Digital Wall: A Power-efficient Solution for Location-based Data Sharing

Jeffrey Junfeng Pan, Sinno Jialin Pan, Vincent Wenchen Zheng and Qiang Yang
Department of Computer Science and Engineering
Hong Kong University of Science and Technology, Hong Kong
panjf,sinnopan,vincentz,qyang@cse.ust.hk

Abstract

With the proliferation of wireless and sensor techniques, data can be shared conveniently through the air. However, wireless communication is vulnerable since unauthorized machine may try to intrude a server without being physically connected. In this paper, we wish to control the communication between a wireless client and the infrastructure based on the client location. Our idea is to implement a digital wall, which is a user-defined boundary so that access is allowed within the boundary and denied outside the boundary. To do this, we need to do accurate location estimation since the decision around the boundary line is critical. Furthermore, computational efficiency is also important since we need to reduce computation cost so as to save power energy. In this paper, we propose k-nearest-neighbor (KNN) based method to determine the location of a mobile client based on received signal strength (RSS) values. We further use information gain as a feature selection criterion to reduce the estimation time. We study how the performance may be affected when the controlled area is confined by physical or virtual walls. Experimental results show that we can well distinguish whether a client device is located within an expected digital wall.

1 Introduction

As the pervasive environments become common, user activity context now turns out to be a sensitive issue for protecting user's privacy [11] and the network security [19]. The research community becomes increasingly interested in the context sensitive access control, which use different information, such as identity, location, for determining whether a user should be authorized for retrieving data from a network. Among those used information, location is a primary piece for context-aware computing, considering a usual case to confine data communication and sharing within an expected area. This suggests a location-based access control.

Digital wall can be treated as a special application of location based access control, which extends the notion of privacy provided by physical walls to the virtual realm. A *digital wall* is a user-define boundary so that access is allowed inside the boundary and denied outside.

Essentially, *Digital wall* is a virtual boundary as proposed in [8] for securing the users' privacy in wireless networks. However, in [8], work was focused on developing a policy language to formalize the semantics of access control using virtual wall. Other similar systems include *Digital Territory project* [3], *pawS* [11], *etc*. They all left the problem unsolved of actively detecting user to decide whether the user should be authorized for data sharing. That is exactly the issue we will address.

Accurate location estimation is needed for solving *digital wall* since the estimation shall be sensitive around the virtual boundary. In this paper, we propose k-nearest-neighbor (KNN) based method for determining whether a client is inside or outside a customized space. More specially, our solution has two phases: an offline training phase and an online localization phase. In the training phase, we customize a virtual boundary and collect data inside and outside the boundary. In the localization phase, KNN approach is used to make a decision based on signal strength values detected in real time.

Meanwhile, computational efficiency is also important in *digital wall*, which shall respond quickly to the change of client locations. Specially, the connection shall be cut shortly after the client leaves the virtual space to protect privacy. Estimated location shall be updated frequently, which may involve too much computational overhead. To solve this problem, we use information gain to selectively pick up a portion of access points for localization.

In this paper, we study how the performance can be affected when the controlled area is confined by physical or virtual walls. We set up several real world experiments to study the performance of our method on actively detecting different locations on both sides of a digital wall. We also vary the number of access points used for localization based on information gain. Experimental results show that we can

0-7695-3113-X/08 $25.00 © 2008 IEEE
DOI 10.1109/PERCOM.2008.81

well distinguish whether a client device is located within an expected *digital wall*.

2 Related Works

2.1 Location based Access Control

For location based access control using proximity, some limited-range radio communication mechanisms are used. In [2], Balfanz *et al.* proposed using location-limited channels to bootstrap a wide range key-exchange protocols. Some other work also proposed the use of limited-range radio broadcasts as a way to verify proximity [15]. A drawback for such systems is that an attacker could relay the authentication exchanges to gain unauthorized access, and thus the security is not strongly guaranteed.

