Information Technology Journal 9(8): 1571-1584, 2010
ISSN 1812-5638

A Novel Vision-based Location Authentication Approach in a Ubiquitous Camera Environment

Hsien-Chou Liao and Po-Ching Lee
Department of Computer Science and Information Engineering, Chaoyang University of Technology,
168 Jifong E. Rd., Wufong Township Taichung County, 41349, Taiwan, Republic of China

Abstract: Location-based services (LBSs) have been emerging in recent years. Mobile devices with built-in GPS receiver (Global Positioning System), called GPS-enabled mobile devices, will also become an important trend in the near future. Location information and authentication of mobile clients are critical for accessing desired LBSs. In this paper, a novel vision-based approach, called VLocAuth, is proposed to incorporate a ubiquitous camera (UbiCam) environment for location authentication. The operation of VLocAuth relies on three kinds of servers, including LBS server, authentication server (AS), and camera server (CS). VLocAuth consists of two phases: initialization and location authentication phases. In the initialization phase, a mobile client requests user identification and related information from LBS server via a secure channel. In the location authentication phase, a temporal identification provided by LBS server is used for data communication among the client, AS, and CS to achieve privacy protection. Then, CS authenticates the location of a mobile client by matching the location and the moving objects in the real-time camera image to ensure the client at the location of the GPS coordinates. The security analysis shows that VLocAuth is secure against replay attack and man-in-the-middle attack. It also satisfies unforgeability, privacy, confidentially, integrity, simplicity, and practicability requirements. A simulation study is designed to show the influence of a node density, type of GPS receiver, and a network delay on VLocAuth. Besides, a matching tool is implemented for measuring the time of the key matching step of VLocAuth in a practical environment. These results show that VLocAuth is feasible for location authentication and meets the trend of UbiCam environment in the near future.

Key words: system security, context-aware security, privacy protection, visual tracking, ubiquitous computing.

INTRODUCTION

The evolution of ICT (information and communication technology) and hardware technology brings about the development of ubiquitous computing. Many countries include ubiquitous computing into their industrial policies, such as U-Korea, U-Japan, and U-Taiwan (Ubiquitous Taiwan). Among the services related to ubiquitous computing, location-based service (LBS) is one of the fastest growing areas in the mobile world. LBSs include security, information, navigation, tracking, and many other services. This was achieved by the removal of the signal-degrading selective availability (SA) from the GPS (global positioning system) signals on the 1st May 2000 (Murakami and Ke, 2006). Many GPS-enabled devices, such as GPS phone, GPS PDA, are currently popular and become important tools with which to access LBSs. For LBSs, the location of a mobile client is very critical information for accessing LBSs. Previous studies have mainly focused on the protection of privacy (Ren *et al*., 2006; Ren and Wenjing, 2007; Al-Muhtadi *et al*., 2002). However, if an attacker or a legal user forges the location information, he can access desired information or services illegally via LBSs. Here are some examples:

- **The violation of personal privacy**: Generally, the access to personal medical records is mainly based on the username and password. If the access is restricted at some specific locations, e.g., home or office, to increase information security, an attacker can forge location information to access such records; the privacy could be violated in such a way.
- **The fraud of m-payment**: Mobile commerce (m-commerce) is getting more and more popular. Mobile payment (m-payment) is an important transaction of m-commerce. If the transaction is committed based on the location authentication, an attacker could forge location information for fraud purpose.

Corresponding Author: Hsien-Chou Liao, Department of Computer Science and Information Engineering,
Chaoyang University of Technology, 168 Jifong E. Road, Wufong Township Taichung County, 41349,
Taiwan, Republic of China Tel: +886-4-23323000/4211 Fax: +886-4-23742375

- **The illegal access of digital contents**: Location information can be used to control the access or distribution of digital contents. For example, Han *et al*. (2004) proposed a protocol for digital rights management (DRM) based on location authentication in a ubiquitous computing environment. An attacker could forge location information to access digital contents illegally.
- **The breach of data security**: Several location-based data encryption approaches have been proposed for data transmission among mobile users. For example, Scott and Denning (2003) proposed a GPS-based data encryption method, called Geo-Encryption. Liao and Chao (2008) proposed a location-based data encryption algorithm (LDEA). The receiver can only decrypt the encrypted data at a specific location specified by the sender. However, an attacker can still forge the location information for breaching data security of such an approach.

