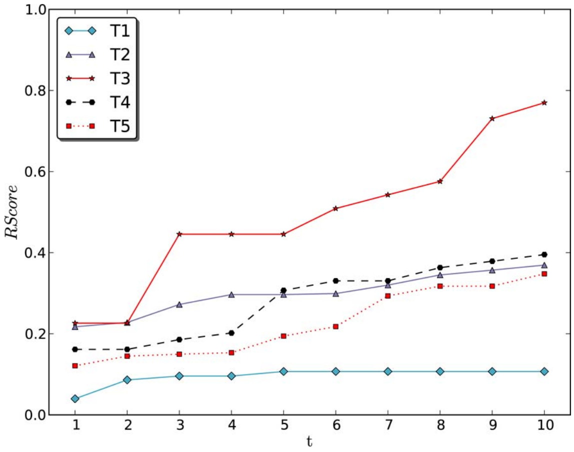

Fig. 4. *RScore* curves. This chart shows variations in *RScores* of five test instances (T1, T2,. . .,T5). There hardly seems a clear cut point. Here, t means ten prototypes.

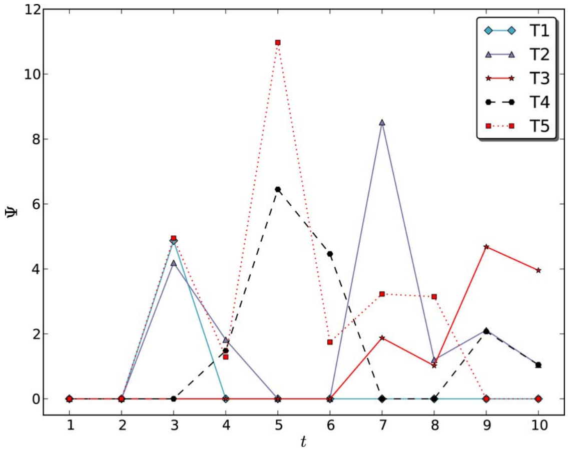

Fig. 5. ψ curves for Fig. 4. This figure shows ψ curves for the instances in Fig. 4. By introducing γ and ψ measures, there are fluctuations in the curves. A set of interim criteria is selected based on the fluctuations. If the denominator is 0, the corresponding ψ value is defined as 0.

$$\psi_r = \left\{ \begin{array}{ll} \dfrac{\gamma_{r+1}}{\gamma_r}, & \text{if } \gamma_{r+1} > \gamma_r \\ \dfrac{\gamma_r}{\gamma_{r+1}}, & \text{otherwise} \end{array} \right\}. \quad (7)$$

If one selects too many candidates for $\mathcal{G}$ in order to contain c_i for the given test instance, the estimation accuracy would be worsened. On the contrary, if one selects too few candidates for $\mathcal{G}$, then the estimation accuracy would also be worsened because c_i would not be included in $\mathcal{G}$. A criterion capable of balancing the number of elements in $\mathcal{G}$ is searched using the procedure in Table III.

TABLE III
RECOMMENDATION

```
For whole $RScore$ values of a given test instance $\mathbf{z}_i$
Step 1 Compute Eq. 6 for all $RScore$ values.
Step 2 Compute $\psi$ values based on Eq. 7.
Step 3 Select set of interim criteria $\eta_{interim}$ based on $\psi$ values
       showing significant variation.
Step 4 Compute $F_{INC} = \sum_{k=1}^{|D|} I_k$ by classifying
       landmark data based on $\mathcal{G}$ for all $\eta_{interim}$.
       (if the $k$th landmark instance is classified correctly,
       $I_k = 1$ else $I_k = 0$)
Step 5 Select the final criterion
       $\eta_{final} = \mathrm{argmax}_{\eta_{interim}} F_{INC}$
```

Figs. 4 and 5 show $RScore$s and ψ values for some test instances. As shown in Fig. 4, it is difficult to find a meaningful variation in the $RScore$ curves. However, the curves in Fig. 5 show prominent changes in ψ values. A set of interim criteria is selected based on the fluctuations in ψ curves. For the landmark instances, a set of interim criteria η_{interim} is selected based on these ψ values. Each element of η_{interim} is determined as ψ_l where $\psi_l \geq \psi_{\text{init}}$ (ψ_{init} is a constant determined using landmark instances). For each element in η_{interim}, η_{final} is a value maximizing F_{INC} in Table III.

