# Kernel-Based Positioning in Wireless Local Area Networks

Azadeh Kushki, *Student Member*, *IEEE*, Konstantinos N. Plataniotis, *Senior Member*, *IEEE*, and Anastasios N. Venetsanopoulos, *Fellow*, *IEEE*

**Abstract**—The recent proliferation of Location-Based Services (LBSs) has necessitated the development of effective indoor positioning solutions. In such a context, Wireless Local Area Network (WLAN) positioning is a particularly viable solution in terms of hardware and installation costs due to the ubiquity of WLAN infrastructures. This paper examines three aspects of the problem of indoor WLAN positioning using received signal strength (RSS). First, we show that, due to the variability of RSS features over space, a spatially localized positioning method leads to improved positioning results. Second, we explore the problem of access point (AP) selection for positioning and demonstrate the need for further research in this area. Third, we present a kernelized distance calculation algorithm for comparing RSS observations to RSS training records. Experimental results indicate that the proposed system leads to a 17 percent (0.56 m) improvement over the widely used K-nearest neighbor and histogram-based methods.

**Index Terms**—Location-dependent and sensitive mobile applications, applications of pattern recognition, nonparametric statistics, support services for mobile computing.

✦

## 1 INTRODUCTION

Since the advent of the Global Positioning System (GPS) [1] and the introduction of the E-911 mandate by the US Federal Communications Commission, positioning systems have been used to deliver Location-Based Services (LBSs) [2], [3] in outdoor environments. The primary role of such positioning systems is to estimate and report geographical information pertaining to their users for the purposes of management, enhancement, and personalization of services.

LBSs are of equal interest in indoor environments in a wide range of personal and commercial applications. These include location-based network management and security [4], [5], medicine and health care [5], [6], personalized information delivery [7], and context awareness [8]. Unfortunately, the level of localization accuracy needed in such indoor applications cannot be achieved by the existing cellular-based methods. Furthermore, coverage of the GPS system is limited in indoor environments and dense urban areas [9], [10]. In this light, a plethora of indoor positioning systems have been proposed, employing various technologies [10], [11] such as proximity sensors, infrared [12], radio frequency and ultrasonic badges [13], [14], Wireless Local Area Network (WLAN) radio signals [15], [16], and visual sensors [17]. Among these, visual surveillance and tracking is the most widely studied and is shown to provide highly accurate estimates [17]. Yet, this type of positioning requires installation of infrastructure and calibration of cameras, which can lead to hardware and labor overheads in pervasive deployments. An even more profound concern is that of privacy in situations where users choose not to be visually monitored.

With these concerns in mind, several works have considered positioning based on WLANs owing to their wide availability and ubiquitous coverage in large environments. Although a small subset of these works have explored the use of Time of Arrival (ToA) [18] and Angle of Arrival (AoA) [19] in the context of indoor WLANs, received signal strength (RSS) is generally the feature of choice for WLAN positioning. This is due to the fact that RSS measurements can be obtained relatively effortlessly and inexpensively without the need for additional hardware [20]. Moreover, RSS-based positioning is noninvasive, as all sensing tasks can be carried out on the mobile client (MC), eliminating the necessity for central processing.

## 2 RELATED WORK

The key technical challenge in WLAN positioning is the estimation of a rule $h(\cdot): \mathbb{R}^L \to \mathbb{R}^2$, relating the RSS vector $\mathbf{r}$ from $L$ access points (APs) to a spatial position in 2D Cartesian coordinates $\mathbf{p} = (x, y)$. In an ideal propagation medium (free-space), the received signal power falls off inversely proportional to the square of the distance to the transmitter. In real environments, however, the position-RSS dependency is much more complex because of severe multipath and shadowing conditions, non-line-of-sight (NLOS) propagation, and interference from other devices [21]. Several existing WLAN methods have aimed to use theoretical models to express these effects through a path loss model whose parameters are estimated based on training data [22]. Given an RSS measurement and this model, the distances from the MC to at least three APs are determined, and trilateration is used to obtain the MC position. The main limitations of such approaches are the

● *The authors are with the Edward S. Rogers Sr. Department of Electrical and Computer Engineering, University of Toronto, 10 King's College Road, Toronto, M5S 3G4, Ontario, Canada. E-mail: {azadeh, kostas, anv}@comm.utoronto.ca.*

dependence on prior topological information, assumption of isotropic RSS contours [23], [24], and invariance to receiver orientation [25].

As an alternative to model-based methods, the RSS-position relationship has been characterized implicitly using a training-based method known as *location fingerprinting*. Fingerprinting-based WLAN positioning is a two-step procedure. The first step, which is generally carried out offline, is the collection of training data for the purpose of obtaining a sufficient representation of spatiotemporal RSS properties in the given environment. This refers to choosing a set of training or *survey* locations and collecting RSS measurements over time at each point to constitute the fingerprint. The second step is initiated when a new RSS measurement from an MC is received during the online operation of the system. This step involves comparing the incoming observation to the training record (fingerprint) at each survey point. The position estimate is one or a combination of the survey locations whose fingerprint(s) most closely match(es) the observation.

There are four key challenges in fingerprinting:

1.  generation of fingerprints,
2.  preprocessing for reducing computational complexity and enhancing accuracy,
3.  selection of APs for use in positioning, and
4.  estimation of the distance between a new RSS observation and the fingerprints.

The factors affecting fingerprint generation are the placement and number of survey points and time samples. Most approaches have selected such parameters experimentally, except [25], where theoretical guidelines were provided by assuming specific statistical models for the distribution of RSS.

After the creation of fingerprints and before positioning is carried out, preprocessing steps can be performed to improve both accuracy and efficiency. For example, Yousief [20] proposes two methods for clustering of spatial locations based on covering APs to decrease computational complexity and improve accuracy and scalability of the positioning algorithm. The first method [26] involves an offline clustering of locations aiming to reduce the search space to a single cluster. The second proposes an incremental trilateration technique. Chen et al. [27] consider the similarity of signal values, as well as the covering APs, to generate a set of clusters using K-means to improve the power efficiency of mobile devices. Both of the above clustering techniques are carried out offline based on the training data. This hampers the operation of the system over time since WLAN infrastructures are highly dynamic and APs can be easily moved or discarded, in contrast to their base-station counterparts in cellular systems, which generally remain intact for long periods of time.

The third fingerprinting challenge is that of AP selection. Recall that estimation of a position in a 2D space requires measurements from at least three APs. In a typical WLAN environment, however, the number of available APs is much greater than three. Clearly, using all available APs increases the computational complexity of the positioning algorithm. Furthermore, the geometric configuration of APs in relation to each other can affect the accuracy of positioning [28] (this

is known as the Geometric Dilution of Precision (GDOP) in the context of GPS). Since RSS is dependent on the relative distance of the MC and each AP, as well as the topology of the environment in terms of obstacles causing NLOS propagation, subsets of available APs may report correlated readings, leading to needless redundancy and possibly biased estimates. This motivates the need for an AP selection component to choose a subset of the available APs for use in positioning. The most commonly used selection methodology is to choose a subset of APs with the highest observation RSS as the strongest APs provide the highest probability of coverage over time [26]. However, it is also known that the variance of measurements from an AP increases with its mean power at a given location. In cases where the measured RSS from an AP exhibits a high degree of variance, the survey values may be very different than the online measurement, degrading the accuracy of estimation [29]. Furthermore, it becomes more difficult to distinguish neighboring points in such cases. The recent work of Chen et al. [27] offers a novel selection strategy based on the discriminant power of each AP quantified through the entropy-based Information Gain criterion. The APs that best discriminate the survey points are the ones selected for positioning. In that work, AP selection is carried out offline and is independent of the distance measure used during positioning. Nevertheless, since distance measurement is performed using the selected AP set, the interplay between the selection strategy and distance measurement must be considered when designing the former.

Most RSS positioning efforts have been geared toward addressing the last challenge in fingerprinting: that of obtaining an estimate based on a new RSS observation. This essentially involves the calculation of a distance between the new RSS observation and the training record at each survey point. In the simplest case, the euclidean distance is used to find the distance between the observation and the center of the training RSS vectors at each survey point [16], [29]. The location estimate is either the survey point whose fingerprint has the smallest distance to the observation (nearest neighbor (NN) classification) or the average of $K$ closest survey points (K-nearest neighbor (KNN)). Despite its simplicity, the euclidean distance may fail to deliver adequate performance in cases where the distribution of RSS training vectors included in the fingerprints are nonconvex and multimodal [25], [30]. Such distributions arise frequently in indoor WLAN settings due to NLOS propagation and the presence of users [25]. The second group of WLAN positioning methods relies on probabilistic techniques to handle uncertainty in RSS measurements [28]. Such methods produce Maximum Likelihood (ML), Maximum *A Posteriori* (MAP), or Minimum Mean Square Error (MMSE) estimates of position using estimates of likelihood and posterior density functions. These densities are estimated through either parametric [31] or nonparametric methods [28], [32], [33]. The histogram is generally used as a nonparametric estimate of the probability density function; yet, its theoretical properties have remained unexplored. For example, the histogram density estimate is highly dependent on the choice of origin and bin width. Although theoretical guidelines are available for determination of
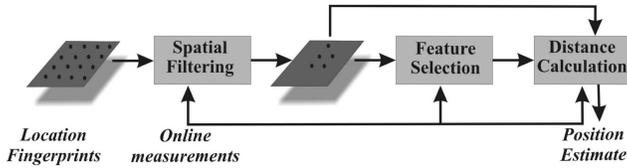
Fig. 1. Overview of the proposed system.

these parameters [34], existing methods have generally relied on experimental parameter tuning. Even with optimal parameters, the histogram's rate of convergence in the sense of Asymptotic Mean Integrated Square Error (AMISE) is relatively slow $(O(n^{-2/d+2}))$, where $n$ is the number of time samples and $d$ is the dimension of the measurement vector [34]. This indicates that a large number of time samples may be needed at each survey location for a histogram-based method to produce an acceptable performance. In [28], a single-observation kernel density estimator (KDE) has been utilized to model the RSS density and shown to have a similar performance to that of the histogram. The accuracy of the KDE is also heavily dependent on the number of training samples and the bandwidth parameters, which are experimentally determined in [28].

Kernel-based learning methods have also been explored in the context of WLAN positioning [35], [36], [37]. Canonical correlation analysis (CCA) is utilized in [35] and [36]; Support Vector Machines (SVMs) are used in [37] for both classification and regression and a comparable performance to KNN positioning was reported. Such methods, however, are not resilient to the loss of APs in a highly dynamic WLAN setting.

The contributions of this paper are threefold. First, a spatial filtering step is introduced to localize the estimation problem to a subset of the environment. Second, we investigate the problem of AP selection for positioning to enhance positioning accuracy, promote resilience to loss of APs, and motivate the examination of the interaction between this component and distance measurement. Third, we propose a kernelized distance for calculating the distance between an RSS observation and the fingerprints. This method nonlinearly maps the original fingerprint data to a high-dimensional feature space where the distribution of RSS training vectors is simplified and then carries out distance calculations in such a space. The outline of the proposed method is shown in Fig. 1.

For convenience, we provide a list of key mathematical symbols used in this paper in Table 1.

## 3  OVERVIEW OF THE PROPOSED METHOD AND PAPER OUTLINE

The fundamental problem in fingerprinting is to produce a position estimate using training information on a discrete grid of training points when a new RSS observation is received. That is, we seek a function $\hat{\mathbf{p}} = g(\mathbf{p}_1, \ldots, \mathbf{p}_N)$, where $\mathbf{p}_i = (x_i, y_i)$ represents the Cartesian coordinates of a point with respect to a predefined reference point $(0, 0)$. For the purposes of this work, the problem of positioning is restricted to the 2D space and, as such, height information shall be ignored. If the function $g$ is restricted to the class of

TABLE 1
List of Symbols Used in This Paper

| | |
|---|---|
| $\alpha$ | spatial filtering threshold |
| $a_i$ | The $i^{th}$ access point (AP) |
| $\mathcal{A}$ | Set of APs |
| $d$ | Number of APs used in positioning |
| $\mathbf{F}(\mathbf{p}_i)$ | Fingerprint record at point $i$ |
| $I_{\mathbf{p}_i}$ | Binary AP coverage vector at point $i$ |
| $L$ | Total number of APs in the environment |
| $L'$ | Number of strongest APs used in feature selection |
| $n$ | Number of RSS samples taken over time |
| $N$ | Total number of survey points |
| $N'$ | Number of survey points after spatial filtering |
| $\mathbf{p}$ | Point in 2D Cartesian space |
| $\mathcal{P}$ | Set of survey points |
| $\mathbf{r}_i(t)$ | Vector of RSS at point $i$ and time $t$ |

linear functions, the problem is reduced to determining a set of weights such that

$$\hat{\mathbf{p}} = \sum_{i=1}^{N} w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))\mathbf{p}_i, \qquad (1)$$

where $\mathbf{r}$ is the observation RSS vector and the fingerprint $\mathbf{F}(\mathbf{p}_i) = [\mathbf{r}_i(1); \ldots; \mathbf{r}_i(n)]$ is an $L \times n$ matrix defined as

$$\mathbf{F}(\mathbf{p}_i) \triangleq \begin{pmatrix} r_i^1(1) & \ldots & r_i^1(n) \\ \vdots & & \vdots \\ r_i^L(1) & \ldots & r_i^L(n) \end{pmatrix}. \qquad (2)$$

The columns of the fingerprint matrix are RSS vectors $\mathbf{r}_i(t) = [r_i^1(t), \ldots, r_i^L(t)]'$ that contain readings from $L$ APs at time $t$ when the MC resides at $\mathbf{p}_i$. Although the minimum number of APs for positioning in two dimensions is three, in practice, $L \gg 3$ due to the pervasiveness of WLANs. RSS measurements are integers in units of decibels below 1 mW with a dynamic range limited by the receiver sensitivity below and range limitations above. For the experimental apparatus described in Section 8, this range is $(-100, 0)$.