From the above examples, we conclude that the location authentication should prevent the attacks from malicious attackers. In previous studies, the authentication is mainly based on trusted devices of a third party, such as hardware sensors or tamper proof GPS modules for the location authentication of a cell phone (Durresi *et al*., 2007). However, the deployment of trusted hardware sensors is not cost-efficient. In addition, a mobile client must be close to a sensor for location authentication. It is also inconvenient for users. The assumption of a tamper proof GPS module is also debatable since a software tool for simulating a GPS receiver is available (GPSGate, 2008).

On the other hand, the camera deployment is towards a ubiquitous camera (UbiCam) environment, especially in an urban area. For example, the area of the Taichung County, Taiwan, is 163.4 square kilometers and the population is about 960 thousands. The number of cameras deployed simply on the streets is more than 3000. On average, at least one camera is deployed at all intersections. If the real-time image of a camera is used for location authentication of a mobile user, it is cost-efficient and convenient for users. Therefore, a novel vision-based location authentication approach, called VLocAuth, is proposed to prevent the location information being forged not only by an attacker but also by a legal user. That is, a user who has authority to access LBSs, but he forges his location for accessing desired information illegally via LBS. Therefore, when a mobile user wants to access LBS, he must pass the location authentication process of VLocAuth.

The location is critical and is easily forged information. However, only a few studies address the solutions to location authentication. The proposed approaches can be classified in ways similar to the classification of Han and Kim (2007):

- **Time-bound based authentication**: In this way, the transmission time is used to measure the distance between the user and authentication server. The authentication is successful when the measured distance is within a legal range. For example, Sastry *et al*. (2003) proposed echo protocol for secure location verification. It can be applied in indoor or outdoor environments. However, the transmission time is influenced by network traffic. It causes the error of measured distance which it is unable to control or predict. Such unreliable information used for authentication is problematic. Besides, it is unable to know the direction of the user simply based on the undirected transmission time. It limits the feasibility of the approach.
- **Authentication via constrained channel**: In this way, the authentication is restricted to go through some constrained channels, such as Transport Layer Security (TLS) (Dierks and Allen, 1999). The channels also include those with a specific communication range, such as Bluetooth or Wi-Fi. For example, Kindberg *et al*. (2002) proposed a context authentication approach. Location is an important context data. A secure channel proxy was designed for the location authentication. When a user wants to authenticate his location, he must communicate with a server via a nearby proxy with Bluetooth or infrared channel. The location can be authenticated successfully since the communication range of a proxy is limited. However, the deployment of the channel proxy is not a cost-efficient way. The approach is difficult to be widely applied. It is also inconvenient since a user must stay in the communication range of the constrained channel.
- **GPS-based authentication**: In this way, the authentication is based on the coordinates acquired from a GPS receiver. Although the coordinate is easily forged, an additional method is designed to achieve the unforgeability of GPS coordinate. For example, Denning *et al*. (1996) proposed a location signature for location authentication. A location signature sensor (LSS) was built into the approach. The location acquired from GPS receiver must be signed by LSS. An attacker can collect the transmission data between the receiver and satellites for replay attack. The same problem of the previous approaches exists here. The authentication is mainly based on a trusted third-party. It is not cost-efficient to deploy the third-party devices. The

feasibility of the proposed approach also suffered from the short communication range of LSS.

In addition, the approach proposed by Han and Kim (2007) is using a model similar to the trusted authority of PKI (public key infrastructure). It is based on a specific location service architecture defined by the Geopriv Working Group. The defined location sensing method is that the location information of a mobile client can be generated by both the client and the trusted operator. However, such architecture and the location sensing method are not available currently. They do, however show the practicability of this approach.

There are some drawbacks for the previous approaches. Therefore, the vision-based approach (VLocAuth) proposed in this paper is novel to location authentication and meets the UbiCam environmental requirements for the near future.

VISION-BASED LOCATION AUTHENTICATION APPROACH

In the previous methods, the main drawback is that the deployment cost of the trusted devices is too high. It is not cost-efficient and is inconvenient for users since they must find and be close to the trusted device before location authentication. Besides, these methods focus on the protection of location information but do not protect the privacy of mobile clients. The location information of any client should not be able to be obtained by any attackers, authentication or LBS servers. The scenario of VLocAuth is presented in Fig. 1. When a mobile client requests services from LBS server, his location must be authenticated by an authentication server (AS). A camera server (CS) handles the authentication by matching the coordinates of the mobile client and any moving objects in the camera image.

