Journal of Advanced Computing and Communication Technologies *(ISSN:* **2347 - 2804***)*
Volume No.1 Issue No. 1, August 2013

Privacy in Location-Based Services using SP-Filtering in Hide and Seek Protocol with Obfuscation- Based Methods

By

Munmun Sharma

California State University East Bay,USA
Munmun.kinshuk@gmail.com

ABSTRACT
Privacy in Location based services has always been a question in terms of Security and Privacy. And lately there has been some work done on Privacy part of Location Based Services (LBS). This Privacy known name as "Privacy aware Proximity based services." In Privacy aware Proximity based services; two mobile users will check if they're in proximity to every alternative while not revealing their actual locations . This paper will describe the implementation of one such protocol which is used for preserving the privacy of user's exact location called "Hide and Seek Protocol using SP-filtering Protocol".

Keywords

Location-Based Services, Privacy, Proximity, Obfuscation Based Methods, SP-Filtering, Hide and Seek Protocol.

1. INTRODUCTION
LBS are position dependent services which are available for IP capable devices and are used to identify the location of a user or object. These services can be either Push based or Query based services. Apart from other technologies used in LBS, positioning the mobile terminal requires specific infrastructure. Positioning is defined as location of an object either in co-ordinate system, route system or areal division. Here in this paper we will be considering Geocoding in order to make reference as coordinates to objects [1].

In LBS, Positioning is defined in 3 main classes Satellite Positioning, Network Positioning and Local Positioning. Satellite Positioning is an infrastructure used between terminals and earth orbiting satellites. Positioning is calculated with the help of 3 or more satellites which sends radio signals to terminals. This method provides 10-40 m accuracy. In Network Based Positioning, Position is calculated on the basis of signals sent by three or more base stations and received by terminal. Local Positioning is used for restricted areas like buildings, malls etc. Where the satellites signals and network based positioning signals are not precise. Some examples can be: Bluetooth[2], WLAN[3], RFID[4] and many more.[1]

2. ARCHITECTURE

In order to ensure that location privacy is enforced, architecture is needed to be described. It is important that Privacy services should not interfere the existing services, it is better to apply privacy thus implementation can be done in service layer. Figure 1 shows service layer will follow a client server architecture model [5].

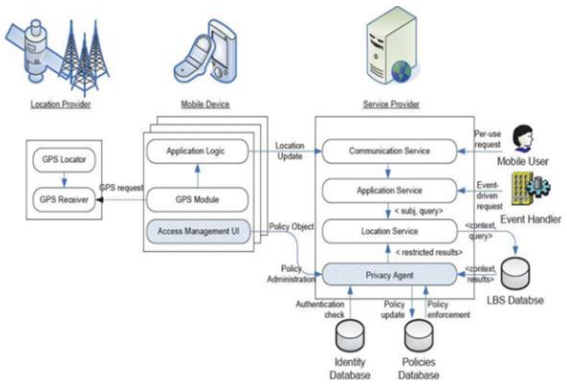

Fig. 1. Architecture of Location Based Services

Client takes the responsibility to allow users to manage privacy policies and server helps in providing authorization to location based services. Here is the description of architecture and relationship of modules with the sub modules. The system architecture of location based services comprises of 3 main components, Location Provider, Mobile Device and Service Provider. Location Provider contains GPS receiver and GPS Locator. GPS Receiver receives GPS request from mobile devices. GPS Locator takes the responsibility of computing the actual physical location. When all these results are computed, they are sent back to the mobile devices which

requested the query. In Mobile Devices all the devices which consist of GPS capabilities like phones and PDAs are included in Mobile Devices. Location updates are sent by these mobile phones to respective service Providers. Once these updates are sent to respective service providers, these service providers will provide protections that are required in LBS. Application logic takes the responsibility for managing the communication between service provider and mobile devices. Application Logic also helps in sending the updates from mobile devices and receiving event notification from SP. Second component is Access Management User Interface. This interface helps mobile users to manage own policies for privacy like creating and updating the policies. It also communicates directly to Privacy Agent Module through secure channel .Context elements and policy objects are sent via secure communication channel. Location based Services are provided by Service Provider to its subscribed mobile users. Services provided can be either event driven or per-use requests. It has Communication Services, Application services Location Services and Privacy agent. And these communicate through Users, Event Handlers, LBS Database, Authentication Database and Policies Database. [5]

In the next sections, Classification of Privacy Preserving Technique are explained, including implementation of Obfuscation Based methods, SP-Filtering and Hide & Seek Protocol.