Although (1) produces the estimate using all survey points, those far away from the observation point must not be influential on the average. Moreover, statistics of RSS measurements, including the mean and variance, are spatially dependent [25]. For these reasons, we propose a spatially localized averaging strategy wherein AP coverage information is used to retain a subset of spatially relevant survey points (Fig. 1). Details of this spatial filtering step are discussed in Section 4.

As the second contribution of this paper, we address the problem of real-time AP selection for the WLAN problem. As previously mentioned, in a typical WLAN environment, the number of available APs is much greater than the minimum three required $(L \gg 3)$; in order to reduce computational cost and enhance accuracy, an AP selection method is needed. We propose a novel and real-time AP selection algorithm in Section 5 to determine a subset of at least three APs for use in positioning. This differs from the method of Chen et al. [27] in that the selection is performed during the online operation of the system to introduce resiliency to loss of APs. Moreover, we motivate the need for the joint design of AP selection and distance calculation components.

The last stage of positioning shown in Fig. 1 is distance calculations through the calculation of weights in the linear estimation rule of (1). These weight functions can be learned using various learning techniques such as regression and density estimation. In this paper, the weight functions are obtained as inner products in a kernel-defined feature space as discussed in Sections 6 and 7. Moreover, we show the connection between the proposed weights and Parzen-style nonparametric density estimation.

Finally, the proposed methods are evaluated using real data from an office environment in Section 8, and Section 9 concludes the paper and provides directions for future work.

## 4   SPATIAL FILTERING

A unique characteristic of the WLAN positioning problem is that feature patterns vary over space. For example, given a particular AP, the spread of RSS measurements at fixed locations decreases as the distance to the AP increases [25]. Thus, RSS patterns collected at survey points closer to the $i$th AP exhibit higher variance in the $i$th dimension of the feature vector when compared to more distant points. The spatial dependence of feature patterns motivates the use of spatially localized algorithms operating in areas where feature patterns are relatively similar [38]. Therefore, we propose a novel spatial filtering preprocessor to generate a subset of survey points $\{\mathbf{p}_1, \ldots, \mathbf{p}_{N'}\}$ deemed spatially relevant to the point of observation on receiving an observation vector $\mathbf{r}$. This is done based on the premise that points that are close in the physical space receive coverage from similar sets of APs.

The proposed spatial filtering uses binary AP coverage vectors generated from RSS vectors or fingerprints. Specifically, we define a coverage vector as $\mathbf{I_p} = [I_\mathbf{p}^1 \ldots I_\mathbf{p}^L]$, where $I_\mathbf{p}^i = 1$ if AP $i$ provides continuous coverage at $\mathbf{p}$. Continuous coverage in the context of this paper refers to the availability of an RSS reading greater than the receiver sensitivity for at least 90 percent of the time. The main reason for intermittent coverage is range limitations. For example, in indoor environments, a typical IEEE 802.11b AP provides a coverage range of less than 100 m at 5.5 megabits per second.[1] In office environments, this range is further reduced due to the presence of walls, furniture, and elevators causing NLOS propagation.

A natural distance measure for comparing the binary AP coverage vectors is the Hamming distance, indicating the number of positions in which they differ:

$$d_H(\mathbf{I_p}, \mathbf{I_{p'}}) = \sum_{i=1}^{L} |I_\mathbf{p}(i) - I_\mathbf{p'}(i)|.$$

Given the Hamming distance, a survey point $\mathbf{p}_i$ passes the spatial filtering stage if its AP coverage vector matches that of the observation, $\mathbf{I_r}$. This is reflected in the final position estimate by assigning a weight of zero to those points that fail spatial filtering:

$$w_s(\mathbf{r}, \mathbf{F}(\mathbf{p}_i), \alpha) = \begin{cases} 0 & \text{if } d_H(\mathbf{I_r}, \mathbf{I_{p_i}}) > \alpha L, \\ 1 & \text{otherwise.} \end{cases} \quad (3)$$

1. http://www.netgear.com, http://www.cisco.com.

The parameter $0 \leq \alpha \leq 1$ regulates the localization of spatial filtering and $L$ is the total number of APs. Large values of $\alpha$ allow most survey points to pass through the filtering stage and lead to less localization in space, whereas small values of this parameter restrict the filtered set to a few points, producing highly localized regions. It is important to note that, due to time variations in the propagation environment, an observation AP vector may not exactly match a fingerprint even if both measurements were collected at the same position [20]. Such variations limit the effectiveness of spatial filtering in practice as small values of $\alpha$ may exclude relevant survey points.

Finally, the set of filtered points is defined as $\mathcal{P}' = \{\mathbf{p}_i : w_s(\mathbf{r}, \mathbf{F}(\mathbf{p}_i), \alpha) = 1\}$ and $|\mathcal{P}'| = N'$. With reference to Fig. 1, AP selection and distance calculation steps are then carried out on this set.

## 5   AP SELECTION

Due to the wide deployment of APs, the dimension of the measurement vector $(L)$ is generally much greater than the minimum three needed for positioning. Recall that, at a spatial point $\mathbf{p}$, the RSS from an AP $a$ is a function of the distance between $\mathbf{p}$ and $a$, as well as environmental factors that lead to shadowing and multipath effects. Since a WLAN positioning system may receive RSS measurements from APs belonging to different networks that are not deployed with the goal of positioning in mind, distinct APs may produce similar measurements, leading to biased estimates and redundant computations, as discussed in Section 3. This motivates the use of AP selection techniques to select a subset of available APs for positioning. Surprisingly, this problem has received little attention in the existing literature [27]. Selection methodologies have been limited to choosing a subset of APs with the highest observation RSS arguing that the strongest APs provide the highest probability of coverage over time [26]. The strongest APs, however, may not always provide the best positioning accuracy as shown in [27].

Although the AP selection problem is one of dimensionality reduction, it is important to distinguish between the problem of feature selection and that of AP selection. Feature selection is a well-studied problem and aims to project the data into a lower dimensional space. Examples of such methods include Principle Component Analysis (PCA) and Linear Discriminant Analysis (LDA) [30]. Due to real-time constraints of the WLAN problem and to ensure comparability to existing methods, we limit the scope of this paper to AP selection, noting that this need not be the case in future developments. Recall that the set of APs covering the survey points is denoted as $\mathcal{A}$ and $|\mathcal{A}| = L$. The objective of AP selection is to determine a set $\mathcal{A}' \subseteq \mathcal{A}$ such that $|\mathcal{A}'| = d < L$.

The goal of this paper with respect to AP selection is to demonstrate the need for AP selection, highlight the interplay among this component and the distance calculation step, and encourage future research in the area. With this in mind, we propose selection methods that reflect the basic principles of minimizing correlation between selected

TABLE 2
Computational and Storage Complexity for Four AP Selection Methods: DM, Bhattacharyya Distance ($\sigma_1 = \sigma_2 = 1$);
B, Bhattacharyya Distance ($\sigma_1 \neq \sigma_2$); IP, Information Potential; and S, Strongest Signal Power

| | Computational Complexity | | | Storage Requirements | |
| | additions | sort operations ([# of sorts]$sort$[# of elements]) | execution time (ms) | integers | floats |
|---|---|---|---|---|---|
| DM | $\binom{L'}{d}\binom{d}{2}$ | $sort[\binom{L'}{d}] + [\binom{L'}{d}\binom{d}{2}]sort[N']$ | 0.56 | 0 | $L^2 N$ |
| B | $\binom{L'}{d}\binom{d}{2}$ | $sort[\binom{L'}{d}] + [\binom{L'}{d}\binom{d}{2}]sort[N']$ | 0.56 | 0 | $L^2 N$ |
| IP | $\binom{L'}{d}\binom{d}{2}$ | $sort[\binom{L'}{d}] + [\binom{L'}{d}\binom{d}{2}]sort[N']$ | 0.56 | 0 | $L^2 N$ |
| S | 0 | $sort[dL']$ | 0.0314 | 0 | 0 |

Parameters $L$, $N$, $L'$, $d$, and $N'$ are as defined in Table 1.

APs, paralleling the GDOP problem. Moreover, the selection strategies are designed to mirror the properties of the distance calculation schemes under investigation. The LDA-like alternative is to devise AP selection methods that consider the discrimination ability across survey points and focus on choosing a data representation that best separates the survey points.

In considering the correlation between signals from the selected APs, the objective is to obtain a representation with minimal redundancy. This is achieved through the use of divergence measures to quantify separability of two APs. Let $R_{\mathbf{p}}^i = [r_{\mathbf{p}}^i(1) \ldots r_{\mathbf{p}}^i(n)]$ be the vector of RSS readings from the $i$th AP at location $\mathbf{p}$. Then,

$$
\begin{aligned}
\mathcal{A}' &= \arg \min_{\mathcal{A}_k \subset \mathcal{A}: |\mathcal{A}_k| = d} D(\mathcal{A}_k) \\
&= \arg \min_{\mathcal{A}_k \subset \mathcal{A}: |\mathcal{A}_k| = d} \sum_{a_i, a_j \in \mathcal{A}_k} D(a_i, a_j),
\end{aligned}
\tag{4}
$$

where $D(\mathcal{A}_k)$ is the total divergence cost of the AP set $\mathcal{A}_k$ calculated as the sum of pairwise divergences, $D(a_i, a_j)$. We define $D(a_i, a_j)$ as the min-max divergence between two APs, corresponding to the minimum divergence over all points: $D(a_i, a_j) = \min_{\mathbf{p}_k} d(R_{\mathbf{p}_k}^i, R_{\mathbf{p}_k}^j)$, where $d(R_{\mathbf{p}_k}^i, R_{\mathbf{p}_k}^j)$ is a measure of distance between the two time series. Although a plethora of divergence measures have been utilized in existing literature [30], we suggest the following to illustrate the ideas presented here:

- **Bhattacharyya distance.** Defined as

$$
d_B = -\ln \left( \int \sqrt{f(x)g(x)dx} \right),
$$

the Bhattacharyya distance measures the distance between two probability density functions $f(x)$ and $g(x)$. The integral is not easily evaluated for general density functions, but a closed-form solution does exist in the case of two Gaussian distributions with known mean and covariance $\mathcal{N}(\mu_i, \Sigma_i)$, $i = 1, 2$. In this case, the divergence measure reduces to [30]

$$
\begin{aligned}
d_B &= \frac{1}{8}(\mu_1 - \mu_2)^T \left[ \frac{\Sigma_1 + \Sigma_2}{2} \right]^{-1} (\mu_1 - \mu_2) \\
&+ \frac{1}{2}\ln \frac{|\frac{\Sigma_1 + \Sigma_2}{2}|}{\sqrt{|\Sigma_1||\Sigma_2|}}.
\end{aligned}
$$

Approximating the RSS distributions at APs $i$ and $j$ at $\mathbf{p}_k$ by Gaussians using the sample statistics

$\mu_{\mathbf{p}_k}^i = \frac{1}{n}\sum_{t=1}^n r_{\mathbf{p}_k}^i(t)$ and $\sigma_{\mathbf{p}_k}^i = \frac{1}{n-1}\sum_{t=1}^n (r_{\mathbf{p}_k}^i(t) - \mu_{\mathbf{p}_k}^i)^2$, the divergence measure can be written as follows:

$$
\begin{aligned}
d_B(R_{\mathbf{p}_k}^i, R_{\mathbf{p}_k}^j) &= \frac{1}{8}(\mu_{\mathbf{p}_k}^i - \mu_{\mathbf{p}_k}^j)^2 \left[ \sigma_{\mathbf{p}_k}^{i^2} + \sigma_{\mathbf{p}_k}^{j^2} \right]^{-1} \\
&+ \frac{1}{2}\log \frac{\frac{1}{2}(\sigma_{\mathbf{p}_k}^{i^2} + \sigma_{\mathbf{p}_k}^{j^2})}{\sigma_{\mathbf{p}_k}^i \sigma_{\mathbf{p}_k}^j}.
\end{aligned}
\tag{5}
$$

For the WLAN problem, $R_{\mathbf{p}_k}^i$ are generally not Gaussian [25]. Despite this, our experiments show that $d_B$ provides an acceptable distance measure for AP selection when the sample statistics from $R_{\mathbf{p}_k}^i$ and $R_{\mathbf{p}_k}^j$ are used.