For location based access control using round trip time, Brands and Chaum [18] described the first distance bounding protocol based on timing the single-bit round trip delay in a cryptographic challenge-response exchange. In [17], Sastry *et. al* proposed an *Echo* protocol. Each time it chooses a verifier sending a packet to a prover via radio frequency (RF), which is echoed back using ultrasound. Then the round trip time is calculated for deciding whether to reject the location claim. A drawback of this protocol is that it needs to use both WiFi RSS and ultrasound to execute the verification. It is more expensive than our method, which use only WiFi RSS data for location detection.

2.2 Localization in Wireless Networks

Location estimation systems based on radio signal strengths can be classified into two main categories [1]: (1) *Radio-Propagation Models* (2) *Empirical-Fit Models*. The latter may not rely on the knowledge of radio propagation.

Tracking systems that adopt *Propagation-based Models* benefit from the knowledge of radio propagation [5]. The locations of access points need to be given in many radio propagation models. Martin *et al.* [12] explored that different access points should behave similarly. Their work presents a Bayesian network model that encodes knowledge about radio-propagation models, which makes use of similarity among the access points and other factors. It also points out that, by incorporating additional knowledge such as the motion constraint of a user, the calibration effort can be further reduced. To reduce the number of hardware used for localization, [24] developed a Device-free Passive localization system by monitoring and processing changes in the received physical signals to detect object location changes in the environment.

Another class of location estimation systems is based on *Learning-based Models*, which employ machine learning techniques. Signal patterns can be captured when sufficient empirical data are manually collected at different locations [23]. The locations of access points are not needed. Typical pattern descriptions include histogram [23], mixture of gaussian[16], kernel matrix [13], or simply the mean value of signal strength at different locations [1]. For example, the LEASE system [10] uses reference tags to dynamically construct and update radio map. A similar technique is used in [22]. To address the problem of data distribution variation, [14] proposes a multi-view based manifold regularization method, called LeManCoR. It tries to apply previous out-of-date RSS samples to the new time slice training.

2.3 Energy Efficiency

In pervasive computing area, there are two major critiria to measure the performance of a system: One concerns accuracy of location estimation, as mentioned in the previous subsection, and the other concerns energy efficiency or power efficiency. This is because the client devices are usually small, maintained by the constrained battery power. Thus, how to save energy while acheiving high location estimation accuracy has atracted more and more attention from researchers. There are two research directions for saving compution energy. One focuses on various hardwares power management such as the disk [9, 6] and CPU [20]. The other foucuses on reducing communication and compution cost [21, 7, 4]. [4] proposed a multiple-decision-tree based approach,*CaDet*, to select an appropriate subset of access points. Localization algorithms are based on the selected access points instead of all the access points so that the computation cost can be reduced.

3 Our solution

3.1 System Architecture

We can abstract and classify different RSS-based localization models by what is carried in the object. In the *receiver-oriented model*, the object carries a receiver and receives the messages from different transmitters in the external infrastructure. The object can estimate its distances to these transmitters by measuring the RSS values on received messages. Based on at least three different distance estimations, the object computes its own location. The receiver-oriented model is a natural extension of the GPS mechanism without using timing information. It is mainly used in 802.11-based localization systems, in which the object is usually a person carrying a notebook computer or Personal Digital Assistant (PDA) with 802.11 card and the transmitters are the access points. Installing *digital wall* on client site is possible. The client itself can detect signal strengths, estimate its location and determine whether it shall response to the infrastructure and send out data packets. However, the

infrastructure may not trust in the location the client claims. In such case, we need to verify the client location from the infrastructure without relying on the computation from the client device.

The *transmitter-oriented model* exchanges the roles of transmitters and receivers in the receiver-oriented model. After receiving the messages from the transmitter carried by the object, the receivers in the external infrastructure estimate the distances to the object. By collecting these distance estimations, the external infrastructure computes the location of the object without involving the object in computation. This model is best fit if the infrastructure wants to track a client device without changing anything on that device and to verify the location the client claims.