3. PRIVACY PRESERVING TECHNIQUES

Numbers of attacks are possible in Location Based Services. Some examples are Multi Query Attacks which includes Shrink Region Attack and Region Intersection Attack [6], Location-Based Quasi-Identifiers [7], Passive Attacks, Active Attacks including Replay Attacks, Timing Attacks and Result Tempering Attacks [8].

As shown in Figure 2 LBS privacy methods are divided into 2 main categories Trusted Third Party Based (TTP-based) and Trusted Third Party–Free Based (TTP-Free based).

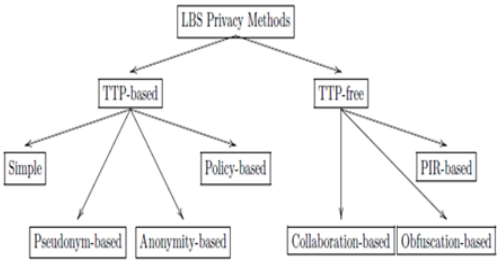

Fig. 2. Classification of LBS Privacy methods

TTP Based schemes are are very common because they are easy to understand/develop, and because they offer a reasonable trade-off between efficiency, accuracy and privacy. These can be classified into Simple scheme, Policy-based schemes, Pseudonym- Based and Anonymity- Based. [9]

TTP-free schemes solves the drawbacks of TTP-Based Schemes (i) the architecture relies on a TTP, so that the user relay the platform mediating between him and the LBS provider; (ii) it is assumed that LBS providers are not malicious but semi-honest, which might turn out to be too much of an idealization; and (iii) the architecture is centralized, which makes it vulnerable to DoS attacks. There are various methods which are used by TTP-free schemes like Collaboration-based methods, PIR- based methods and Obfuscation-based methods.[9] Some of the Protocols which works on TTP- Free based schemes are, Boneh Protocol - 3[10],Hide and Crypt Protocol [4] and other protocols which are TTP based are Boneh Protocol-2 [10] Homomorphic Encryptions [11].

In the coming sections Obfuscation Based methods, SP-Filtering, Hide and Seek Protocol and their Implementation are explained.

4. OBFUSCATION BASED METHODS

In Obfuscation Based Methods, the space is modeled as a graph where vertices are locations and edges indicate adjacency. Hence, in order to obtain an imprecise location, the user sends a set of vertices instead of the single vertex in which he is located. The LBS provider cannot distinguish which of the vertices is the real one. The article proposes negotiation algorithms that allow users to increase the QoS whilst maintaining their privacy. The main problem of this technique is that users and providers must share the graph modeling the space or other methods where the real location of LBS users is replaced by circular areas of variable center and radius. The main advantages of this method are: (i) no TTP and no collaboration are needed; (ii) the closest interest point is always found; (iii) the location of the user is hidden in a controlled area. However, due to the shortage of collaboration, this technique isn't ready to attain the k-anonymity and/or the l-diversity properties.[12]

In the coming Sections , we will explain how this Obfuscation Based Methods will be used during the Implementation of SP-Filter Protocol and Hide & Seek Protocol.

5. SP- FILTERING PROTOCOL

SP-Filtering Protocol is called a Three Party Protocol where there are three parties Alice, Bob and Service Provider (SP).SP act as a server between 2 clients Alice and Bob. This Protocol is used to provide proximity between 2 buddies Alice and Bob. Privacy provided by this protocol is a minimum level of Privacy and computes the Proximity of Bob to Alice [13].

Here is the working of the protocol. Whenever Alice updates her location, she doesn't send the exact location to the Service Provider. Instead, she will send a generalized location to the service Provider.

This generalized location is computed as a function of G^{U}_{A} and the granule $G^{SP}_{A}(i)$ where A is located. More precisely, A sends to SP the location $L_A(i)$ that is computed as the union of the granules of G^{U}_{A} that intersects with $G^{SP}_{A}(i)$. [13]

$$L_A(i) = \bigcup_{i' \in N | GUA(i') \cap GSPA(i) \neq \Theta}$$

Similarly Bob will also update his location and the index j will be used and the location will be defined in $G^{SP}_{B}(j)$. It is the responsibility of Service Provider to compute the $L_B(j)$.After this is done Service Provider Computes the distances between $L_A(i)$ and $L_B(j)$. Since these are granules, there will be 2 distances. One will be d which will be the minimum distance between $L_A(i)$ and $L_B(j)$ and another will be D which is maximum distance between $L_A(i)$ and $L_B(j)$.