- **Information potential.** Used in [39] as a clustering evaluation function, it is developed as a nonparametric estimate of Renyi's quadratic information to measure the distance between two probability density functions. This measure is defined as

$$
d_I(R_{\mathbf{p}_k}^i, R_{\mathbf{p}_k}^j) = -\ln \left( \frac{1}{n^2}\sum_{t=1}^n \sum_{t'=1}^n k(r_{\mathbf{p}_k}^i(t), r_{\mathbf{p}_k}^j(t')) \right).
\tag{6}
$$

In (6),

$$
k(x_i, x_j) = \frac{1}{\sqrt{2\pi(\sigma_1^2 + \sigma_2^2)}} \exp \left( \frac{-\|x_i - x_j\|^2}{2(\sigma_1^2 + \sigma_2^2)} \right).
$$

The parameters $\sigma_1$ and $\sigma_2$ control the width of the Gaussian kernel—automatic methods for selection of these parameters will be discussed in Section 6. To produce a minimum distance of zero, $d_I(R_{\mathbf{p}_k}^i, R_{\mathbf{p}_k}^j)$ must be normalized as in [39].

Since AP selection is performed on the subset of survey points generated by the spatial filtering stage, it must be carried out in real time. However, the proposed selection methods involve an exhaustive search over $\binom{L}{d}\binom{d}{2}$ sets. In order to reduce complexity and ensure that the selected APs provide coverage to survey points near the observation, AP selection is performed on the strongest $L' < L$ APs. We summarize the computation and storage requirements of each of the proposed methods in Table 2. Calculations assume offline computation of the pairwise divergences $d(R_{\mathbf{p}_k}^i, R_{\mathbf{p}_k}^j)$ in all cases, and execution times are reported for a Matlab implementation on a Pentium 4 processor with 2 Gbytes of RAM. Execution times are averaged over 500 runs with $L' = 5$ and $d = 3$ to mirror our experimental setup in Section 8.

Notice that the Bhattacharyya method relies only on the first two moments of the RSS distribution at each survey point. The information potential method uses nonparametric probability density estimates and is consequently able to capture higher order structure in the data. The experiments in Section 8 show that this additional information may not necessarily lead to significant improvements.

## 6 DISTANCE CALCULATION

Having thus related the details of spatial filtering and AP selection, we turn to the problem of distance calculation between an observation and fingerprints and discuss the calculation of weights appearing in (1). Specifically, we require that these are decreasing functions of the distance between an observation vector and the training record. That is, survey points whose training records closely match the observation should receive a higher weight. In particular, the functions $w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))$ should satisfy the following properties:

1. $\sum_{i=1}^{N} w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) = 1$ so that the estimated position belongs to the convex hull defined by the set of survey positions. This can be achieved by including a normalization term in (1) and will not be discussed further.

2. $w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))$ is a monotonically decreasing function in both arguments with respect to a distance measure $d(\mathbf{r}, \mathbf{F}(\mathbf{p}_j))$:

$$d(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) \geq d(\mathbf{r}', \mathbf{F}(\mathbf{p}_i)) \Rightarrow$$
$$w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) \leq w(\mathbf{r}', \mathbf{F}(\mathbf{p}_i)), \tag{7}$$

$$d(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) \geq d(\mathbf{r}, \mathbf{F}(\mathbf{p}_j)) \Rightarrow$$
$$w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) \leq w(\mathbf{r}, \mathbf{F}(\mathbf{p}_j)). \tag{8}$$

Although any weight function satisfying the above conditions can be chosen, we propose using the average normalized inner product as the weight function for reasons that will become clear shortly:

$$w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) = \frac{1}{n} \sum_{t=1}^{n} \frac{\langle \mathbf{r}, \mathbf{r}_i(t) \rangle}{\|\mathbf{r}\| \, \|\mathbf{r}_i(t)\|}, \tag{9}$$

where $\langle \mathbf{r}, \mathbf{r}_i(t) \rangle = \mathbf{r}\mathbf{r}_i(t)^T$ denotes the canonical inner product in $\mathbb{R}^d$ and $d$ is the number of APs used for positioning. These weights satisfy (7) and (8) since they are inversely proportional to the square of the average normalized euclidean distance between the observation and training vectors as shown in Theorem 1:

$$d_{AE}(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))^2 = \frac{1}{n} \sum_{t=1}^{n} \left\| \frac{\mathbf{r}}{\|\mathbf{r}\|} - \frac{\mathbf{r}_i(t)}{\|\mathbf{r}_i(t)\|} \right\|^2 \tag{10}$$

$$= \left\| \frac{\mathbf{r}}{\|\mathbf{r}\|} - \frac{1}{n} \sum_{t=1}^{n} \frac{\mathbf{r}_i(t)}{\|\mathbf{r}_i(t)\|} \right\|^2 + C, \tag{11}$$

where $C = 1 - \|\frac{1}{n} \sum_{t=1}^{n} \frac{\mathbf{r}_i(t)}{\|\mathbf{r}_i(t)\|}\|^2$ relates the average distance to the distance from the center of mass of the training
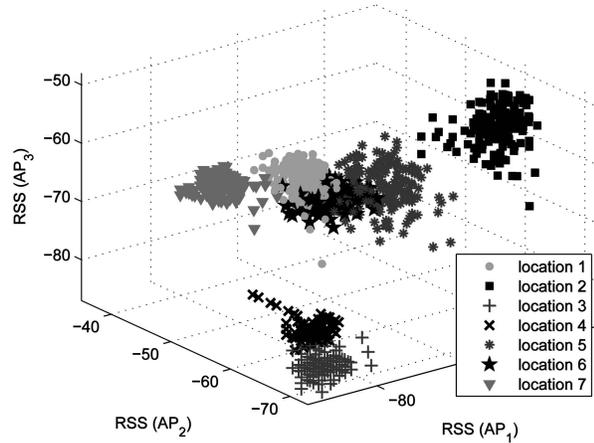


Fig. 2. Example of RSS measurement space for various locations (three APs).

vectors. Note that the normalization is necessary to ensure compliance with (7) and (8) and that the distance is well behaved as $\|\mathbf{r}\| > 0$ in a WLAN context. We now state a theorem on the properties of the weight function. The proof of this theorem can be found in Appendix A.

**Theorem 1 (Properties of weight function).** *The average normalized inner product $w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))$ has four properties:*

1. $0 \leq w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) \leq 1$.
2. $w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) = \langle \frac{\mathbf{r}}{\|\mathbf{r}\|}, \frac{1}{n} \sum_{t=1}^{n} \frac{\mathbf{r}_i(t)}{\|\mathbf{r}_i(t)\|} \rangle$. *That is, the weight function is the inner product between the observation and mean of the training vectors in a normalized space.*
3. $w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) = -\alpha(d_{AE}(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))^2 - C)$, *where $\alpha = 1/2$ and $C = 2$.*
4. $w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))$ *is a monotonically decreasing function in both arguments with respect to $d_{AE}(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))^2$ defined in (10):*

$$d_{AE}^2(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) \geq d_{AE}^2(\mathbf{r}', \mathbf{F}(\mathbf{p}_i)) \Rightarrow$$
$$w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) \leq w(\mathbf{r}', \mathbf{F}(\mathbf{p}_i)), \tag{12}$$

$$d_{AE}^2(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) \geq d_{AE}^2(\mathbf{r}, \mathbf{F}(\mathbf{p}_j)) \Rightarrow$$
$$w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) \leq w(\mathbf{r}, \mathbf{F}(\mathbf{p}_j)). \tag{13}$$

As seen in (9), the weight function is the average of the cosines between the observation and the training vectors. The minimum therefore occurs when the observation is orthogonal to all training vectors. Experimental results, reported in Section 8, indicate that this angular measure is not an effective similarity measure between two RSS vectors as the maximum angle between them is rather small in $\mathbb{R}^d$. Furthermore, the presence of users and NLOS propagation results in complex distributions of RSS patterns for a given survey point, as shown in Fig. 2. Motivated by the complexity of RSS patterns in this euclidean space, we question whether any improvements can be achieved by using more sophisticated nonlinear weight functions. To this end, we use the nonlinear

mapping $\phi : \mathbf{r} \in \mathbb{R}^d \mapsto \phi(\mathbf{r}) \in \mathcal{F}$ to map the input data to a higher (possibly infinite) dimensional space $\mathcal{F}$ where the weight calculations take place.

At first glance, the calculation of weights in a possibly infinite dimensional space may seem computationally intractable. Fortunately, the *kernel trick* [40] can be used to calculate the inner product in $\mathcal{F}$ without the need for explicit evaluation of the mapping $\phi$. The kernel trick allows the replacement of inner products in $\mathcal{F}$ by a kernel evaluation on the input vectors. In the WLAN context, the kernel is a function $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ such that $k(\mathbf{r}, \mathbf{r}') = \langle \phi(\mathbf{r}), \phi(\mathbf{r}') \rangle$ [41]. The reason for the particular choice of the weight function now becomes clear: Since the training data only enter weight calculations through inner products, the kernel trick can be used to carry out inner products in $\mathcal{F}$ without the need for explicit evaluation of mapping $\phi$. The kernelized weight function then becomes

$$w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) = \frac{1}{n} \sum_{t=1}^{n} \frac{\langle \phi(\mathbf{r}), \phi(\mathbf{r}_i(t)) \rangle}{\|\phi(\mathbf{r})\| \, \|\phi(\mathbf{r}_i(t))\|} \qquad (14)$$

$$= \frac{1}{n} \sum_{t=1}^{n} \frac{k(\mathbf{r}, \mathbf{r}_i(t))}{\sqrt{k(\mathbf{r}, \mathbf{r}) k(\mathbf{r}_i(t), \mathbf{r}_i(t))}}. \qquad (15)$$

Mercer's Theorem [40] guarantees the correspondence between a kernel function and an inner product in a feature space $\mathcal{F}$, given that the kernel is a positive definite function. Examples of such functions are the Polynomial kernel

$$k(x, y) = (\langle x, y \rangle + c)^d,$$

Exponential kernel $k(x, y) = \exp(k_1(x, y))$, and Gaussian kernel $k(x, y) = \exp\left(-\frac{\|x - y\|^2}{2\sigma^2}\right)$, where $c$, $d$, and $\sigma$ are parameters and $k_1(x, y)$ is a valid kernel function.

# 7 CHOICE OF KERNEL AND PARAMETER SELECTION

In this section, the choice of the kernel function and its parameters is discussed. In order to motivate the choices, the nature of the feature space induced by a kernel function is first briefly reviewed.

## 7.1 Kernel-Induced Feature Space

Two common instantiations of the feature space associated with a positive definite kernel are the Mercer Kernel Map and the Reproducing Kernel Hilbert Space (RKHS) [40]. For brevity, we use the RKHS for examining the choice of the kernel and note that the two Hilbert spaces are isometrically isomorphic.