After constructing $\mathcal{G}$, the RSS values of some prototypes are modified. If a location is adjacent to a known location (i.e., landmark data), the amount of RSS value change of the location is likely to be similar to the RSS value change of the known location. This idea is incorporated to adjust the RSS values of the APs of the prototypes in $\mathcal{G}$.

3) Estimation: The location of a test instance is estimated at this step based on the weighted p-norm distance. For a test instance $\mathbf{z}_i$ and $\mathcal{G}$, location c_i^* for $\mathbf{z}_i$ is estimated as follows:

$$c_i^* \leftarrow \mathrm{argmin}_{r \in \mathcal{G}} \mathrm{dist}(P_r, \mathbf{z}_i) \quad (8)$$

$$\mathrm{dist}(P_r, \mathbf{z}_i) = \sum_{d=1}^{99} (w_d \cdot |a_{r,d} - z_{i,d}|^p)^{\frac{1}{p}} \quad (9)$$

where $p = 1.7$ (p was optimized using landmark instances), $w_d = (|a_{r,d} - z_{i,d}|)/g$ (if $a_{r,d}$ or $z_{i,d}$ is unobserved, then $w_d = 0$, $g = \sum_{d=1}^{99} z_{i,d}$ for the observed $z'_{i,d}s$).

IV. EXPERIMENTAL RESULTS

In Fig. 6, the feature relevance network built from $\mathcal{T}$ is shown. Here, linked APs are simultaneously observed pairs and they are assumed to be adjacent. Interestingly, we could observe some hub APs and conjecture that the feature relevance network assumes a complex network structure whose degree distribution can be represented by the power law.

A. Quality of Candidate Recommendation

After the recommendation, the cluster $\mathcal{G}$ for a $\mathbf{z}_i$ is obtained. An example $\mathcal{G}$ for a test case where $c_i = 203$ is illustrated in Fig. 7. As shown in Fig. 7, the recommended locations are likely to be adjacent to the target location. For more comprehensive analysis, we present two concepts describing the degree of adjacency (*AD*) and hit ratio, respectively. The *AD* corresponds to the density of $\mathcal{G}$, i.e., the average distance between the locations in $\mathcal{G}$. It was calculated using the offered location map. We assign coordinates to each location by setting the left-bottom location as $(0, 0)$ and the distance between adjacent locations as one. This map was *only* used for computing the *AD* and had not been used in the learning and estimation processes. Then, AD_i of the