There is another term called δ_A which explains the preferred proximity of Alice .Now three conditions establishes which are shown in Figure 3. A) If $D < \delta_A$, then Bob is in proximity of Alice.

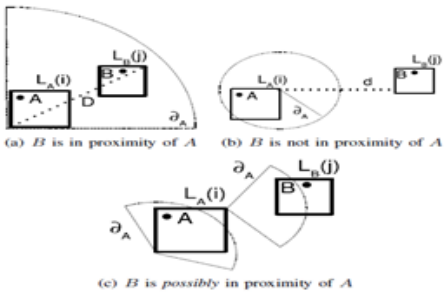

Fig 3. Regions LA and LB

B) If $D > \delta_A$, then Bob is not in proximity of Alice. C) If $d \leq \delta_A \leq D$, then Bob Possibly be in Proximity of Alice. Condition C arises because exact location of Alice and Bob is not known. Bob can reside anywhere in L_B (j).If Bob is located in bottom left corner of L_B (j) and Alice is located in upper right corn or of L_A (i), then Bob will be in proximity of Alice. Otherwise it is not.

SP- filtering works very well in the Conditions A and B. But for condition C it is not able to show the accurate results. Therefore, In order to solve the Problem of Sp-Filtering, the SP-protocol can be combined with more accurate method of Hide and Seek which is discussed in the next section.

6. HIDE AND SEEK PROTOCOL

To remove the drawback of SP-Filtering Protocol (When it declares Bob possibly in proximity of Alice), we are combining the Hide and Seek Protocol in order to provide accurate proximity results.

In this paper, we are considering Hide and Seek Protocol as a 2-party protocol. In 2- party protocol, first Alice will send to Bob the values of i' and δ_A. Here i' is granule where G^U_A is located and δ_A is the preferred proximity of Alice. Now, Bob will compute minimum distance between any 2 points $G^U_A(i')$ and $G^U_B(j')$,where $G^U_B(j')$ is the granule where B is located. This minimum distance will be given by d'. Now 2 conditions will arise A) If $d' > \delta_A$ then Bob is not in Proximity B) If $d' < \delta_A$ then Bob is not in Proximity.[13]

In the next Sections, we will explain how these, Obfuscation method, SP-Filter Protocol and Hide and Seek Protocol are combined and implemented together, in order to provide proximity results.

7. IMPLEMENTATION

Implementation of Hide and Seek starts with the implementation of SP-Filtering Protocol. There will be 2 users A and B and a Service Provider. User A won't send his exact location to the Service Provider. Instead she will send a generalized location to A. Now two questions arise, who is going to decide the scope of granules and how the granules will be defined? Here are the answers. User A herself will decide the scope of granularity through her mobile device. Second, The granules will be decided using the obfuscation method [12]. Here we will be using co-ordinate system to define the scope of granularity.

To start with the implementation, there will requirement for a couple of statistics. First, In order to know, weather the buddy Bob is in Proximity or not, there will be a requirement of preferred proximity which is δA [4]. Second the exact location of Alice in the form of latitude and longitude.

After these two fields are known to the mobile device, another eight co-ordinates will be generated in a form of square around the exact location defining granularity of A i.e. $G^U_A(i)$.

δA is in the form of Distance (eg. Miles). In order to get the other co-ordinates, it needs to change into degrees. 1° of Latitude is 69 miles and 1° of Longitude is 53 miles.

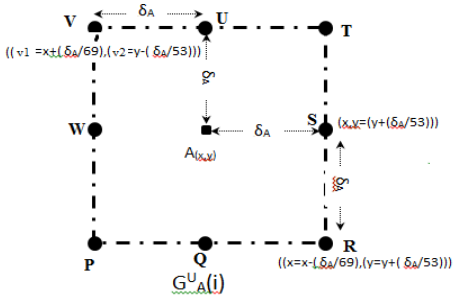

Fig 4. Defining Granularity of A

As shown in Figure 4, with the help of A(x,y) , eight co-ordinates are generated. Taken A as center and δA as radius, first 4 points are generated. U, W, S and Q. Now, with the help of these four points another four points will be calculated. Since, now U and W are already known, point V (v1, v2)can be calculated.

v1= x+(δA/69) and v2= y- (δA/53)

Similarly P, V and R can be found out. Once all the 8 points are calculated, then all the eight points are sent to Service Provider (SP).