VLocAuth consists of two phases: initialization and location authentication phases. In the initialization phase, a mobile client requests a *UserID*, hash function *h*, random seed, and MAC (Message Authentication Code) function *C* from the LBS Server. The above information is transmitted over a secure channel, such as Intranet or VPN (virtual private network), to ensure data security in the location authentication phase. The random seed is the initial value of the one-way hash function, such as MD5 (Message-Digest Algorithm 5). When the mobile client wants to obtain the LBSs in the location authentication phase, the mobile client acquires temporal identification, denoted TID_C, from the LBS server using *UserID*, hash function *h*, and random seed. Then the mobile client requests location authentication to AS using TID_C. The mobile client submits GPS coordinates to AS. AS forwards coordinates to CS. Then, CS transforms GPS coordinates to image coordinate, and tracks the moving objects in a circle candidate area where its center is the image coordinate. The candidate area is changing due to the movement of the mobile client. Therefore, if the GPS coordinates of the mobile client are not forged, the corresponding moving object should appear in the candidate area. Those moving objects are filtered according to the candidate area until only one object is in the candidate area. The location authentication is successful when the successive moving behavior of the object satisfies specific criteria. Two phases are presented in detailed as follows.

Initialization Phase: The process of the initialization phase is depicted in Fig. 2. There are two sub-phases: registration and operation sub-phase. In the registration sub-phase, a mobile client requests

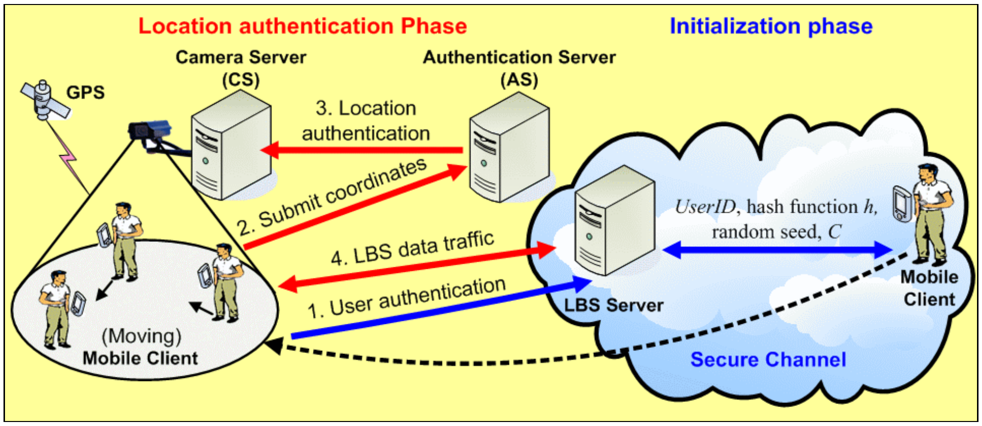


Fig. 1. The scenario of the proposed approach

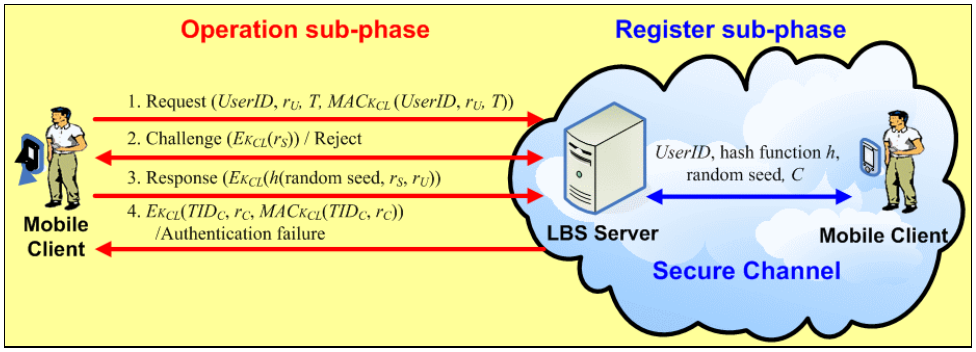

Operation sub-phase
Register sub-phase
1. Request ($UserID$, r_U, T, $MACK_{CL}$($UserID$, r_U, T))
2. Challenge ($EK_{CL}(r_S)$) / Reject
3. Response (EK_{CL}(h(random seed, r_S, r_U))
4. EK_{CL}(TID_C, r_C, $MACK_{CL}$(TID_C, r_C))
/Authentication failure
Mobile Client
LBS Server
$UserID$, hash function h, random seed, C
Mobile Client
Secure Channel

Fig. 2. The process of the initialization phase

a *UserID*, hash function *h*, random seed, and MAC function *C* from LBS Server. The operation sub-phase consists of four steps described below.