3.2 Problem Statement

We define the *digital wall* problem in a *receiver-oriented* model as follows. Assume that there are N access points fixed in an area that we are interested in. These access points periodically send out beacon signals. The locations of the access points are not necessarily known. There are one or more *mobile* devices of unknown locations. At time t, a *mobile* device can measure the RSS sent by the N access points by detecting their signals which give a vector $\mathbf{s}_t = (s_{t1}, s_{t2}, \ldots, s_{tN})' \in \mathbb{R}^N$. In addition, we have collected l labeled training data which are signal-location pairs $\{(\mathbf{s}_{t_i}, l_{t_i})\}_{i=1}^{l}$ at two locations. One location is inside a digital wall and the other is outside.

Our objective is to determine the location $l'_t \in \{inside, outside\}$ of a *mobile* node based on the signal vector $\mathbf{s}_t$ measured at time t.

3.3 Power-efficient Classification

Our solution to the *digital wall* has two phases : an offline training phase and an online localization phase. In the offline phase, we customize the digital wall, collect data and select access points based on a feature selection criterion. In the online phase, we apply k nearest neighbor method to determine whether the current location of a mobile device is inside or outside the digital wall.

3.3.1 • Offline Training Phase

1. Customize the digital wall in an area we are interested in. It may be physical walls around a room or virtual ones inside a room.

2. Collect l labelled signal-location pairs at two locations. One location is inside the digital wall and the other is outside. Denote the dataset as $\{(\mathbf{s}_{t_i}, l_{t_i})\}_{i=1}^{l}$.

3. Compute and rank Information Gain for the N access points. Pick up the top M discriminative access points which Information Gains are higher than some threshold θ. Denote the subset of M access points as F.

The discriminative power of an access point is measured by the Information Gain when its value is known. Specifically, it is calculated as the reduction in entropy as:

$$InfoGain(AP_i) = H(G) - H(G|AP_i)$$

where $H(G)$ is the entropy of the two sampling locations and $H(G|AP_i)$ is the conditional entropy when the value of AP_i is known.

In such a way, we can pick up the topmost useful access points and avoid computational overhead.

3.3.2 • Online Localization Phase

1. At time t, the *mobile* device collects a signal vector $\tilde{\mathbf{s}}_t$. For those access points that are far away, which signals are too weak to detect, we fill in a small value, e.g. -100dBm.

2. Pick up the signal values that are in the subset of selected access points F.

3. Apply k nearest neighbor method and determine whether the mobile device is inside the digital wall or outside.

3.4 System Prototype

We implement a prototype of *digital wall* in *receiver-oriented model*. It is a convenient platform that runs in Windows XP. It supports general wireless cards and has useful functions for map building, signal collecting, location labelling and signal-based tracking.

Data collection is a time consuming and error-prone process. A convenient tool is needed to relief our burden and reduce potential error. More specifically, our WiFi experimental platform for *digital wall* has the following features:

- **Support of general wireless devices**. Our platform is developed in Windows XP. It uses Network Driver Interface Specification User Mode Input/Output (NDISUIO) to detect beacon frames from different access points and measure their signal strengths. NDISUIO is initially used by Wireless Zero Configuration Service and is supported by most wireless adaptors.

- **Multi-layer map building tool**. We can create, drag, rescale, remove, load and save maps conveniently.

- **Label access points and client devices**. By supporting a map, we can mark down the current locations of client devices and the readings of signal strengths. We can also selectively mark down the locations of access points on the map. We visualize all labelled access points and client devices.
- **Received-signal-strength based Tracking**. We implement two tracking algorithms based on RSS readings. One is a simple propagation model that relies on the locations of access points. Another is a KNN method which uses the calibrated data from client devices.

Although the system is developed in Windows XP, the kernel codes are mostly ready for migrating to other operating systems since the platform-dependent codes are well separated from the kernel modules.