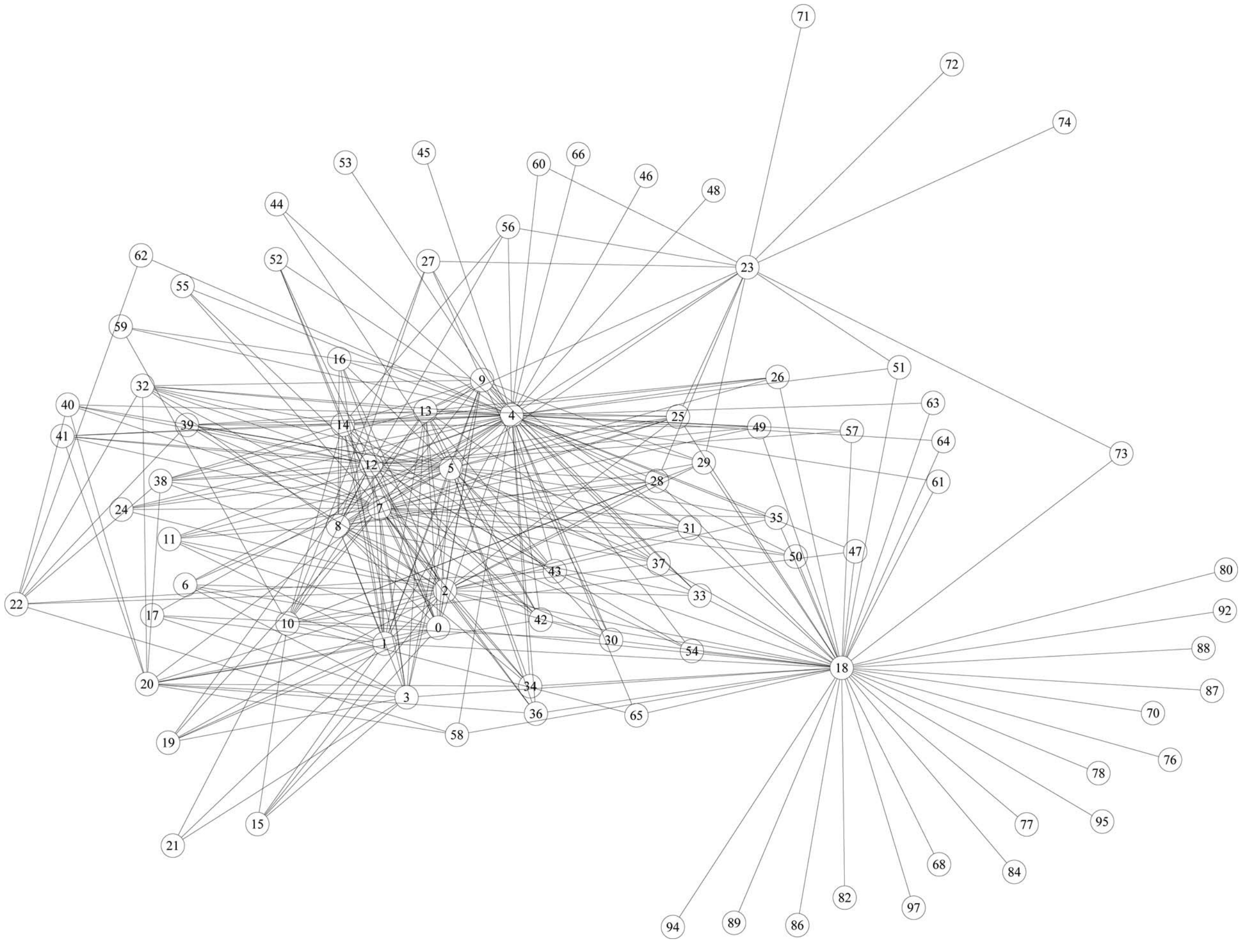

Fig. 6. Generated relevance network. This figure displays the relatedness between the 99 APs. Linked APs are more likely to be adjacent to each other than other pairs. Here, weights or adjacencies are not shown. These links represent all the concurrency among the APs estimated from $\mathcal{T}$.

Target location: 203

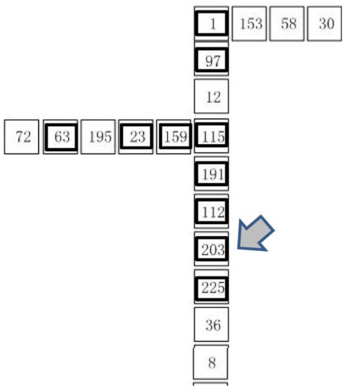

Fig. 7. Recommended locations for a test instance whose real location is 203. For location 203, locations $112, 191, 115, 225, 203, 159, 63, 97, 1$, and 23 are recommended by our method.

ith location is computed using the following equation

$$AD_i = \frac{\sqrt{\sum_k (\mathbf{x}_k - o_i)^2}}{\#\text{ of elements in }\mathcal{G}}. \quad (10)$$

(o_i: the (artificial) center point of $\mathcal{G}$, $\mathbf{x}_k$: the kth element in $\mathcal{G}$).