Construction of the RKHS begins by considering $\phi : \mathbf{r} \mapsto k(\cdot, \mathbf{r})$, mapping an input vector to a function that measures similarity between the input and all elements in the input space (in this case, $\mathbb{R}^d$). Now, consider the space of functions $f : \mathbb{R}^d \to \mathbb{R}$ of the form $f(\cdot) = \sum_{i=1}^{n} \alpha_i k(\cdot, \mathbf{r}_i)$ for $n > 0$, $\alpha_i \in \mathbb{R}$, and $\mathbf{r}_i \in \mathbb{R}^d$. The space is endowed with an inner product defined for two elements $f(\cdot) = \sum_{i=1}^{n} \alpha_i k(\cdot, \mathbf{r}_i)$ and $g(\cdot) = \sum_{j=1}^{m} \beta_j k(\cdot, \mathbf{r}_j)$ as

$$\langle f, g \rangle = \sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_i \beta_j k(\mathbf{r}_i, \mathbf{r}_j). \qquad (16)$$

The reproducing property then follows: $f(\mathbf{r}) = \langle f, k(\cdot, \mathbf{r}) \rangle$; consequently, $\langle k(\mathbf{r}, \cdot), k(\mathbf{r}', \cdot) \rangle = k(\mathbf{r}, \mathbf{r}')$. Completing the space in the norm $\|f\| = \sqrt{\langle f, f \rangle}$ results in a Hilbert space associated with a kernel $k$ in which $k(\mathbf{r}, \mathbf{r}') = \langle \phi(\mathbf{r}), \phi(\mathbf{r}') \rangle$. Using this notation, the weight function of (14) can be written as an inner product in the RKHS as follows:

$$w_{RKHS}(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) = \frac{1}{n} \sum_{t=1}^{n} \frac{k(\mathbf{r}, \mathbf{r}_i(t))}{\sqrt{k(\mathbf{r}, \mathbf{r}) k(\mathbf{r}_i(t), \mathbf{r}_i(t))}}$$

$$= \left\langle \frac{1}{n} \sum_{t=1}^{n} \frac{k(\cdot, \mathbf{r}_i(t))}{\sqrt{\langle k(\cdot, \mathbf{r}_i(t)), k(\cdot, \mathbf{r}_i(t)) \rangle}}, \frac{k(\cdot, \mathbf{r})}{\sqrt{\langle k(\cdot, \mathbf{r}), k(\cdot, \mathbf{r}) \rangle}} \right\rangle. \qquad (17)$$

**Theorem 2.** *The weight function* $w_{RKHS}(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))$ *is a decreasing function of the squared distance to the center of the training vectors in* $\mathcal{F}$:

$$w_{RKHS}(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) = -\frac{1}{2}\left(d_k(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))^2 - C\right), \qquad (18)$$

*where* $d_k(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))^2 = \left\| \frac{\phi(\mathbf{r})}{\|\phi(\mathbf{r})\|} - \frac{1}{n}\sum_{t=1}^{n} \frac{\phi(\mathbf{r}_i(t))}{\|\phi(\mathbf{r}_i(t))\|} \right\|^2$ *and*

$$C = 1 + \frac{1}{n^2} \sum_{t=1}^{n} \sum_{t'=1}^{n} \frac{k(\mathbf{r}_i(t), \mathbf{r}_i(t'))}{\sqrt{k(\mathbf{r}_i(t), \mathbf{r}_i(t)) k(\mathbf{r}_i(t'), \mathbf{r}_i(t'))}}$$

*is constant for a given survey point.*

The proof of the theorem is provided in Appendix B.

## 7.2 Choice of Kernel

The choice of the kernel function $k(\cdot, \mathbf{r})$ defines the new feature space $\mathcal{F}$ and is therefore an important design decision. Genton [42] categorizes kernel functions into three main classes: stationary, locally stationary, and nonstationary. Stationary kernels are those that depend only on the lag vector between their arguments. An isotropic stationary kernel results when only the magnitude of the lag vector is considered. The commonly used Gaussian kernel is an example of an isotropic stationary kernel. Locally stationary kernels are a product of two kernels,

$$K(x, z) = K_1\left(\frac{x + z}{2}\right) K_2(x - z),$$

where $K_1$ reflects the global structure in the data and $K_2$ represents the local structure. Finally, nonstationary kernels are those that depend directly on their arguments and include the polynomial kernel.

Since the weight function of (14) is the average of inner products in a space where all patterns have unit norm and RSS readings have been shown to be stationary over small timescales [25], we choose to use a stationary kernel so that $\|\phi(\mathbf{r})\|^2 = k(\phi(\mathbf{r}), \phi(\mathbf{r})) = k(0), \forall \mathbf{r} \in \mathbb{R}^d$. Although the measurements are correlated over time, this choice provides acceptable positioning accuracy in the experiments in Section 8. Within the class of stationary kernels, the Gaussian kernel has been widely studied and applied to a broad range of pattern recognition problems. Using a Gaussian kernel, all mapped points lie within the same orthant in the feature space, ensuring that $0 \leq w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) \leq 1$. Moreover, for a set of distinct but correlated training points $\mathbf{r}_i(1), \ldots, \mathbf{r}_i(t)$, the mapped points $\phi(\mathbf{r}_i(1)), \ldots, \phi(\mathbf{r}_i(t))$ are linearly independent and span a $t$-dimensional subspace of the infinite

dimensional RKHS defined on the domain $\mathbb{R}^d$ [40]. Using the Gaussian kernel, the weights become

$$w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) = \frac{1}{n} \sum_{t=1}^{n} \exp\left(\frac{-\|\mathbf{r} - \mathbf{r}_i(t)\|^2}{2\sigma^2}\right). \qquad (19)$$

### 7.3 Choice of Kernel Parameters

The parameter $\sigma$ controls the width of the kernel and has a great effect on the shape of the feature space. To see this, consider two extreme cases:

- As $\sigma \to 0$, $\langle \phi(\mathbf{r}), \phi(\mathbf{r}') \rangle = k(\mathbf{r}, \mathbf{r}') \to \delta(\|\mathbf{r} - \mathbf{r}'\|)$, where $\delta(\cdot)$ is the Kronecker delta function. In other words, the input vectors become orthogonal in the RKHS. Moreover, $w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) \to \frac{1}{n} \sum_{t=1}^{n} \delta(\|\mathbf{r} - \mathbf{r}_i(t)\|)$, risking significant overfitting of the training data.
- As $\sigma \to \infty$, $\langle \phi(\mathbf{r}), \phi(\mathbf{r}') \rangle = k(\mathbf{r}, \mathbf{r}') \to 1$ and the input patterns generate collinear functions in the RKHS. The weight function $w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) \to 1$, underfitting the data.

Determination of the kernel parameter $\sigma$ is not an easy task and has been extensively studied in the literature using techniques such as cross validation [43]. Such methods aim to optimize the kernel parameter by minimizing an empirical leave-out-one error measure over the training set. Due to real-time computation constraints, limited number of training patterns, and large variances between validation and test experiments, cross validation is not used in this paper.

An alternative approach to determination of the kernel width parameter capitalizes on the knowledge available in Parzen-window density estimation. Specifically, given the samples $\mathbf{r}_i(t)$, $t = 1, \ldots, n$, from a sequence of independent and identically distributed (i.i.d.) random variables, the Kernel Density Estimate (KDE) of the unknown density $f(\mathbf{r}|\mathbf{p}_i)$ is

$$\hat{f}(\mathbf{r}|\mathbf{p}_i) = \frac{\sigma^{-d}}{n} \sum_{t=1}^{n} k\left(\frac{\mathbf{r} - \mathbf{r}_i(t)}{\sigma}\right) = \sigma^{-d} w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)), \qquad (20)$$

where $k(\cdot)$ is a zero-mean nonnegative kernel function with unit area (for example, Gaussian kernel normalized to unit area). In general, the parameter $\sigma$ is determined to minimize the AMISE between the estimated and true densities. To ensure the consistency of the resulting estimate (vanishing bias and variance), $\sigma$ must be chosen such that $\sigma \to 0$ and $n\sigma \to \infty$ as $n \to \infty$ [34].

It can be shown that, for the multivariate KDE, the optimal bandwidth is on the order of $O(n^{\frac{-1}{4+d}})$, corresponding to a minimum AMISE that is on the order of $O(n^{\frac{-4}{4+d}})$ [34]. In particular, Silverman [44] recommends the following formula as a quick estimate of the parameter for a Gaussian kernel but cautions against overfitting:

$$\sigma^* = \left(\frac{4}{2d+1}\right)^{1/(d+4)} \hat{\sigma} n^{-1/(4+d)}. \qquad (21)$$

Here, $\hat{\sigma}^2 = \frac{1}{d} \sum_{l=1}^{d} \sigma_{r_i^l}^2$ is the average of marginal variances.

Before proceeding further, we list some of the conditions that make this estimate suboptimal:

- The i.i.d. assumption used in the derivation of the above estimate is violated for RSS samples as they are correlated over time periods ranging from a few seconds [25] to several minutes [32]. Depending on the degree of correlation, either the sampling period can be decreased or other correlation handling methods can be used. For example, Youssef and Agrawala [32] propose a parametric solution for dealing with the effect by assuming that variations of RSS over time can be modeled as an autoregressive process with white Gaussian noise. However, the extension of this method to the nonparametric case is unclear. Experiments in Section 8 show that, despite this correlation, the proposed method performs well.
- Although the above estimate is asymptotically optimal, its effectiveness clearly suffers when only a finite number of training samples is available.
- The above estimate uses a single kernel width across all dimensions and ignores any cross-correlations.
- Accurate density estimates may not lead to accurate estimates of weight functions (regression).

Despite the above suboptimalities, the above estimate has been successfully used in [45] in the context of kernel-based classification. In this view, we use the scaled Gaussian weights

$$w_{SG}(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) = (2\pi)^{-d/2} \sigma^{-d} \frac{1}{n} \sum_{t=1}^{n} \exp\left(\frac{-\|\mathbf{r} - \mathbf{r}_i(t)\|^2}{2\sigma^2}\right). \qquad (22)$$

Theorem 3 reveals important properties of this weight function (the proof is given in Appendix C).

**Theorem 3 (Properties of kernelized weight function).** *The kernelized weights defined in (22) have the following properties:*

1. $0 \leq w_{SG}(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) \leq (2\pi)^{-d/2} \sigma^{-d}$.
2. $w_{SG}(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) = (2\pi)^{-d/2} \sigma^{-d} \langle \phi(\mathbf{r}), \frac{1}{n} \sum_{t=1}^{n} \phi(\mathbf{r}_i(t)) \rangle$. *That is, the weight function is the scaled inner product between the observation and mean of the training vectors in the kernel space.*
3. $w_{SG}(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) = -\alpha(d_k(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))^2 - C)$, *where* $d_k = \|\phi(\mathbf{r}) - \frac{1}{n} \sum_{t=1}^{n} \phi(\mathbf{r}_i(t))\|$ *is the distance from the observation to the center of the training vectors in $\mathcal{F}$,* $\alpha = (\sqrt{2\pi}\sigma)^d / 2$, *and*

$$C = 1 + \frac{1}{n^2} \sum_{t=1}^{n} \sum_{t'=1}^{n} \exp\left(\frac{-\|\mathbf{r}_i(t) - \mathbf{r}_i(t')\|^2}{2\sigma^2}\right).$$

4. $w_{SG}(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))$ *is a monotonically decreasing function in both arguments with respect to* $d_k(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))^2$:

$$d_k^2(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) \geq d_k^2(\mathbf{r}', \mathbf{F}(\mathbf{p}_i)) \Rightarrow$$
$$w_{SG}(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) \leq w_{SG}(\mathbf{r}', \mathbf{F}(\mathbf{p}_i)), \qquad (23)$$

$$d_k^2(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) \geq d_k^2(\mathbf{r}, \mathbf{F}(\mathbf{p}_j)) \Rightarrow$$
$$w_{SG}(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) \leq w_{SG}(\mathbf{r}, \mathbf{F}(\mathbf{p}_j)). \qquad (24)$$

**Inputs:**
  $0 \leq \alpha \leq 1$: spatial filtering threshold;
  $d$: dimension of measurement vector;
  $\mathcal{P}$: set of survey points;
  $\mathbf{F}((\mathbf{p}_i))$, $i = 1, \ldots, N$: RSS fingerprints.
**Output:**
  $\hat{\mathbf{p}}$: position estimate.
**Positioning algorithm:**
  Step 1. Spatial Filtering (using (3)):
      $\mathcal{P}' = \{\mathbf{p}_i : w_s(\mathbf{r}, \mathbf{F}(\mathbf{p}_i), \alpha) = 1\}$, $|\mathcal{P}'| = N'$.
  Step 2. AP selection (using (4)):
      $\mathcal{A}' = \arg\min_{\mathcal{A}_k : |\mathcal{A}_k| = d} D(\mathcal{A}_k)$, $|\mathcal{A}'| = d$.
  Step 3. Distance calculation (using (21), $d$ APs, $\mathbf{p}_i \in \mathcal{P}'$):
      $w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) = \frac{1}{n(\sqrt{2\pi}\sigma)^d} \sum_{t=1}^{n} \exp\left(\frac{-||\mathbf{r} - \mathbf{r}_i(t)||^2}{2\sigma^2}\right)$,
  Step 4. Estimation (using (1)):
      $\hat{\mathbf{p}} = \frac{\sum_{\mathbf{p}_i \in \mathcal{P}'} w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))\mathbf{p}_i}{\sum_{\mathbf{p}_i \in \mathcal{P}'} w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))}$

Fig. 3. Outline of the positioning algorithm.

It is important to comment on the first part of this theorem as we required the weights to lie in [0, 1] previously. Note that the distance from an observation to each survey is measured in a distinct space. In particular, because the kernel parameter $\sigma$ is determined based on the statistics of the samples at a given point, the corresponding RKHS are in fact different. Normalization by the variance ensures comparability among the different survey points by ensuring that similarity functions are defined on regions of unit volume, that is, $\int k(\mathbf{r}, \mathbf{r}_i)d\mathbf{r} = 1$.

The outline of the proposed positioning algorithm using the above weights is shown in Fig. 3.