Similarly User B will also send his eight points in the form of G^U_B (j) to the Service Provider. Once the service provider gets all the 16 points, implementation of SP-Filtering Protocol begins.

SP-Filtering Protocol works on the principle of comparing distance. Once the comparison is made, results are declared in the form of In-Proximity, Might-be in Proximity and not in Proximity [13].

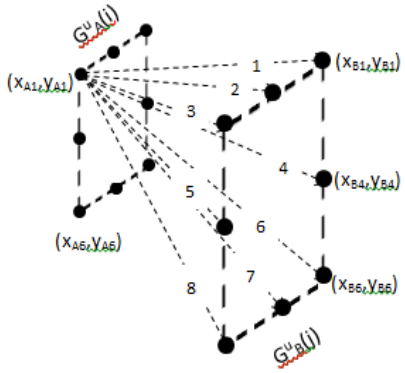

Fig 5. Distance calculation by Service Provider

Figure 5 above shows the description of how the distances are calculated among these sixteen co-ordinates.

Distance 1=$\sqrt{(xA1-xB1)^2+(yA1-yB1)^2}$

Each point of G^U_A (i) will be compared to each point of G^U_B (j). For example, if point (X_{A1}, Y_{A1}) are taken then distance of (X_{A1}, Y_{A1}) are compared to all eight points of G^U_B (j).As given in figure distance are shown from number 1 to 8.

When distance of each point with every point is calculated, there will be total 64 distances calculated. From the set of these 64 distances, Maximum Distance (D) and Minimum Distance (d) is calculated.

Once all the distances D and d are calculated, Proximity Calculation begins. If D≤ δ_A then result shows "B is in proximity of A". If d≥δ_A then "B is not in Proximity of A". If d≤ δ_A≥D the "B might be in Proximity of A". [13]

First two conditions are straight forward in SP-Filtering Protocol. Problem arises when there is third condition i.e. might be in-proximity condition. Because in Might be Condition User B may be or may not be in proximity shown in Figure 3.

To remove this problem, Protocol Hide and Seek comes in action. When there is a might be condition, the last thing done by Service provider is that, it will divide the whole area into smaller grids and index is provided to the grids. Grids are given index number and size of grid is similar to the Granules of B i.e. G^U_B (j).

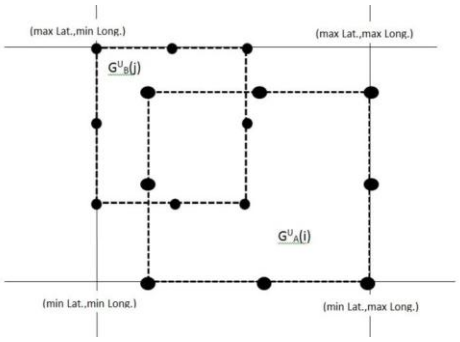

Fig. 6 Defining of Area Gamma for dividing into the grids

Figure 6 shows the Area which will be divided into the grids. Area will be defined in the form of square, where both G^U_A (i) and G^U_B (j) are residing. Co-ordinates of this square will be defined as maximum latitude and minimum longitude of G^U_B (j) to minimum latitude and maximum longitude of G^U_A (i) and from minimum latitude and minimum longitude of G^U_A (i) and G^U_B (j) to maximum latitude to maximum longitude of G^U_A (i) and G^U_B (j).

As of now, the area Gamma of further division is defined. Now the actual division into grids will take pace as shown in figure 7 and index will be provided.

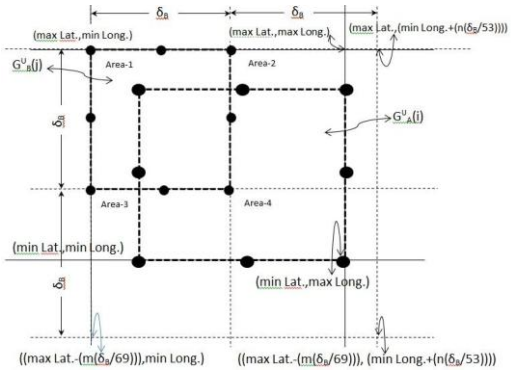

Fig 7. Division of area into smaller grids

Area1, Area2, Area 3 and Area4 are the four grids into which area Gamma is divided. Table 1 below shows the index numbers and area defined by the index with their coordinates. For implementation purpose, only one co-ordinate is used to define the area of grid. Here values of m and n represents the number of granules in the area Gamma.