- **Mobile Client→LBS Server**: The mobile client generates a random value r_U and a time stamp *T*. Then, a key K_{CL} is generated by using hash function *h*, random seed, and r_U. The key K_{CL} and C is used to generate a MAC code $MAC_{K_{CL}}(UserID, r_U, T)$. Then, the mobile client sends the request to the LBS Server.
$$K_{CL} = h(\text{random seed}, r_U) \tag{1}$$
$$\text{Request} = (UserID, r_U, T, MAC_{K_{CL}}(UserID, r_U, T))$$
- **LBS Server→Mobile Client**: The LBS Server generates the key K_{CL} from r_U contained in the received request. The MAC code, $MAC_{K_{CL}}(UserID, r_U, T)$ contained in the request is called the received MAC. The LBS server generates the expected MAC code from *UserID*, r_U, and *T* by using the MAC function *C* and K_{CL}. If the received MAC is identical to the expected MAC, it means the mobile client is registered to access the LBS server. If the client is registered, the LBS server generates a random value r_S and encrypted it by using the DES algorithm and K_{CL}. Then, the LBS Server sends a challenge, $E_{K_{CL}}(r_S)$, to the mobile client. Otherwise, the LBS server rejects the request.
$$\text{Challenge} = E_{K_{CL}}(r_S) \tag{2}$$
- **Mobile Client→LBS Server**: When the mobile client receives the challenge, he uses the key K_{CL} for the DES decryption. The mobile client generates a response by using hash function *h*, random seed, r_U, and r_S. Then, the response is encrypted by using the DES algorithm and K_{CL}. The mobile client sends the response, $E_{K_{CL}}(h(\text{random seed}, r_U, r_S))$, to LBS server.
$$\text{Response} = E_{K_{CL}}(h(\text{random seed}, r_U, r_S)) \tag{3}$$
- **LBS Server**: When the LBS server receives the response, it uses the key K_{CL} for the DES decryption. The LBS server generates an expected response by using hash function *h*, random seed, r_U, and r_S. If the received response is identical to the expected one, it means that the mobile client is authorized. Otherwise, the authentication process is terminated. When the mobile client is authorized, the LBS server generates a temporal identification TID_C and a random value r_C for the location authentication phase. After the location authentication phase is performed, the LBS server sends the authentication result, either $E_{K_{CL}}(TID_C, r_C, MAC_{K_{CL}}(TID_C, r_C))$ or authentication failure, to the mobile client.

Location Authentication Phase: The mobile client is proceeding to this phase when the client obtains the TID_C from the LBS server. The process is presented in Fig. 3 and described as follows.

- **LBS Server→AS**: Firstly, the LBS server generates the key K_{CA} by using hash function *h*, random seed, and r_C. A MAC code, $MAC_{K_{CA}}(TID_C, r_C)$, is also generated. The LBS server sends TID_C, r_C, K_{CA}, and $MAC_{K_{CA}}(TID_C, r_C)$ to AS. We assume that there is a secure channel, e.g., IPsecESP mode (Kent and Atkinson, 1998), between the LBS server and AS for secure communication.
$$K_{CA} = h(\text{random seed}, r_C) \tag{4}$$
- **Mobile Client→AS**: The mobile client generates the key K_{CA} by using hash function *h*, random seed, and r_C. Then, a MAC code, $MAC_{K_{CA}}(TID_C, r_C)$, is also generated. The mobile client sends the TID_C, r_C, and $MAC_{K_{CA}}(TID_C, r_C)$ that is encrypted by using the DES algorithm and K_{CA} to AS.