### 7.4 Interpretation as the Nadaraya-Watson Regression Estimator

Using the kernel-based weights and assuming a stationary kernel, the estimate of (1) becomes

$$\hat{\mathbf{p}} = \frac{\sum_{i=1}^{N'} \mathbf{p}_i \sum_{t=1}^{n} \sigma_i^{-d} k\left(\frac{\mathbf{r} - \mathbf{r}_i(t)}{\sigma_i}\right)}{\sum_{i=1}^{N'} \sum_{t=1}^{n} \sigma_i^{-d} k\left(\frac{\mathbf{r} - \mathbf{r}_i(t)}{\sigma_i}\right)}. \tag{25}$$

This formulation reveals an interesting connection. For the case of $n = 1$, the proposed estimate reduces to the nonparametric Nadaraya-Watson regression estimator of the posterior mean $E(\mathbf{p}|\mathbf{r})$ [34] used in positioning in cellular systems [2]. For $n > 1$, we obtain the regression estimator with multiple RSS samples per survey point. In contrast to the one-dimensional one-observation KDE used in [28] as a density estimate, (25) uses multiple training samples to form a multidimensional kernel regression. Both of these are important properties in view of the need for multiple time samples to characterize time variations in RSS and to model possible correlation among APs on the same channel.

The formulation of (25) also corresponds to the MMSE estimator of position in the case where the empirical probability density estimate (epdf) is used as the prior $f(\mathbf{p})$ [38].

### 7.5 Complexity

The computational and storage complexity of the proposed method is compared to those of two state-of-the-art methods in Table 3. Calculations assume offline computation of observation-independent parameters including histograms, sample means and variances, and kernel bandwidths. Execution times are reported for a Matlab implementation on a Pentium 4 processor with 2 Gbytes of RAM. Note that the number of points after filtering ($N'$) can be controlled through the spatial filtering threshold $\alpha$ so that $N' \ll N$. Also, as reported in Section 8, the number of time samples used for the kernel method, $n$, can be as small as 10. Execution times are averaged over 500 runs and reported for $\alpha = 1$, $n = 10$, $K = 4$ for the KNN, and $K = 66$ for the histogram to reflect the experimental setup of Section 8.

## 8 EXPERIMENTS AND RESULTS

This section provides details on the experimental evaluation of the proposed methods using real data from an office environment that reflects a typical setting for offering LBSs.

### 8.1 Setup

#### 8.1.1 The Environment

Experimental results are reported for an office environment that mimics characteristics of a typical WLAN positioning application in terms of area, number of APs, and propagation environment.[2] Specifically, the experiments were carried out on the fourth floor of an eight-story building at the University of Toronto. The dimensions of the experimentation site were $36 \times 42$ m, which are comparable to those reported in [23], [25], and [43]. A total of 33 APs were detectable throughout the floor, each providing only a partial coverage of the environment. In particular, the average number of APs per point was 9.6, and out of these, only three are used for positioning as described below. It has been previously shown that increasing the number of APs leads to increased positioning accuracy [28]. We therefore use three APs to demonstrate the worst case performance of the methods. The measurements were collected using a Toshiba Satellite laptop with a Pentium M processor, an Intel PRO/Wireless 2915ABG Network Adapter, and the Windows XP operating system. RSS measurements were obtained by a publicly available network sniffer software, NetStumbler,[3] providing a maximum sampling rate of 2 samples/sec.

#### 8.1.2 Training

The location of the survey points can be determined in two ways [34]. The first scenario is a random design where the pairs $(\mathbf{p}_i, \mathbf{r}_i)$ are assumed to come from a joint probability distribution $f(\mathbf{p}, \mathbf{r})$. In positioning applications, this may correspond to observing RSS measurements during routine activities where the position of the MC is known. Ultimately, this results in having survey points along the most traveled routes in the environment. The second choice, which is adopted by all existing systems, is a fixed design where the location of survey points is chosen by the designer. For the purposes of these experiments, we rely on

---

2. This data is available online at http://www.dsp.utoronto.ca/wlan.
3. http://www.netstumbler.org.

TABLE 3
Computational and Storage Complexity of Distance Calculations

| | Computations | | | | | Storage | |
|---|---|---|---|---|---|---|---|
| | addition | multiplication/division | exp | sorts (# of floats) | time (ms) | integers | floats |
| KNN | $3(K-1) + (2d-1)N'$ | $dN' + 2$ | 0 | $N'$ | 0.11 | 0 | $LN$ |
| Histogram | $3(K-1)$ | $2K + (2d-1)N' + 4$ | 0 | $N'$ | 0.13 | 0 | $bLN$ |
| Kernel | $2(dn+1)N'$ | $2(1+d)nN' + 2N' + 2$ | $nN'$ | 0 | 0.69 | $dnN$ | $2N$ |

Parameters $L$, $N$, $N'$, $n$, and $d$ are as defined in Table 1, where $b$ is the number of histogram bins and $K$ is the number of points used in averaging.

a fixed design to ensure comparability to existing methods in the literature.

In the absence of prior information regarding the location of the mobile user, the survey locations are chosen to lie on a uniform grid (worst case scenario). Unfortunately, realization of a uniform grid may not always be practical in an indoor environment because of the presence of walls, furniture, and other obstructions, preventing measurements in certain areas. Such a grid causes the estimation algorithm to produce estimates with variable resolution in different areas.

Another important parameter is the number of survey points $N$ and the spacing between such points. A dense grid leads to redundancy in the fingerprints and high training and computational costs, whereas a sparse grid may result in insufficient resolution. Existing works generally assume a spacing of 1 to 3 m in each direction between grid points. For the purposes of this work, a grid spacing of 2 m has been used. Fig. 4 shows the experimentation area layout and depicts the training points as black circles.

The training data was collected over two days at 66 points in the environment at a rate of 1 sample/sec. The experiments described in the following sections use 100 samples per survey point with the exception of Section 8.6.2, where we vary the number of samples from 4 to 200 to examine the effect of this parameter on positioning results. During training, the orientation of the laptop remained the same and is indicated by arrows in Fig. 4.
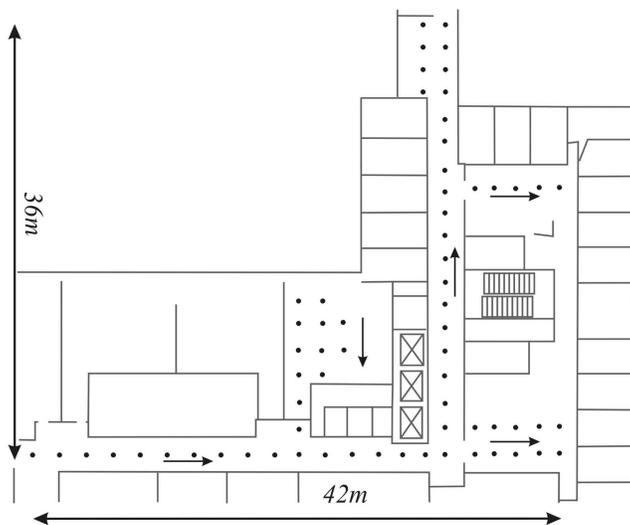


Fig. 4. Map of the experimentation environment.

### 8.1.3 Validation
The data for the validation set was collected on a different day than the training data and was used to tune system parameters. It is essential that such a novel set is used for validation to avoid overfitting in a time-varying environment that results from changing conditions such as number and position of people in the environment, doors opening and closing, and location of small furniture. A total of 60 samples were collected at each survey point at a rate of 1 sample/sec for the laptop orientation shown in Fig. 4.

### 8.1.4 Testing
Test measurements were collected over two days, separate from training and validation days, to capture a variety of environmental conditions. This novel set has been used to test the system under environmental conditions different than that of training and validation, reflecting the mismatch between training and testing conditions in the real-life operation of the system. The test samples were collected at 44 test points situated on and off the training points. Moreover, to test the sensitivity of the proposed method to the orientation of the receiver, test measurements were collected for four orientations of the laptop, namely, north, south, east, and west. A total of 60 samples were collected per test point per orientation at a rate of 1 sample/sec.

Due to the random variations in RSS over time, a true evaluation must consider not only a large number of test samples, but also the performance of the estimator for various survey sets. This, however, is not practically feasible due to the need for manual collection of data in the WLAN setting. Therefore, we consider performance evaluation over one training set to gain insight into the performance of the proposed methods and the effect of their parameters.

## 8.2 Figure of Merit
Since the position of the MC is reported in Cartesian coordinates, the positioning error is commonly reported as the $L_2$ norm of the difference between the true position and its estimate,

$$\epsilon_i(t) = \|\mathbf{p}_i(t) - \hat{\mathbf{p}}_i(t)\|, \tag{26}$$

where $\mathbf{p}_i(t)$ and $\hat{\mathbf{p}}_i(t)$ denote the $t$th instance of the $i$th test point and its estimate.

The aggregated positioning error is reported numerically as the Average Root Mean Square Error (ARMSE) since the RMSE measure provides an indication of both bias and variance of the estimator:
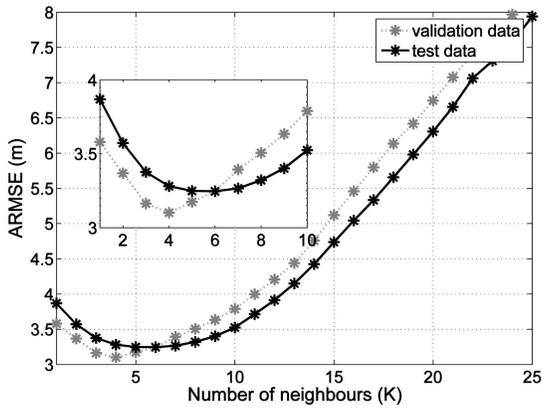
Fig. 5. Effect of the number of neighbors (*K*) on the performance of KNN.

$$\bar{\epsilon} = \frac{1}{N_\mathbf{p}} \sum_{i=1}^{N_\mathbf{p}} \sqrt{\frac{1}{N_t} \sum_{t=1}^{N_t} \epsilon_i^2(t)}, \tag{27}$$

$$= \frac{1}{N_\mathbf{p}} \sum_{i=1}^{N_\mathbf{p}} \sqrt{\frac{1}{N_t} \sum_{t=1}^{N_t} \|\mathbf{p}_i(t) - \hat{\mathbf{p}}_i(t)\|^2}. \tag{28}$$

In the above equations, $N_t$ and $N_\mathbf{p}$ are the numbers of test samples per point and testing points.

### 8.3 Comparison to Prior Work

The performance of the proposed method is compared to those of the KNN and histogram-based methods as described below.

#### 8.3.1 KNN

This method, first introduced in [16], has been widely used as a benchmark in WLAN positioning. Despite its simplicity, it has been shown to provide excellent positioning accuracy when compared to more sophisticated techniques such as SVMs [37]. This method computes the position estimate as the arithmetic average of $K$ points whose fingerprints are closest to the observation. Let the set $\{\mathbf{p}_{(1)}, \ldots, \mathbf{p}_{(N')}\}$ denote the ordering of survey points with respect to increasing distance between their sample mean and the observation. The position estimate is obtained as

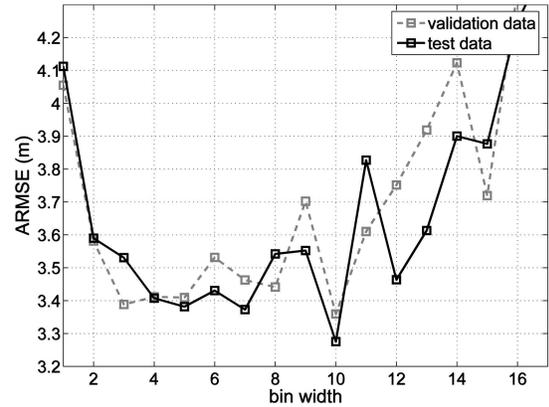$$\hat{\mathbf{p}} = \frac{1}{K} \sum_{i=1}^{K} \mathbf{p}_{(i)}. \tag{29}$$

Although weighted versions of the above equation have also been proposed, we note that these in fact correspond to special cases of the proposed method using different kernel functions. Fig. 5 shows the ARMSE as a function of the number of neighbors $K$ for the KNN method. It can be seen that $K = 4$ provides the lowest ARMSE on the validation data set and is therefore used throughout the rest of the experiments.

#### 8.3.2 Histogram-Based Techniques

Perhaps the most widely used WLAN positioning technique is the histogram-based method, which uses a histogram estimate of the probability density at each survey point to obtain a likelihood or posterior probability.



(a)



(b)

Fig. 6. Effect of system parameters on ARMSE for the histogram. (a) ARMSE versus the number of neighbors (*K*). (b) ARMSE versus histogram bin width.