TABLE IV
QUALITY OF CANDIDATE RECOMMENDATION

$Analysis$	$Value$
Average AD$_i$	2.671
Hit ratio (%)	69.7%
Maximum size of a cluster	10
Minimum size of a cluster	4
Average size of a cluster	6.773

The AD denotes the quality of a cluster. The higher is AD, the worse is the quality of $\mathcal{G}$. However, even low AD values could be useless if actual c_i is not included in $\mathcal{G}$. Hit ratio means the ratio of $\mathcal{G}$ containing actual c_i of each $\mathbf{z}_i$. With higher hit ratio, it is possible to obtain more higher accuracy.

The maximum size (it means the number of locations in $\mathcal{G}$) of $\mathcal{G}$ is 10. The minimum size is 4. The average size of a cluster is 6.773. As explained in Table IV, the recommendation capability is satisfiable. In other words, 69.7% of each c_i is contained in the recommended cluster, $\mathcal{G}$. Furthermore, the recommended locations are actually adjacent ones according to the average AD value of 2.671.

B. Prediction Performance

In order to evaluate the prediction performance, we carry out two experiments. In the ideal setting, the group of optimal

TABLE V
EXPERIMENTAL RESULTS. RANK1 AND RANK2 ARE THE BEST AND THE SECOND ESTIMATION RESULTS AT 2007 IEEE ICDM DMC CONTEST [8]

ID	$Accuracy$
Proposed method (upper bound)	0.5831
Proposed method (using Table III)	0.3238
$Rank1$	0.3223
$Rank2$	0.3149
Naive Bayes	0.1992
Recommendation rule only	0.191
Random forest	0.1781
kNN (k=1)	0.1771
kNN (k=2)	0.1640
kNN (k=3)	0.1550
Neural networks	0.1544
Classifier only	0.0

adjacent locations $\mathcal{G}^o$ is determined through reference to the actual location of each test instance in order to show the upper bound in the ideal case with the given condition. In the other case, $\mathcal{G}$ is determined based on the procedure in Table III.

In each case, the accuracy is determined by the following definition.

Definition 2

$$\text{Accuracy} = \frac{\sharp \text{ of correct predictions}}{\sharp \text{ of all test instances}}.$$

In Table V, the estimation results are presented. With the proposed algorithm, the adjacent locations, $\mathcal{G}$ for each $\mathbf{z}_i$, are recommended during the recommendation step. Therefore, instances with similar feature vectors are given to a classifier. As a result, it is possible to achieve relatively higher accuracy with simple classifier. For 69.7% of hit ratio, the accuracy is 0.3238. This performance is better than the first ranked result of the 2007 IEEE ICDM DMC Task #2 and the conventional machine learning algorithms such as neural networks, kNN, random forest, and naive Bayes (algorithms implemented in WEKA 3.4.11 have been employed for the comparison). The broadly used algorithms do not provide promising results. Because other classifiers do not have the transfer learning capability, the poor performance is expected. These results are presented to show that the given task cannot be solved well with conventional methods.

The winners of the ICMD 07 contest employed the Minkowski distance and NUM (nearest unlike neighbor) distance. The winners computed the initial class centers and assigned a class label to the unlabelled instances. After that, new class centers were computed using the newly labeled instance and the given labeled training instances. Finally, the Minkowski distance weights for each class were assigned. Contrary to the ICDM 07 contest winner, our method focuses on relational properties among APs. The relational properties are represented as relevance network. For a given test instance, candidate locations for this test instance are selected based on feature relevance network. Because the proposed method infers the likelihood of a feature change, our method can be applied to other domains with missing or changing attributes. This is one novelty of our method.