Table 1: Defining and Naming the index of the grids

Area	Index	m	n	Latitude	Longitude
Area 1	1	1	1	max Lat.of GUB	min Long.of GUB
Area 2	2	1	2	Max.Lat.of GUB	min Long of GUB +n((δB/53))
Area 3	3	2	1	max.Lat.of GUB(j)-(m(δ_B/69))	min.Long.of GUB(j)
Area 4	4	2	2	max.Lat.-(m(δ_B/69))	min.Long.+(n(δ_B/53))

When all the grids are defined G^U_A (i´) will be Area 1, area 2, area 3 and Area 4 as A's exact location is not known and G^U_B (j') will be definitely area 1. Now these G^U_A (i´) and G^U_B (j') will be sent by SP to user A and User B with message "B might be in proximity". From this point Hide and Seek protocol will start.

After receiving this message, A will start a 2 party protocol and send B the value of i´ and δ_A where i´ is the grid where A is located. Suppose A is in area 3, then it will send index 3 to B.

Now B will calculate the minimum distance between any 2 points of G^U_A (i´) and G^U_B (j') and will be given the name d´. If d´ is greater than or equal to δ_A then result will be send by B to A that B is not in proximity else it will send B is in proximity.

In the next coming sections, Code, Screenshots ,Analysis , Road blocks and conclusion are described.

8. ANALYSIS

Analysis of project Hide and Seek includes the time to calculate the proximity of user B from A. For this purpose we have chosen clock to find out the time. The Performance analysis is done on the basis of Table 2 :

Table 2: Parameters used in calculating performance

Parameter	Values
Proximity (δ) in Miles(M)	1, 3 , 5, 10
Area of GUA($M^{2)}$	4 ,36,100 ,400
Number of Buddies in Group	2,3,4,5,6,7,8,9,10

Fig. 11. shows the performance of SP-Filtering Protocol. This is done so as to analyze the time in calculating the server side computation.X-axis shows number of buddies and Y-axis shows the computation time in milliseconds. It can be seen that as the number of buddies increases computation time increases and thus the time increases.

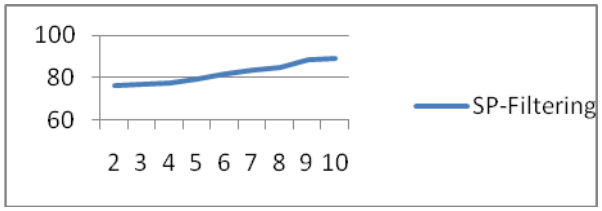

Fig. 11. Performance of SP-Filtering Protocol

Fig. 12. Shows the performance of Hide and Seek Protocol . This is done to understand the client to client communication performance. It can be seen fro mthe table that as the number of buddies increases , computation time also increases.

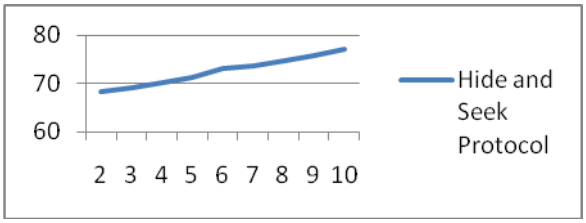

Fig. 12. Performance of Hide and Seek Protocol

Fig. 13 shows the comparison between 2 protocol performance. It can be seen that both the performance are increasing linearly but server side time computation is more as they are dealing with big granules , where as Hide and seek computes the area of index and that too only one ppoint, which is considered small.

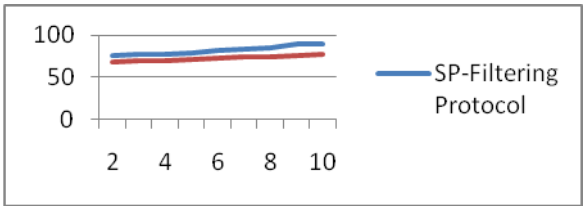

Fig 13. Comparison between Hide and Seek and SP-Filtering Protocol

9. REFERENCES

[1] Stefano Berretti, Alberto Del Bimbo, Pietro Pala, Francisco Jos`e SilvaMata ,"Face Recognition By SVM Classification Of 2D And 3D Radial Geodesics" University of Firenze, Italy IEEE 2008.

[2] Nils J. Nilsson, "Introduction to Machine Learning", Department of Computer Science Stanford University, December 1996.