In order to remedy the curse of dimensionality, the histogram estimates are computed independently in each dimension and the estimated probability is obtained as the product of the marginal values $\hat{f}(\mathbf{r}|\mathbf{p}_i) = \prod_{l=1}^{d} \hat{f}(r^l|\mathbf{p}_i)$. Let the set $\{\mathbf{p}_{(1)}, \ldots, \mathbf{p}_{(N')}\}$ denote the ordering of survey points with respect to decreasing $\hat{f}(\mathbf{r}|\mathbf{p}_i)$. The position estimate is then calculated as the weighted average of $K$ survey points with highest posterior probability [20], [28]:

$$\hat{\mathbf{p}} = \frac{1}{K} \sum_{i=1}^{K} \hat{f}(\mathbf{r}|\mathbf{p}_i) \mathbf{p}_{(i)}. \tag{30}$$

Note that $K = 1$ and $K = N$ correspond to the MAP and MMSE estimation given a uniform prior [28]. As seen in Fig. 6a, the performance of the histogram method improves as $K$ increases. Thus, the value of $K = 66$ is used for the histogram for the rest of the experiments.

Another important issue, raised by Roos et al. [28] but not addressed in the existing literature, is the determination of histogram parameters, namely, the origin and bin width. Since RSS values are integers and the histogram estimate has the lowest bias at the center of each bin, we have chosen the origin to be at $-99.5$ dBm. Furthermore, although theoretical methods for determining the histogram bin width exist [34], we have determined this parameter using the validation data as the domain specific knowledge indicates that this bin
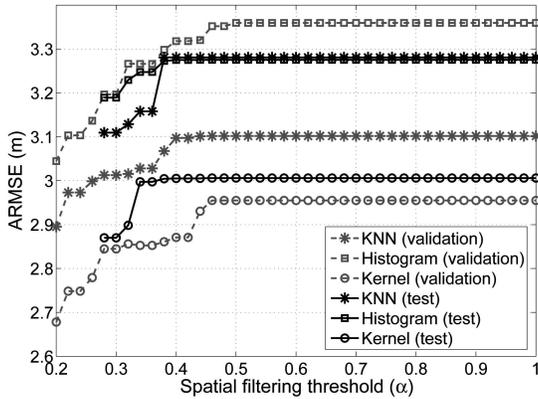
Fig. 7. ARMSE versus spatial filtering threshold $\alpha$ ($\alpha = 1$ corresponds to no filtering).

width value is an integer. Experimental results reported in Fig. 6b indicate that a bin width of 10 provides the lowest positioning error for the histogram for both validation and testing data sets. This value is consistent with the theoretical value of 13 approximated by Sturges's Rule [34], considering that this rule leads to an oversmoothed histogram. Moreover, such a large bin is needed to avoid overfitting and to cope with the variations in the environment and orientation during testing.
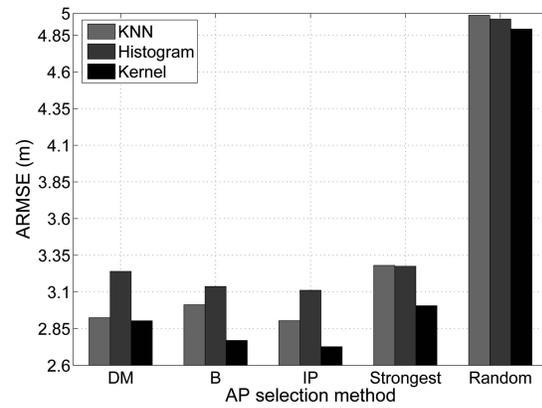
## 8.4 Spatial Filtering

The objective of this section is to examine the effects of spatial filtering on the overall positioning accuracy. To this end, positioning was carried out using the above setup with 100 training samples per point and three APs selected by the traditional method of strongest signal power. Fig. 7 shows the ARMSE for various values of the spatial threshold for the KNN, histogram, and kernel-based methods for both the validation and testing data.
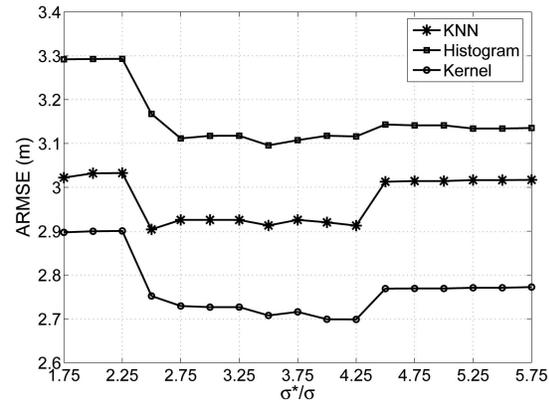
The results are consistent for the three positioning methods in that spatial filtering leads to an improvement in positioning accuracy (5.2, 2.6, and 4.5 percent for the KNN, histogram, and kernel methods, respectively). It can be seen that, for $\alpha < 0.2$ for validation data and $\alpha < 0.28$ for test data, no estimate is available as all survey points have been filtered out. This confirms that time variations in the environment may lead to the exclusion of all relevant survey points when small values of $\alpha$ are used. Moreover, a larger value of $\alpha$ is needed for the test data since points not on the original grid and different laptop orientations are considered in this set. The parameter $\alpha$ should thus be chosen slightly higher than that obtained during validation. Such an adjustment is unnecessary when the validation and test data cover the same conditions.

## 8.5 AP Selection

The effect of various AP selection methods on positioning accuracy and their interaction with the distance measurement methods have been examined for five methods, namely, distance of means (DM) or the Bhattacharyya distance with $\sigma_1 = \sigma_2 = 1$, the Bhattacharyya distance with sample variances (B), information potential (IP), selection of



(a)



(b)

Fig. 8. Effect of the AP selection method on ARMSE: DM, Bhattacharyya distance ($\sigma_i = 1$, $i = 1, 2$); B, Bhattacharyya distance ($\sigma_1 \neq \sigma_2$); IP, information potential; Strongest, strongest signal power; Random, random selection of APs. (a) ARMSE versus AP selection method. (b) Information potential method versus kernel width $\sigma$.

APs with the strongest mean, and random selection. The results for the five methods are shown in Fig. 8. Positioning has been carried out using 3 APs, 100 training samples per point, and no spatial filtering. AP selection has been performed on the five strongest APs to reduce computational complexity and ensure AP coverage on nearby survey points. This number was determined using the validation data.

As expected, the effectiveness of the AP selection method is dependent on the distance calculation technique used. In particular, we make the following observations:

- **Bhattacharyya distance**. The excellent performance of the Bhattacharyya distance with $\sigma_1 = \sigma_2 = 1$ seen in Fig. 8 is expected for the KNN method as these distances are directly related to the representative RSS value per survey point (sample mean). Maximizing such distances between APs therefore leads to decreased redundancy.

  The inclusion of variance in the Bhattacharyya distance leads to a gain in accuracy for the histogram and kernel as these methods use knowledge of the entire density during distance calculations. However, care must be taken when using variance estimates as this value is more susceptible to changes
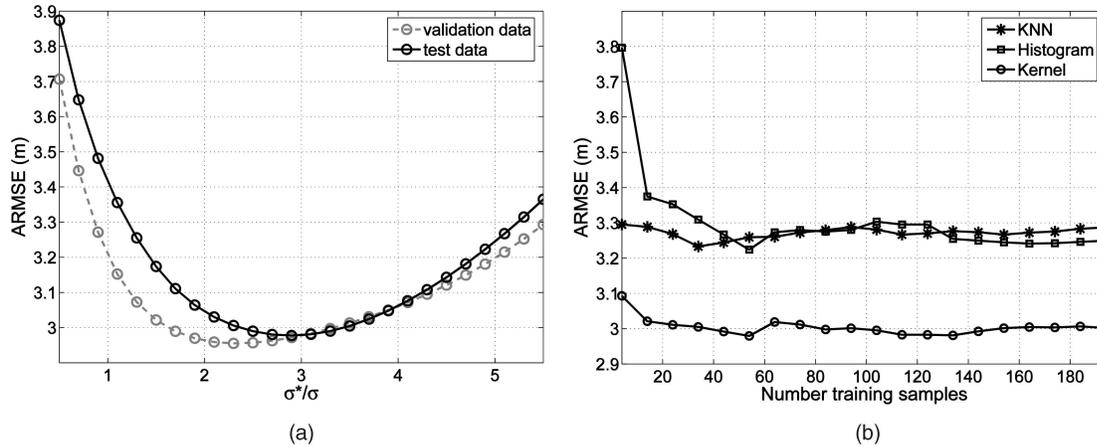
Fig. 9. Effect of system parameters on ARMSE. (a) ARMSE versus kernel width $\sigma$. (b) ARMSE versus number of training samples.

in a time-varying environment than the mean [25]. Thus, the sample variance estimated from the training data may not always be suitable for use with data collected at later times.

- **Information potential (IP)**. This method uses an estimate of the entire density to obtain a measure of divergence and should be contrasted to the other AP selection methods presented here that rely only on first and second moments. Fig. 8b shows the effect of the kernel width $\sigma$ used in the calculation of the IP measure on the positioning accuracy (kernel width for weight calculations remains constant). Although the IP measure has the capability to outperform the Bhattacharyya distance for the kernel method, it requires additional parameter tuning to determine the kernel width.

- **Strongest signal power**. The strongest signal method selects the APs with the strongest received signal power from the observation vector. The results reported in [25] indicate that RSS variance increases with an increase in mean RSS. The degradation in positioning accuracy for the KNN and kernel method is therefore attributed to the increased variance. Despite this, the strongest power method has the advantage that it increases the chance of AP coverage for survey points near the observation. This is especially important in the case of the histogram, as the estimate of the joint density is obtained as the product of marginal distributions. This makes the histogram sensitive to cases where no signal is received from an AP. Carrying out the above selection methods (DM, B, and IP) on a subset of $L' > d$ strongest APs therefore combines the advantages of both methods.

## 8.6 Distance Calculation

In this section, the positioning accuracy of kernelized weight functions is examined as a function of the parameters $\sigma$ and the number of training points per sample $n$ and compared to the euclidean distance method of KNN and the probabilistic distance of the histogram. The

following results were generated using three APs with strongest signal, 100 training points per sample, and no spatial filtering. For the purposes of the results presented in this section, similar results are obtained with different AP selection methods.

### 8.6.1 Kernel Width $\sigma$

Fig. 9a depicts the ARMSE as a function of kernel width $\sigma$ for both testing and validation data. Evidently, the simple bandwidth selection method of (21) introduces overfitting and hampers the performance on testing and validation data, where environmental conditions differ from that of training. It is encouraging, however, to note that the kernel width obtained during validation is near-optimal for the test data. This reiterates the importance of parameter tuning using a novel validation set. In our experiments, a kernel width two to four times that of (21) yields the best results.

### 8.6.2 Number of Training Points

Since the computation and storage complexity of the kernel-based distance measure increases with the number of training samples per survey point, we report the effects of this parameter on the positioning accuracy. Fig. 9b shows that the histogram method is more sensitive to the number of training samples used. Although both the KNN and proposed methods reach near-optimal results with as few as 10 samples per point, the performance of the histogram continues to improve as the number of samples increases. The work of Youssef et al. [26], for example, uses 300 samples (5 min, 1 sample/sec) for the histogram. Recall that the rate of convergence of the histogram density estimator in the AMISE is $O(n^{-2/d+2})$, which is slower than the $O(n^{-4/4+d})$ rate achieved by the KDE.

### 8.6.3 Kernelized Weights

The above results demonstrate the advantages of the kernelized distance over the KNN and histogram distance measurement techniques. Distance measurement in the kernel space leads to an improved accuracy of 8 percent (0.24 m), 6 percent (0.18 m), and 8 percent (0.28 m) over the KNN for the Bhattacharyya, IP, and strongest RSS AP
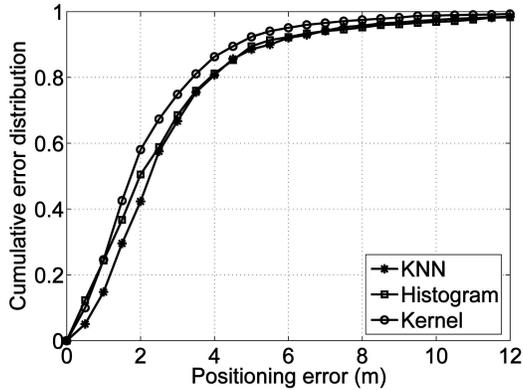
Fig. 10. Cumulative average error for KNN, histogram, and the proposed method.

selection methods. There is no improvement in the unit variance for the Bhattacharyya case due to the mismatch between the AP selection and kernelized distance measurement components, as previously discussed. The kernel method outperforms the histogram by 10 percent (0.34 m), 12 percent (0.37 m), 12 percent (0.38 m), and 8 percent (0.27 m) for each of the AP selection methods, respectively.