In order to present the upper limit of the proposed method in the current setting (with the current relevance network), we built an optimal cluster $\mathcal{G}^o$, which always includes the actual location of $\mathbf{z}_i$. In this case, the accuracy is 0.5831. The difference between the ideal case and the other case is due to the recommendation rule. The estimation accuracy in the case of using Table III can be improved further by calibrating the recommendation rule. With the current experimental setting, we have only 48 samples (discarding repeated locations) for estimating the characteristics of the test environment. With cues corresponding 19.4% of the target locations, there is much room for the improvement. By calibrating the relevance network with more instances, it is possible to improve the accuracy to the upper bound.

In order to verify the validity of the estimation after recommendation, we also compared the accuracy at each stage. The accuracy with *recommendation rule only* means the case where the location with the lowest $RScore_i$ is determined as c_i for the given $\mathbf{z}_i$. The accuracy with *classifier only* means the case where the classifier of (9) is used for estimation without forming $\mathcal{G}$. Because the distance measure in (9) ignores unobserved APs, *classifier only* estimation result is inferior to other conventional classifier. However, our method improves the estimation accuracy significantly by combining the recommendation procedure and a simple distance measure.

V. DISCUSSION

In spite of the difficulties of the indoor location estimation problem, the proposed method achieves meaningful improvement. The idea behind the improvements is as follows: the difficulty due to distribution change could be overcome by building environmentally invariant properties. We represented co-occurrences among APs as a simple graph structure. Using this graph space, we obtained a method enabling location estimation despite the distribution change. The difficulty due to too few training examples was circumvented by giving up construction of location models. We could not build reliable location models with the given scarce labeled training data. Furthermore, even if there exist location models, we could not transfer the parameters due to the lack of labeled training data from the test environment. By focusing on feature relatedness, we got rid of the need for a model per a location.

The problem of insufficient labeled data always harasses researchers. In the proposed method, the effect of insufficient labeled data is a decline of the quality of prototypes. As a consequence, the seed matching/expansion steps are deteriorated. However, the proposed method has a unique strength in transfer learning. Our method focuses on the invariant relationship among features. Therefore, one could extract more meaningful invariant relations with plentiful unlabeled data and these upgraded relations result in a more informative relevance network. As a result, plentiful unlabeled data can offset the deteriorating effect of insufficient labeled data. With more unlabeled data from the test environment, we could secure another cue for estimating the RSS value change. Therefore, it is possible to enhance the estimation result further.

Another virtue of the proposed method is in its recommendation step. Through recommendation, we formed a cluster of

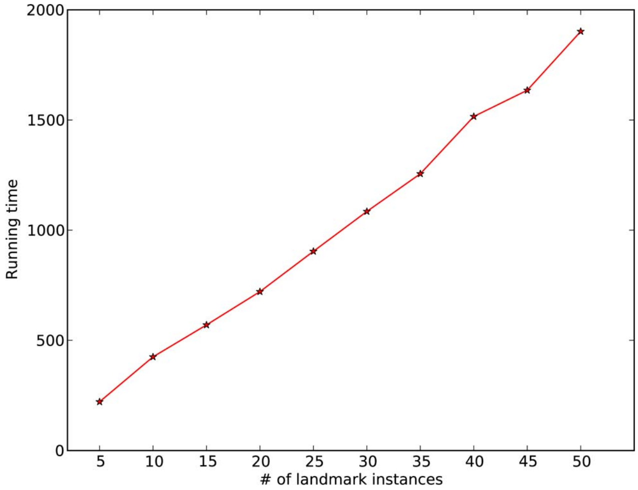

Fig. 8. Running time for determining ψ value. This chart shows the running time along the number of landmark instances. The time is measured in seconds. The running environment: Intel core 2 CPU 6600 2.40 GHz and 2.39 GHz with 2.00GB RAM.

adjacent locations. By focusing on these neighboring locations, we were able to estimate the unknown location more successfully. Our method is suitable for such tasks in which the number of classes is very large, the independency among classes is weak, and the number of training instances per class is small. The indoor location estimation problem is a typical example for this kind of domain, thus, we succeeded in introducing a useful estimation method.