[3] Jun-Ying Gan, Si-Bin He, "Face Recognition Based on 2DLDA and Support Vector Machine", Proceedings of the 2009 International Conference on Wavelet Analysis and Pattern Recognition, Baoding, IEEE 2009, pp.211-214.

10. [4] Praseeda Lekshmi V, Dr. M Sasikumar, Divya S Vidyadharan "Facial Expression Classification from Gabor features using SVM" **.**

[5] Du Hongle, Teng Shaohua, Zhu Qingfang, "Fast SVM Incremental Learning Based on Clustering Algorithm", College of Mathematics

Luoyang normal university Luoyang, China, IEEE 2009, pp.13-17.

[6] Qian-Ying Chen, Qiang Yang' "Segmentation Of Images Using Support Vector Machines" Proceedings of the Third International Conference on Machine Learning and Cybernetics, Shanghai, 26-29 August 2004.

[7] M. Palaniswami, A. Shilton, D. Ralph and B.D. Owen " Machine Learning Using Support Vector Machines " The University of Melbourne, Victoria-3101, Australia.

[8] Ethem Alpaydin, "Introduction To Machine Learning"**,** The MIT Press Cambridge, Massachusetts London, England, 2004 Massachusetts Institute of Technology.

[9] Shigeo Abe, "Support Vector Machines for pattern Classification" , Springer-Verlag London Limited 2005.

[10] Kebin Cui, Yingshuag Du, "Application of Boolean Kernel Function SVM in Face Recognition",2009 International Conference on Networks Security, Wireless Communications and Trusted Computing.

[11] Xia Sun, Qingzhou Zhang, Ziqiang Wang, " Face Recognition Based on NMF and SVM ", 2009 Second International Symposium on Electronic Commerce and Security.

[12] Yi - Min Wen and Bao - Liang Lu, "Incremental Learning of Support Vector Machines by Classifier Combining", Department of Computer Science and Engineering, Shanghai Jiao Tong University.

[13] Stefan R.Uping. "Incremental Learning with Support Vector Machines"**,** Department of Computer Science, AI Unit University of Dortmund, Germany. ICDM 2001.

[14] Du Hongle, Teng Shaohua, Zhu Qingfang, "Fast SVM Incremental Learning Based on Clustering Algorithm", IEEE 2009, pp. 13-17.

[15] Yuanzhi Wang, Fei Zhang, Liwei Chen, "An Approach to Incremental SVM Learning Algorithm", 2008 ISECS International Colloquium on Computing, Communication, Control, and 2008 ISECS International Colloquium on Computing, Communication, Control, and Management, IEEE 2008, pp.352-354.

[16] Aouatif Amine, Sanaa Ghouzali, Mohammed Rziza, Driss Aboutajdine, "Investigation of Feature Dimension Reduction based DCT/SVM for Face Recognition", 2008 IEEE, pp. 188-193.

[17] Guo-Yun Zhang, Shi-Yu Peng, Hong-Min Li "Combination Of Dual- Tree Complex Wavelet And SVM For Face Recognition", Proceedings of the Seventh International Conference on Machine Learning and Cybernetics, Kunming, 12-15 July 2008.

[18] Praseeda Lekshmi V, Dr. M Sasikumar, Divya S Vidyadharan "Facial Expression Classification from Gabor features using SVM" **.**

[19] Zhong-We Li, Jian-Pe Zhang, Jing Yang, "A Heuristic Algorithm To Incremental Support Vector Machine Learning", Proceedings of Third International Conference on Machine Learning and Cybernetics, Shanghai, IEEE 2004, pp.1764-1767.

[20] Jin-Long An, Zhengou Wang, Zhen-Ping Ma, "An Incremental Learning Algorithm For Support Vector Machine", Proceedings of the Second International Conference on Machine Learning and Cybernetics, San, IEEE 2003, pp.1153-1156.

[21] Pei Jiang, Yongjie Li, "A Hybrid Face Recognition Algorithm Based on WT, NMFs and SVM", University of Electronics Science and Technology of China, IEEE 2008, pp.734-737.

[22] Xiaoguang Lu, "Image Analysis for Face Recognition", Dept. of Computer Science & Engineering Michigan State University, East Lansing.

[23] C. C. Chang and C.J. Lin. "LIBSVM - A Library for support Vector Machines" Available on: http://www.csie.ntu.edu.tw/~cjlin/libsvm/ 2008.

[24] A. Shilton, M. Palaniswami, "Incremental Training of Support Vector Machines"Available on: http://people.eng.unimelb.edu.au/shiltona/publications/increment.pdf/2008