## 8.7 Comparison to Other Methods

Finally, we compare the positioning accuracy of the proposed system using all three steps of spatial filtering ($\alpha = 0.3$), AP selection (Bhattacharrya), and kernel-based weights ($\sigma = 2\sigma^*$) to that of the KNN (K = 4) and the histogram-based methods ($K = N, \text{bin width} = 10$) with no spatial filtering and strongest three APs. Fig. 10 depicts the comparison graphically, whereas Table 4 provides various error measures. The values in this table are obtained using parameter values obtained through validation with ARMSE used as the optimality criterion.

The proposed method leads to improvements of 0.57 m (17 percent) and 0.56 m (17 percent) over the KNN and histogram-based methods in the ARMSE sense. Table 4 also shows the ARMSE for the angular distance based method of (9). Clearly, this distance is ineffective in capturing the complexity of RSS distributions.

## 8.8 Summary of Recommendations

The experiments of the previous sections indicate that the complexity and accuracy of a positioning system is affected by the design of spatial filtering, AP selection, and distance calculation components. In particular, spatial filtering and

AP selection can be employed to reduce the computational complexity of the system by limiting the number of survey points and reducing the dimension of the measures, respectively. More importantly, we have discussed that the interplay between the AP selection and distance measurement components can be utilized to introduce significant improvements in positioning accuracy.

Several technical problems remain to be explored within the context of the proposed framework. We summarize these in the following points:

- This paper has considered the problem of AP selection in the input space. Investigation of efficient and real-time feature selection and dimensionality reduction methods in the kernel space can prove to be beneficial in future developments. Additionally, design of kernel functions geared toward the unique characteristics of the WLAN positioning problem (for example, correlated samples) must be examined.
- As power measurements are operator and device dependent, the sensitivity of existing methods to such parameters must be investigated. For example, one possibility for overcoming such limitations is the use of multiple fingerprints and fusion of results.
- The use of fingerprinting-based positioning solutions in noncooperative settings for enforcing network security requires centralized positioning, necessitating attention to secure transmission of location information, as well as mitigation strategies for hostile power fluctuations by the MC.
- As low-power and battery operated network devices such as Personal Digital Assistants (PDAs) proliferate, further research on power constrained implementations of WLAN positioning is necessary.

## 9 CONCLUSION

The rising prominence of location-based services necessitates the development of indoor positioning systems. Due to their ubiquity, WLANs provide a suitable and cost-effective infrastructure for devising such systems. This paper has examined the problem of WLAN positioning using RSS and has proposed a three-step training-based solution to manage the complexity of the indoor propagation environment. During the first step, spatial localization was proposed to address the variability of RSS patterns over space. Because the number of available APs is generally

TABLE 4
Various Error Measures (in Meters) for KNN, Histogram, the Proposed Framework, and the Angular Distance

| Method | RMSE | Mean Error | Variance | Median Error | 75 Percentile | 90 Percentile | ARMSE | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| | | | | | | | rooms | hallways |
| KNN | 3.28 | 2.74 | 8.53 | 2.00 | 3.20 | 5.50 | 2.98 | 3.58 |
| Histogram | 3.27 | 2.68 | 6.66 | 2.00 | 3.44 | 5.35 | 3.24 | 3.31 |
| Kernel-based | 2.71 | 2.31 | 4.57 | 1.73 | 3.01 | 4.58 | 2.76 | 2.65 |
| Angular distance | 10.32 | 10.27 | 15.90 | 10.49 | 13.15 | 15.47 | 9.81 | 10.84 |

much greater than the minimum three needed for positioning, the second step dealt with the problem of AP selection. Both parametric and nonparametric divergence measures were used to select a subset of the available APs, and the interaction between selection component and distance measurement was examined. Motivated by the complexity of RSS patterns, we proposed a kernelized measure for evaluation of similarity between an observation RSS vector and the training RSS records. The connection between the proposed kernelized functions and the nonparametric Nadaraya-Waston regression estimator was also discussed. Finally, the proposed methods were evaluated using real data from an office environment and shown to be effective as compared to the commonly used KNN and histogram-based solutions.

The nonparametric nature of the proposed methods makes them particularly suitable for application to other areas where probabilistic distributions of sensor patterns are unknown or complicated by operations such as quantization performed to meet power and bandwidth constraints. Future research directions therefore include the extension of methods proposed herein to distributed and constrained settings of sensor networks used for positioning and self-localization.

# APPENDIX A
## PROOF OF THEOREM 1
**Proof.**

1. Note that $w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) = \frac{1}{n}\sum_{t=1}^{n}\cos(\theta_i)$, where $\theta_i$ is the angle between the observation $\mathbf{r}$ and the $i$th training vector. Since RSS values reported in units of decibels below 1 mW are in the range $(-100, 0)$, RSS vectors lie in the same orthant and $0 \le \cos(\theta_i) \le 1$.
2. Proof follows from the linearity and scalability of the inner product.
3. The third property is proven using properties of the inner product:

$$d_{AE}(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))^2 = \frac{1}{n}\sum_{t=1}^{n}\left\|\frac{\mathbf{r}}{\|\mathbf{r}\|} - \frac{\mathbf{r}_i(t)}{\|\mathbf{r}_i(t)\|}\right\|^2$$

$$= \frac{1}{n}\sum_{t=1}^{n}\left\langle \frac{\mathbf{r}}{\|\mathbf{r}\|} - \frac{\mathbf{r}_i(t)}{\|\mathbf{r}_i(t)\|}, \frac{\mathbf{r}}{\|\mathbf{r}\|} - \frac{\mathbf{r}_i(t)}{\|\mathbf{r}_i(t)\|}\right\rangle$$

$$= \frac{\|\mathbf{r}\|^2}{\|\mathbf{r}\|^2} - \frac{2}{n}\sum_{t=1}^{n}\left\langle \frac{\mathbf{r}}{\|\mathbf{r}\|}, \frac{\mathbf{r}_i(t)}{\|\mathbf{r}_i(t)\|}\right\rangle + \frac{1}{n}\sum_{t=1}^{n}\frac{\|\mathbf{r}_i(t)\|^2}{\|\mathbf{r}_i(t)\|^2}$$

$$= 2 - \frac{2}{n}\sum_{t=1}^{n}\left\langle \frac{\mathbf{r}}{\|\mathbf{r}\|}, \frac{\mathbf{r}_i(t)}{\|\mathbf{r}_i(t)\|}\right\rangle = = 2(1 - w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)))$$

$$\Rightarrow w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) = -\frac{1}{2}(d_{AE}(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))^2 - 2).$$

4. Proof follows from the above relationship between $w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))$ and $d_{AE}(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))^2$. □

# APPENDIX B
## PROOF OF THEOREM 2
**Proof.**

$$d_k(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))^2 = \left\|\frac{\phi(\mathbf{r})}{\|\phi(\mathbf{r})\|} - \frac{1}{n}\sum_{t=1}^{n}\frac{\phi(\mathbf{r}_i(t))}{\|\phi(\mathbf{r}_i(t))\|}\right\|^2$$

$$= \left\langle \frac{\phi(\mathbf{r})}{\|\phi(\mathbf{r})\|} - \frac{1}{n}\sum_{t=1}^{n}\frac{\phi(\mathbf{r}_i(t))}{\|\phi(\mathbf{r}_i(t))\|}, \frac{\phi(\mathbf{r})}{\|\phi(\mathbf{r})\|} - \frac{1}{n}\sum_{t=1}^{n}\frac{\phi(\mathbf{r}_i(t))}{\|\phi(\mathbf{r}_i(t))\|}\right\rangle$$

$$= \left\langle \frac{\phi(\mathbf{r})}{\|\phi(\mathbf{r})\|}, \frac{\phi(\mathbf{r})}{\|\phi(\mathbf{r})\|}\right\rangle + \left\langle \frac{1}{n}\sum_{t=1}^{n}\frac{\phi(\mathbf{r}_i(t))}{\|\phi(\mathbf{r}_i(t))\|}, \frac{1}{n}\sum_{t=1}^{n}\frac{\phi(\mathbf{r}_i(t))}{\|\phi(\mathbf{r}_i(t))\|}\right\rangle$$

$$\quad -2\left\langle \frac{\phi(\mathbf{r})}{\|\phi(\mathbf{r})\|}, \frac{1}{n}\sum_{t=1}^{n}\frac{\phi(\mathbf{r}_i(t))}{\|\phi(\mathbf{r}_i(t))\|}\right\rangle$$

$$= \frac{k(\mathbf{r}, \mathbf{r})}{\sqrt{k(\mathbf{r}, \mathbf{r})k(\mathbf{r}, \mathbf{r})}} + \frac{1}{n^2}\sum_{t=1}^{n}\sum_{t'=1}^{n}\left\langle \frac{\phi(\mathbf{r}_i(t))}{\|\phi(\mathbf{r}_i(t))\|}, \frac{\phi(\mathbf{r}_i(t'))}{\|\phi(\mathbf{r}_i(t'))\|}\right\rangle$$

$$\quad -\frac{2}{n}\sum_{t=1}^{n}\left\langle \frac{\phi(\mathbf{r})}{\|\phi(\mathbf{r})\|}, \frac{\phi(\mathbf{r}_i(t))}{\|\phi(\mathbf{r}_i(t))\|}\right\rangle$$

$$= 1 + \frac{1}{n^2}\sum_{t=1}^{n}\sum_{t'=1}^{n}\frac{k(\mathbf{r}_i(t), \mathbf{r}_i(t'))}{\sqrt{k(\mathbf{r}_i(t), \mathbf{r}_i(t))k(\mathbf{r}_i(t'), \mathbf{r}_i(t'))}}$$

$$\quad -\frac{2}{n}\sum_{t=1}^{n}\frac{k(\mathbf{r}, \mathbf{r}_i)}{\sqrt{k(\mathbf{r}, \mathbf{r}), k(\mathbf{r}_i, \mathbf{r}_i)}}$$

$$= 1 + \frac{1}{n^2}\sum_{t=1}^{n}\sum_{t'=1}^{n}\frac{k(\mathbf{r}_i(t), \mathbf{r}_i(t'))}{\sqrt{k(\mathbf{r}_i(t), \mathbf{r}_i(t))k(\mathbf{r}_i(t'), \mathbf{r}_i(t'))}}$$

$$\quad - 2w_{RKHS}(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))$$

$$\Rightarrow w(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) = -\frac{1}{2}d_k(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))^2 + \frac{1}{2}$$

$$\quad + \frac{1}{n^2}\sum_{t=1}^{n}\sum_{t'=1}^{n}\frac{k(\mathbf{r}_i(t), \mathbf{r}_i(t'))}{\sqrt{k(\mathbf{r}_i(t), \mathbf{r}_i(t))k(\mathbf{r}_i(t'), \mathbf{r}_i(t'))}}.$$
□

# APPENDIX C
## PROOF OF THEOREM 3
**Proof.**

1. Using the property of the $\exp(\cdot)$, we note that

$$0 \le \exp\left(-\frac{\|\mathbf{r} - \mathbf{r}_i\|^2}{2\sigma^2}\right) \le 1 \Rightarrow$$
$$0 \le w_{SG}(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) \le (\sqrt{2\pi}\sigma)^{-d}.$$

2. Noting that $k(\mathbf{r}, \mathbf{r}') = \langle\phi(\mathbf{r}), \phi(\mathbf{r}')\rangle$, we have

$$w_{SG}(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)) = (\sqrt{2\pi}\sigma)^{-d}\frac{1}{n}\sum_{t=1}^{n}k(\mathbf{r}, \mathbf{r}_i(t))$$

$$= (\sqrt{2\pi}\sigma)^{-d}\frac{1}{n}\sum_{t=1}^{n}\langle\phi(\mathbf{r}), \phi(\mathbf{r}_i(t))\rangle$$

$$= (\sqrt{2\pi}\sigma)^{-d}\langle\phi(\mathbf{r}), \frac{1}{n}\sum_{t=1}^{n}\phi(\mathbf{r}_i(t))\rangle.$$

3. From Theorem 2

$$d_k(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))^2 = 1 - 2w_{RKHS}(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))$$
$$+ \frac{1}{n^2} \sum_{t=1}^{n} \sum_{t'=1}^{n} \frac{k(\mathbf{r}_i(t), \mathbf{r}_i(t'))}{\sqrt{k(\mathbf{r}_i(t), \mathbf{r}_i(t))k(\mathbf{r}_i(t'), \mathbf{r}_i(t'))}}$$
$$= 1 + \frac{1}{n^2} \sum_{t=1}^{n} \sum_{t'=1}^{n} \exp\left(\frac{-\|\mathbf{r}_i(t) - \mathbf{r}_i(t')\|^2}{2\sigma^2}\right)$$
$$- \frac{2}{n} \sum_{t=1}^{n} \exp\left(\frac{-\|\mathbf{r} - \mathbf{r}_i\|^2}{2\sigma^2}\right)$$
$$= 1 + \frac{1}{n^2} \sum_{t=1}^{n} \sum_{t'=1}^{n} \exp\left(\frac{-\|\mathbf{r}_i(t) - \mathbf{r}_i(t')\|^2}{2\sigma^2}\right)$$
$$- 2(\sqrt{2\pi}\sigma)^{-d} w_{SG}(\mathbf{r}, \mathbf{F}(\mathbf{p}_i)).$$

4. Proof follows from the above relationship between $w_{SG}(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))$ and $d_k(\mathbf{r}, \mathbf{F}(\mathbf{p}_i))^2$. □

## ACKNOWLEDGMENTS

## REFERENCES

[1] P. Enge and P. Misra, "Scanning Special Issue/Technology on Global Positioning System," *Proc. IEEE,* vol. 87, no. 1, 1999.