4 Experiments

4.1 Experimental Setup

We perform our experiments in the laboratory area of the Computer Science and Engineering Department in the Academic Building of the Hong Kong University of Science and Technology (HKUST) as shown in Figure 1. The mobile device used to collect training and test data is a T40 IBM laptop. There are totally sixteen different access points detected in the laboratory area. We evaluate our solution in two different scenarios: *physical wall* and *virtual wall*.

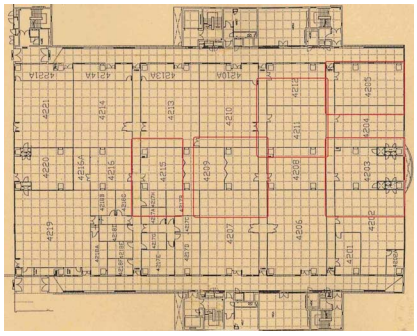

Figure 1. Test Bed in the laboratory area of the Computer Science and Engineering Department

For physical wall scenario, we test our system in four offices. For each office, we use the laptop to collect RSS records in a fixed location in-office and another fixed location out-of-office as training data. For testing, we collect RSS records at various locations in-office and out-of-office as test data. Our goal is to predict whether the laptop is in office or out of office. For virtual wall scenario, we test our system around a council board in a office. We imagine that there were a virtual wall in the middle of the council board, as shown in Figure 2. We collect RSS records in a fixed location in either side of the boundary as training data, separately. For testing, we collect several RSS records in various locations in each side of the boundary (as shown in Figure 2, we collect test data in location a, b, ...,n). Our goal is to predict whether the laptop is in the side above the boundary or below the boundary. In order to measure the performance, we use location prediction accuracy as the evaluation criterion.

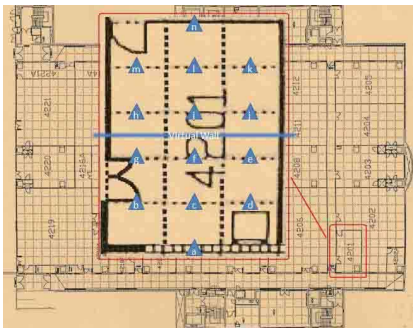

Figure 2. Virtual boundary in an office

4.2 Experimental Result

4.2.1 Experiments on Physical Wall

The first two experiments evaluate the performance of our system in the physical wall scenario. In each office region, our system first apply *CaDet* [4] to select important access points, then based on the selected access points, our system uses k-nearest-neighbor algorithm to predict locations of the laptop.

Figure 3 shows the average performance of our system in different distances from the laptop to the door of the office. We can see that, our system can get high accuracy when the distance from the laptop to the door is equal to or lager than 2 meters. That means when a user walks into an office and is not nearby the door, our system can support wireless services correctly.

Figure 4 shows the effort of the number of access points on average performance of different office regions, where we fix the distance between the laptop and the door being 2 meters. In order to reduce the computation cost used to locate the mobile device, we expect to select as few as access points to achieve a high accuracy. It can be seen from the figure that when the number of access points equals to eleven, our system can achieve the best accuracy of 87%.

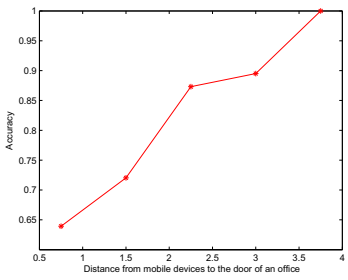

Figure 3. Location estimation accuracy versus the distance from the laptop to the door of an office

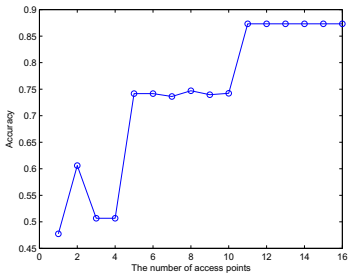

Figure 4. The number of APs versus average accuracy in physical wall

4.2.2 Experiments on Virtual Wall

In this section, we evaluate the performance of our system in the virtual wall scenario. We first apply *CaDet* to selecte access points, then use k-nearest-neighbor algorithm to predict locations. Figure 5 shows the performance of the system in different the distance from the laptop to the virtual boundary. It can be seen that the performance of the system is not high when the laptop is nearby the boundary. However, when the distance from the laptop to the boundary is larger than or equal to 2 meters, our system can get around 85% accuracy.