One of the shortcomings of the proposed method is its computational cost. The proposed method builds a relevance matrix of fixed size $F \times F$ (F: the number of features). Because the relevance network is obtained from co-observation of the APs, the large size of training data is not a serious computational burden. The heaviest computational burden comes from the process of fixing ψ values based on $RScore$ values. Although the running time for determining ψ values is very high (as shown in Fig. 8), there is a margin for further optimization. The running time is proportional to the number of landmark instances and the computation time of each ψ value could be improved with more instances. In the employed MST algorithm, the running time per location is $l \times O(m\log n)$ (l: the number of locations, i.e., 247; m: the number of APs; and n: the number of edges, i.e., 4851). In the equation $O(m\log n)$, m is fixed and n can be reduced by employing some dimensionality reduction techniques such as multidimensional clustering (MDS). With enough data showing distribution change (labeled or unlabeled ones), it is possible to eliminate less informative elements from the the relevance network based on the co-occurrence. In addition, more informative prototypes (represented by less features) could be obtained and these refined prototypes could contribute to pruning the edges. As a result, n in $O(m\log n)$ could become smaller and computational burden could be relieved given a sufficient number of informative instances.

A feature relevance network represents how closely two features are related to each other. Therefore, the proposed method can be extended to other problems if a few preconditions are met. The preconditions are as follows: 1) there should be a reasonable basis to support the assumption that features are interrelated in a pairwise manner—in the case of indoor location estimation, it is highly probable that the property of adjacent APs would be similar and 2) the unvaried relation among features is easy to guess or represent—in the case of indoor location estimation, adjacent APs can easily be represented by their observation values, which should be highly correlated to one another. There could be lots of domains satisfying these preconditions. Transfer learning of object classes [30] is a typical example. In the object recognition domain, the contour corresponds to an AP in indoor location estimation and the relations among contours are invariant. In this case, the relations among contours can be represented as a feature relevance network. As a result, it is possible to recognize an object even some contours are missing based on the relevance network.

VI. Conclusion

We proposed a novel estimation algorithm for data with varying distributions. The proposed method builds a feature relevance network based on invariant properties and given instances are expanded on the feature relevance network using an MST algorithm. For a given test instance, several candidates constitute a cluster based on the expansion cost, i.e., distance. Because the cluster is composed of adjacent candidates in a newly created problem space, we could achieve better classification result using a relatively simple classifier. The proposed method performed better than the first ranked result at 2007 IEEE ICDM DMC Task #2 with the same setting. By the simple idea of focusing on constant properties regardless of environmental changes, the estimation accuracy was improved. Although the algorithm imposes an additional computational burden, the experimental results show that it is attractive for tasks with varying distributions.

References

[1] Q. Yang, S. J. Pan, and V. W. Zheng, “WiFi-based indoor location estimation: An overview of the 2007 IEEE ICDM Contest,” *IEEE Intell. Syst.*, vol. 23, no. 1, pp. 8–13, Jan./Feb. 2008.

[2] H. Liu, H. Darabi, P. Banergee, and J. Liu, “Survey of wireless indoor positioning techniques and systems,” *IEEE Trans. Syst., Man, Cybern., Part C: Appl. Rev.*, vol. 37, no. 6, pp. 1067–1080, Nov. 2007.

[3] R. Caruana, “Multitask learning,” *Mach. Learn.*, vol. 28, pp. 41–75, 1997.