[2] M. McGuire, K. Plataniotis, and A. Venetsanopoulos, "Data Fusion of Power and Time Measurements for Mobile Terminal Location," *IEEE Trans. Mobile Computing,* vol. 4, no. 2, pp. 142-153, Mar.-Apr. 2005.

[3] T. Roos, P. Myllymaki, and H. Tirri, "A Statistical Modeling Approach to Location Estimation," *IEEE Trans. Mobile Computing,* vol. 1, no. 1, pp. 59-69, Jan.-Mar. 2002.

[4] CISCO Sytems, "Wi-Fi Based Real-Time Location Tracking: Solutions and Technology," http://www.cisco.com/, 2006.

[5] Ekahau, http://www.ekahau.com/, 2006.

[6] M. Rodriguez, J. Favela, E. Martinez, and M. Munoz, "Location-Aware Access to Hospital Information and Services," *IEEE Trans. Information Technology in Biomedicine,* vol. 8, no. 4, pp. 448-455, 2004.

[7] H. Harroud, M. Ahmed, and A. Karmouch, "Policy-Driven Personalized Multimedia Services for Mobile Users," *IEEE Trans. Mobile Computing,* vol. 2, no. 1, pp. 16-24, Jan.-Mar. 2003.

[8] C. Patterson, R. Muntz, and C. Pancake, "Challenges in Location-Aware Computing," *IEEE Pervasive Computing,* vol. 2, no. 2, pp. 80-89, 2003.

[9] G. Sun, J. Chen, W. Guo, and K.R. Liu, "Signal Processing Techniques in Network-Aided Positioning: A Survey of State-of-the-Art Positioning Designs," *IEEE Signal Processing Magazine,* vol. 22, no. 4, pp. 12-23, 2005.

[10] M. Hazas, J. Scott, and J. Krumm, "Location-Aware Computing Comes of Age," *Computer,* vol. 37, no. 2, pp. 95-97, Feb. 2004.

[11] J. Hightower and G. Borriello, "Location Systems for Ubiquitous Computing," *Computer,* vol. 34, no. 8, pp. 57-66, Aug. 2001.

[12] A. Harter and A. Hopper, "A Distributed Location System for the Active Office," *IEEE Network,* vol. 8, no. 1, pp. 62-70, 1994.

[13] M. Hazas and A. Hopper, "Broadband Ultrasonic Location Systems for Improved Indoor Positioning," *IEEE Trans. Mobile Computing,* vol. 5, no. 5, pp. 536-547, May 2006.

[14] N.B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The Cricket Location-Support System," *Proc. MobiCom,* pp. 32-43, 2000.

[15] M. Youssef and K. Agrawala, "Continuous Space Estimation for WLAN Location Determination Systems," *Proc. IEEE 13th Int'l Conf. Computer Comm. and Networks,* 2004.

[16] P. Bahl and V. Padmanabhan, "RADAR: An In-Building RF-Based User Location and Tracking System," *Proc. INFOCOM,* vol. 2, pp. 775-784, 2000.

[17] K. Plataniotis and C. Regazzoni, "Visual-Centric Surveillance Networks and Services," *IEEE Signal Processing Magazine,* special issue on video and signal processing for surveillance networks and services, vol. 22, no. 2, 2005.

[18] X. Li and K. Pahlavan, "Super-Resolution TOA Estimation with Diversity for Indoor Geolocation," *IEEE Trans. Wireless Comm.,* vol. 3, no. 1, pp. 224-234, 2004.

[19] D. Niculescu and B. Nath, "VOR Base Stations for Indoor 802.11 Positioning," *Proc. MobiCom,* pp. 58-69, 2004.

[20] M. Yousief, "Horus: A WLAN-Based Indoor Location Determination system," PhD dissertation, Univ. of Maryland, 2004.

[21] A. Goldsmith, *Wireless Communications.* Cambridge Univ. Press, 2005.

[22] R. Singh, L. Macchi, C. Regazzoni, and K. Plataniotis, "A Statistical Modelling Based Location Determination Method Using Fusion in WLAN," *Proc. Int'l Workshop Wireless Ad-Hoc Networks,* 2005.

[23] Z. Xiang, S. Song, J. Chen, H. Wang, J. Huang, and X. Gao, "A Wireless LAN-Based Indoor Positioning Technology," *IBM J. Research and Development,* vol. 48, nos. 5/6, pp. 617-626, 2004.

[24] P. Krishnan, A. Krishnakumar, W.-H. Ju, C. Mallows, and S. Ganu, "A System for LEASE: Location Estimation Assisted by Stationary Emitters for Indoor RF Wireless Networks," *Proc. INFOCOM,* vol. 2, pp. 1001-1011, 2004.

[25] K. Kaemarungsi and P. Krishnamurthy, "Properties of Indoor Received Signal Strength for WLAN Location Fingerprinting," *Proc. First Ann. Int'l Conf. Mobile and Ubiquitous Systems: Networking and Services (MOBIQUITOUS '04),* pp. 14-23, 2004.

[26] M. Youssef, A. Agrawala, and A.U. Shankar, "WLAN Location Determination via Clustering and Probability Distributions," *Proc. First IEEE Int'l Conf. Pervasive Computing and Comm.,* pp. 143-150, 2003.

[27] Y. Chen, Q. Yang, J. Yin, and X. Chai, "Power-Efficient Access-Point Selection for Indoor Location Estimation," *IEEE Trans. Knowledge and Data Eng.,* vol. 18, no. 7, pp. 877-888, July 2006.

[28] T. Roos, P. Myllymki, H. Tirri, P. Misikangas, and J. Sievnen, "A Probabilistic Approach to WLAN User Location Estimation," *Int'l J. Wireless Information Networks,* vol. 9, no. 3, pp. 155-164, 2002.

[29] K. Kaemarungsi and P. Krishnamurthy, "Modeling of Indoor Positioning Systems Based on Location Fingerprinting," *Proc. INFOCOM,* vol. 2, pp. 1012-1022, 2004.

[30] R.O. Duda, P.E. Hart, and D.G. Stork, *Pattern Classification,* second ed. Wiley, 2001.

[31] A. Haeberlen, E. Flannery, A.M. Ladd, A. Rudys, D.S. Wallach, and L.E. Kavraki, "Practical Robust Localization over Large-Scale 802.11 Wireless Networks," *Proc. MobiCom,* pp. 70-84, 2004.

[32] M. Youssef and A. Agrawala, "The Horus WLAN Location Determination System," *Proc. Third Int'l Conf. Mobile Systems, Applications, and Services,* pp. 205-218, 2005.

[33] A.M. Ladd, K.E. Bekris, A. Rudys, G. Marceau, L.E. Kavraki, and D.S. Wallach, "Robotics-Based Location Sensing Using Wireless Ethernet," *Proc. MobiCom,* pp. 227-238, 2002.

[34] D.W. Scott, *Multivariate Density Estimation.* John Wiley & Sons, 1992.

[35] J. Pan, J. Kwok, Q. Yang, and Y. Chen, "Accurate and Low-Cost Location Estimation Using Kernels," *Proc. Int'l Joint Conf. Artificial Intelligence,* pp. 1366-1370, 2005.

[36] J.J. Pan, J.T. Kwok, Y. Chen, and Q. Yang, "Multidimensional Vector Regression for Accurate and Low-Cost Location Estimation in Pervasive Computing," *IEEE Trans. Knowledge and Data Eng.,* vol. 18, no. 9, pp. 1181-1193, Sept. 2006.

[37] M. Brunato and C. Kallo, "Transparent Location Fingerprinting for Wireless Services," Technical Report DIT-02-0071, Dipt. Informatica e Telecomunicazioni, Univ. of Trento, Sept. 2002.

[38] A. Kushki, K. Plataniotis, and A.N. Venetsanopoulos, "Location Tracking in Wireless Local Area Networks with Adaptive Radio Maps," *Proc. IEEE Int'l Conf. Acoustics, Speech, and Signal Processing (ICASSP '06),* vol. 5, pp. 741-744, 2006.

[39] E. Gokcay and J. Principe, "Information Theoretic Clustering," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 24, no. 2, pp. 158-171, Feb. 2002.

[40] B. Schölkopf and A.J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond.* MIT Press, 2002.

[41] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis.* Cambridge Univ. Press, 2004.

[42] M.G. Genton, "Classes of Kernels for Machine Learning: A Statistics Perspective," *J. Machine Learning Research,* vol. 2, pp. 299-312, 2001.

[43] R. Battiti, M. Brunato, and A. Villani, "Statistical Learning Theory for Location Fingerprinting in Wireless LANs," Technical Report DIT-020086, Dipt. Informatica e Telecomunicazioni, Univ. of Trento, Oct. 2002.

[44] B.W. Silverman, *Density Estimation for Statistics and Data Analysis.* Chapman and Hall, 1986.

[45] R. Jenssen, D. Erdogmus, J. Principe, and T. Eltoft, "The Laplacian Spectral Classifier," *Proc. IEEE Int'l Conf. Acoustics, Speech, and Signal Processing (ICASSP '05),* vol. 5, pp. 325-328, 2005.

**Azadeh Kushki** (S'02) received the BASc and MASc degrees from the Department of Electrical and Computer Engineering at the University of Toronto in 2002 and 2003, respectively. She is currently working toward the PhD degree with particular emphasis on radio and video positioning, multimedia systems, adaptive signal processing, and data fusion methods. She is a student member of the IEEE.

**Konstantinos N. Plataniotis** (S'90-M'92-SM'03) received the BEng degree in computer engineering and informatics from the University of Patras, Greece, in 1988 and the MS and PhD degrees in electrical engineering from Florida Institute of Technology (Florida Tech) in Melbourne, Florida, in 1992 and 1994, respectively. He is now an associate professor with the Edward S. Rogers Sr. Department of Electrical and Computer Engineering at the University of Toronto, Ontario. He is a registered professional engineer in the province of Ontario and a member of the Technical Chamber of Greece. He is the 2005 recipient of IEEE Canada's Outstanding Engineering Educator Award "for contributions to engineering education and inspirational guidance of graduate students" and the corecipient of the 2006 *IEEE Transactions on Neural Networks* Outstanding Paper Award for the paper "Face Recognition Using Kernel Direct Discriminant Analysis Algorithms" (Juwei Lu, Konstantinos N. Plataniotis, and Anastasios N. Venetsanopoulos, *IEEE Transactions on Neural Networks*, vol. 14, no. 1, 2003). He is a senior member of the IEEE.

**Anastasios N. Venetsanopoulos** (S'66-M'69-SM'79-F'88) received the BS degree in electrical and mechanical engineering from the National Technical University of Athens (NTU) in 1965 and the MS, MPhil, and PhD degrees in electrical engineering from Yale University in 1966, 1968, and 1969, respectively. He is currently the vice president for research and innovation at Ryerson University, Canada. He has served as the 12th dean of the Faculty of Applied Science and Engineering of the University of Toronto, as a lecturer in 138 short courses to industry and continuing education programs, and as a consultant to numerous organizations. He is a coauthor of four and a contributor to 35 books and has published more than 800 papers in refereed journals and conference proceedings. He is a fellow of the IEEE, a fellow of the EIC, and a fellow of the Canadian Academy of Engineering. He has served as chair on numerous boards, councils, and technical conference committees of IEEE. In 2003, he served as the chair of the Council of Deans of Engineering of the Province of Ontario (CODE). He was awarded an honorary doctorate by the National Technical University of Athens in October 1994. In October 1996, he was awarded the "Excellence in Innovation Award" of the Information Technology Research Centre of Ontario and the Royal Bank of Canada "for innovative work in color image processing and its industrial applications." He is the recipient of the Millennium Medal of the IEEE. In 2003, he was the recipient of the highest award of the Canadian IEEE and the MacNaughton Award. He is also a corecipient of the 2006 *IEEE Transactions on Neural Networks* Outstanding Paper Award.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/publications/dlib.