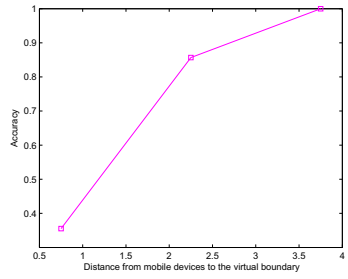

Figure 5. Location estimation accuracy versus the distance from the laptop to the virtual boundary

In the above experiment, we use the first five access points selected by *CaDet*. The impact of the number of access points on performance is shown in Figure 6, where we fixed the distance from the laptop to the boundary being around 2 meters. We can see that selecting access points not only can reduce the computation cost but also improves accuracy. The performance using five access points is much larger than that using all access points.

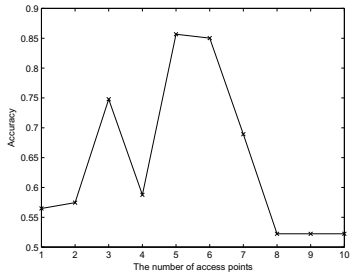

Figure 6. The number of APs versus accuracy in virtual wall

4.3 Multiple Digital Walls

We have shown experiments of our solution on either physical walls or single virtual wall scenario. Our solution can be extended to multiple virtual walls or hybrid walls (both virtual walls and physical walls). That means we can build a security area by multiple virtual walls or hybrid walls. A mobile device is supported with sepecial wireless services when it is within this area. In principle, the area can be any shape. However, how to achieve high performance and reduce as much energy in this case is very challenging. We put this in our future work.

5 Conclusion

In this paper, we describe *digital wall*, a power-efficient solution for location-based data sharing. To apply this system, we first define a digital wall in an interesting area. In the offline training phase, we collect training data at two locations. One is inside the digital wall and the other outside. After that, Information Gain is used to select the most useful access points and reduce computational overhead. In the online localization phase, we apply k nearest neighbor to determine a mobile client is inside a digital wall or not. We vary serveral parameters to verify the effectiveness of the algorithm. We test our method in different rooms, the wall of which may be physical or virtual. We vary the number of access points and the sampling locations. Experimental results show that we can well distinguish whether a client device is located within an expected *digital wall*.

6. Acknowledgement

This work is supported by NEC China Lab (NECLC05/06.EG01). The authors thank Junhui Zhao for useful discussion.

References

[1] P. Bahl and V. Padmanabhan. RADAR: An in-building RF-based user location and tracking system. In *Proceedings of the Conference on Computer Communications*, volume 2, pages 775–784, Tel Aviv, Israel, March 2000.

[2] D. Balfanz, D. Smetters, P. Stewart, and H. Wong. Talking to strangers: Authentication in adhoc wireless networks, Feburary 2002.

[3] L. Beslay and H. Hakala. Digital territory: Bubbles. *European Visions for the Knowledge Age*, 2005.

[4] Y. Chen, J. Yin, and X. Chai. Power-efficient access-point selection for indoor location estimation. *IEEE Transactions on Knowledge and Data Engineering*, 18(7):877–888, 2006.

[5] A. Goldsmith. *Wireless Communications*. Cambridge University Press, Cambridge, 2005.

[6] S. Gurumurthi, A. Sivasubramaniam, M. Kandemir, and H. Franke. Drpm: dynamic speed control for power management in server class disks. In *Proceedings of the 30th annual international symposium on Computer architecture*, pages 169–181, New York, NY, USA, 2003. ACM Press.

[7] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *HICSS*, 2000.