[4] C. B. Do and A. Y. Ng, “Transfer learning for text classification,” in *Proc. Adv. Neural Inf. Process. Syst. 18*, 2006, pp. 299–306.

[5] S. J. Pan and Q. Yang. (2008). *A Survey on Transfer Leaning* [Online]. Available: http://www.cse.ust.hk/ sinnopan/Survey TL.htm

[6] S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira, “Analysis of representation for domain adaption,” in *Proc. Adv. Neural Inf. Process. 19*, 2007, pp. 137–144.

[7] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*. Cambridge, MA: MIT Press, 2001, pp. 595–601.

[8] [Online]. Available: http://www.cse.ust.hk/~qyang/ICDMDMC07/

[9] R. Raina, A. Battle, H. Lee, B. Packer, and A. Y. Ng, “Self-taught learning: Transfer learning from unlabeled data,” in *Proc. 24th Int. Conf. Mach. Learn.*, Jun. 2007, pp. 759–766.

[10] H. Daumé III and D. Marcu, “Domain adaptation for statistical classifiers,” *J. Artif. Intell. Res.*, vol. 26, pp. 101–126, 2006.

[11] Z. Wang, Y. Song, and C. Zhang, “Transferred dimensionality reduction,” in *Proc. Eur. Conf. Mach. Learn. Knowl. Discov. Databases*, Sep. 2008, pp. 550–565.

[12] S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira, "Analysis of representations for domain adaptation," in *Proc. Adv. Neural Inf. Proc. 19*, 2007, pp. 137–144.

[13] W. Dai, G.-R. Xue, Q. Yang, and Y. Yu, "Transferring naive Bayes classifiers for text classification," in *Proc. 22rd AAAI Conf. Artif. Intell.*, Jul. 2007, pp. 540–545.

[14] J. Huang, A. Smola, A. Gretton, K. M. Borgwardt, and B. Schölkopf, "Correcting sample selection bias by unlabeled data," in *Proc. Adv. Neural Inf. Syst. 19*, 2007, pp. 601–608.

[15] T. Jebara, "Multi-task feature and kernel selection for SVMs," presented at the 21th Int. Conf. Mach. Learn., Banff, AB, Canada, Jul. 2004.

[16] S.-I. Lee, V. Chatalbashev, D. Vickrey, and D. Koller, "Learning a meta-level prior for feature relevance from multiple related tasks," in *Proc. 24th Int. Conf. Mach. Learn.*, Jun. 2007, pp. 489–496.

[17] N. D. Lawrence and J. C. Platt, "Learning to learn with the informative vector machine," in *Proc. 21st Int. Conf. Mach. Learn.*, Jul. 2004, pp. 512–519.

[18] T. Evgeniou and M. Pontil, "Regularized multi-task learning," in *Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, Aug. 2004, pp. 109–117.

[19] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," in *Proc. Adv. Neural Inf. Syst. 16*, 2004, pp. 321–328.

[20] R. K. Ando and T. Zhang, "A framework for learning predictive structures from multiple tasks and unlabeled data," *J. Mach. Learn. Res.*, vol. 6, pp. 1817–1853, 2005.

[21] E. Pekalska and R. P. W. Duin, "Beyond traditional kernels: Classification in two dissimilarity-based representation spaces," *IEEE Trans. Syst., Man Cybern., Part C: Appl. Rev.*, vol. 38, no. 6, pp. 729–744, Nov. 2008.

[22] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," in *Proc. Adv. Neural Inf. Process. Syst. 14*, 2001, pp. 585–591.

[23] D. Cai, X. Wang, and X. He, "Probabilistic dyadic data analysis with local and global consistency," in *Proc. 26th Annu. Int. Conf. Mach. Learn.*, 2009, pp. 105–112.

[24] C. Chen, J. Zhang, and R. Fleischer, "Distance approximating dimension reduction of Riemannian manifolds," *IEEE Trans. Syst., Man, Cybern., Part B: Cybern.*, vol. 40, no. 1, pp. 208–217, Feb. 2010.