[8] A. Kapadia, T. Henderson, J. Fielding, and D. Kotz. Virtual walls: Protecting digital privacy in pervasive environments. In *Proceedings of the Fifth International Conference on Pervasive Computing*, volume 4480, pages 162–179, May 2007.

[9] R. Kravets and P. Krishnan. Power management techniques for mobile communication. In *Mobile Computing and Networking*, pages 157–168, 1998.

[10] P. Krishnan, A. S. Krishnakumar, W. Jun, C. Mallows, and S. Ganu. A system for LEASE: Location estimation assisted by stationary emitters for indoor rf wireless networks. In *Proceedings of the Conference on Computer Communications*, Hong Kong, 2004.

[11] M. Langheinrich. A privacy awareness system for ubiquitous computing environments. In *Proceedings of the 4th international conference on Ubiquitous Computing*, pages 237–245, London, UK, 2002. Springer-Verlag.

[12] D. Maligan, E. Elnahrawy, R. Martin, W. Ju, P. Krishnan, and A. Krishnakumar. Bayesian indoor positioning systems. In *Proceedings of the Conference on Computer Communications*, volume 2, pages 1217–1227, Miami, FL, USA, March 2005.

[13] J. J. Pan, J. T. Kwok, Q. Yang, and Y. Chen. Accurate and low-cost location estimation using kernels. In *Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence*, pages 1366–1371, Edinburgh, Scotland, 2005.

[14] S. J. Pan, J. T. Kwok, Q. Yang, and J. J. Pan. Adaptive localization in a dynamic wifi environment through multi-view learning. In *Proceedings of the Twenty-Second National Conference on Artificial Intelligence*, pages 1108–1113, Vancouver, Canada, 2007.

[15] K. B. Rasmussen, S. Capkun, and M. Cagalj. Secnav: secure broadcast localization and time synchronization in wireless networks. In *Proceedings of the 13th annual ACM international conference on Mobile computing and networking*, pages 310–313, New York, NY, USA, 2007. ACM Press.

[16] T. Roos, P. Myllymaki, H. Tirri, P. Misikangas, and J. Sievanen. A probabilistic approach to WLAN user location estimation. *International Journal of Wireless Information Networks*, 9(3):155–164, July 2002.

[17] N. Sastry, U. Shankar, and D. Wagner. Secure verification of location claims. In *Proceedings of the 2nd ACM workshop on Wireless security*, pages 1–10, New York, NY, USA, 2003. ACM Press.

[18] D. C. Stefan Brands. Distance bounding protocols. In *Workshop on the Theory and Application of Cryptographic Techniques*, pages 344–359, 1994.

[19] P. Tao, A. Rudys, A. M. Ladd, and D. S. Wallach. Wireless lan location-sensing for security applications. In *Proceedings of the 2003 ACM workshop on Wireless security*, pages 11–20, New York, NY, USA, 2003. ACM Press.

[20] M. Weiser, B. Welch, A. J. Demers, and S. Shenker. Scheduling for reduced CPU energy. In *Operating Systems Design and Implementation*, pages 13–23, 1994.

[21] Y. Xu and W.-C. Lee. On localized prediction for power efficient object tracking in sensor networks. In *ICDCSW '03*, page 434, Washington, DC, USA, 2003. IEEE Computer Society.

[22] J. Yin, Q. Yang, and L. Ni. Adaptive temporal radio maps for indoor location estimation. In *Proceedings of the 3rd Annual IEEE International Conference on Pervasive Computing and Communications*, pages 85–94, Kauai Island, HI, USA, 2005.

[23] M. Youssef, A. Agrawala, and U. Shankar. WLAN location determination via clustering and probability distributions. In *Proceedings of the First IEEE International Conference on Pervasive Computing and Communications*, pages 143–150, Fort Worth,TX, USA, March 2003.

[24] M. Youssef, M. Mah, and A. Agrawala. Challenges: device-free passive localization for wireless environments. In *Proceedings of the 13th annual ACM international conference on Mobile computing and networking*, pages 222–229, New York, NY, USA, 2007. ACM Press.