[25] Y. Pan and S. A. Billings, "Neighborhood detection for the identification of spatiotemporal systems," *IEEE Trans. Syst., Man, Cybern., Part B: Cybern.*, vol. 38, no. 3, pp. 846–854, Jun. 2008.

[26] T. W. S. Chow, W. Piyang, and E. W. M. Ma, "A new feature selection scheme using a data distribution factor for unsupervised nominal data," *IEEE Trans. Syst., Man, Cybern., Part B: Cybern.*, vol. 38, no. 2, pp. 499–509, Apr. 2008.

[27] Y. Gwon, R. Jain, and T. Kawahara, "Robust indoor location estimation of stationary and mobile users," in *Proc. IEEE INFOCOM*, Mar. 2004, vol. 2, pp. 1734–1743.

[28] J. J. Pan, J. T. Kwok, Q. Yang, and Y. Chen, "Accurate and low-cost location estimation using kernels," in *Proc. 19th Int. Joint Conf. Artif. Intell.*, Jul. 2005, pp. 1366–1371.

[29] J. Yin and Q. Yang, "Learning adaptive temporal radio maps for signal-strength-based location estimation," *IEEE Trans. Mobile Comput.*, vol. 7, no. 7, pp. 869–883, Jul. 2008.

[30] G. Heitz, G. Elidan, and D. Koller, "Transfer learning of object classes: From cartoons to photographs," *NIPS 2005 Workshop - Inductive Transfer: 10 Years Later*, 2005.

Ho-Sik Seok received the B.S. and M.S. degrees from the School of Computer Science and Engineering, Seoul National University (SNU), Seoul, Korea, in 1999 and 2001, respectively. He is currently working toward the Ph.D. degree with the Biointelligence Laboratory, School of Computer Science and Engineering, SNU.

From July 2001 to June 2004, he was a Lecturer in the Department of Computer Science, Korea Military Academy (KMA) as an army officer. His current research interests include transfer learning and probabilistic graphical models of learning.

Kyu-Baek Hwang (M'07) received the B.S.E., M.S.E., and Ph.D. degrees all from the School of Computer Science and Engineering, College of Engineering, Seoul National University, Seoul, Korea.

He is currently an Assistant Professor in the School of Computing, Soongsil University, Seoul, Korea. Prior to joining Soongsil as a faculty member, he was a Postdoctoral Researcher at the Biointelligence Laboratory, Seoul National University, Seoul, Korea. During the Ph.D. study at Seoul National University, he visited the Children's Hospital Informatics Program, Harvard Medical School, Boston, MA, as a Short-term Visiting Scholar. His research interests include machine learning, natural language processing, and bioinformatics.

Byoung-Tak Zhang (M'95) received the B.S. and M.S. degrees in computer science and engineering from Seoul National University (SNU), Seoul, Korea, in 1986 and 1988, respectively, and the Ph.D. degree in computer science from the University of Bonn, Bonn, Germany, in 1992.

He is currently a Professor with the School of Computer Science and Engineering and the Graduate Programs in Bioinformatics, Brain Science, and Cognitive Science, SNU, and directs the Biointelligence Laboratory and the Center for Bioinformation Technology (CBIT). Prior to joining SNU, he was a Research Associate with the German National Research Center for Information Technology (GMD) from 1992 to 1995. From August 2003 to August 2004, he was a Visiting Professor with the Computer Science and Artificial Intelligence Laboratory (CSAIL), Massachusetts Institute of Technology (MIT), Cambridge. His research interests include probabilistic models of learning and evolution, biomolecular/DNA computing, and molecular learning/evolvable machines. He serves on the Editorial Board of Genetic Programming and Evolvable Machines and Applied Soft Computing.

Prof. Zhang serves as an Associate Editor for the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, Advances in Natural Computation, and Genomics and Informatics.