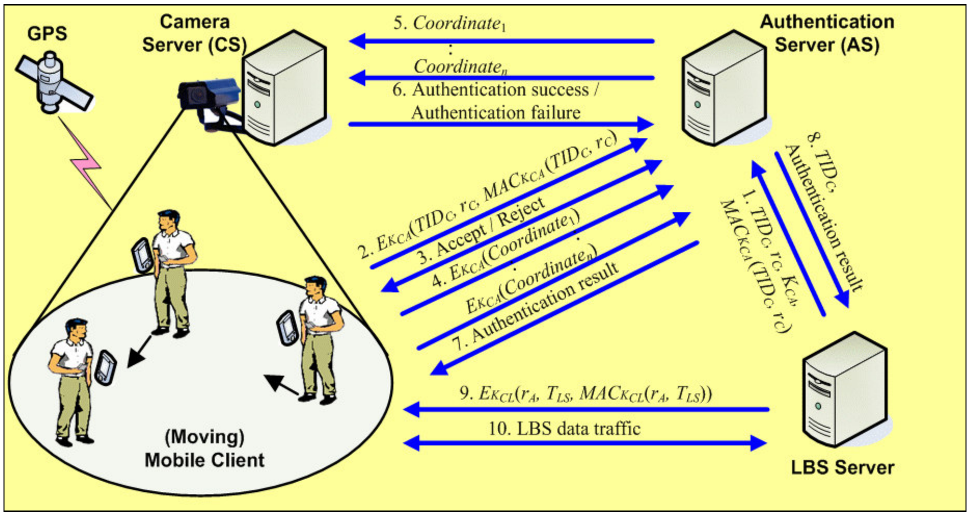

GPS
Camera Server (CS)
Authentication Server (AS)
5. Coordinate1
:
Coordinaten
6. Authentication success / Authentication failure
2. EKCA(TIDC, rC, MACKCA(TIDC, rC)
3. Accept / Reject
4. EKCA(Coordinate1)
:
EKCA(Coordinaten)
7. Authentication result
1. TIDC, rC, KCA, MACKCA(TIDC, rC)
8. TIDC, Authentication result
9. EKCL(rA, TLS, MACKCL(rA, TLS))
10. LBS data traffic
(Moving) Mobile Client
LBS Server

Fig. 3. The process of location authentication phase

- **AS→Mobile Client**: AS uses the key K_{CA} for the DES decryption to decrypt the ciphertext received from mobile client. AS verifies the authorization of the mobile client by using the TID_C and two MAC codes received from the LBS server and the mobile client. If the mobile client is authorized, AS sends an accept message to the client. Otherwise, the authentication process is terminated and a reject message is sent to the client.
- **Mobile Client→AS**: The mobile client acquires the coordinates from a GPS receiver. A coordinate is encrypted by using the DES algorithm and K_{CA}, i.e., $E_{K_{CA}}(Coordinate_i)$, and sent to AS.
- **AS→CS**: AS uses the key K_{CA} for DES decryption. The decrypted GPS coordinate is sent to a CS that the coordinate is in the field-of-view (FOV) of the managed cameras for the location authentication. A secure tunnel is also assumed between the AS and CS.
- **CS→AS**: CS starts the matching of successive GPS coordinates received from the mobile client and moving objects in the real-time camera image. This is the key step to determine the success or failure of location authentication which is presented in detail later. When the authentication result is generated, CS sends the result to AS.
- **AS→Mobile Client**: When AS receives the authentication result from CS, it forwards the result to the mobile client.
- **AS→LBS Server**: AS also sends the location authentication result with the TID_C to LBS server.
- **LBS Server→Mobile Client**: If the LBS server receives a success message from AS, it generates a random value r_A, valid period T_{LS} for the LBSs, and the MAC code $MAC_{K_{CL}}(r_A, T_{LS})$. The generated data is encrypted by using the DES algorithm and K_{CL} and sent to the mobile client.

Table 1. Notations of the location authentication approach

Notation	Description
TD	Tolerate distance, i.e., the radius of the candidate area
TE	The valid time period for location authentication
CSet	A candidate set including all the possible mobile clients
SV	The accumulated value of successive moving vector differences

- **Mobile Client**: The received $E_{K_{CL}}(r_A, T_{LS}, MAC_{K_{CL}}(r_A, T_{LS}))$ from LBS server is decrypted by using the key K_{CL} for DES decryption. Then, the mobile client generates the key K_{CS} by using hash function h, random seed, and r_A. The transmission of LBS data traffic between the LBS server and the mobile client is encrypted simply by using the key K_{CS}. If the valid period T_{LS} has expired, the mobile client must authenticate again by obtaining a new LBS authority.

$$K_{CS} = h(\text{random seed}, r_A) \tag{5}$$

During the process illustrated above, the matching of the GPS coordinates and the moving objects in the CS is critical and described in detail below since it determines the authentication result. The notations used throughout the description of the location authentication rules are listed in Table 1.

Firstly, the CS transforms the GPS coordinates to image coordinates C_i. The transformation can be based on Tsai, (1987). Since the GPS coordinates are inaccurate, a tolerate distance, denoted as TD, is thus defined. A circular area, called the candidate area, is

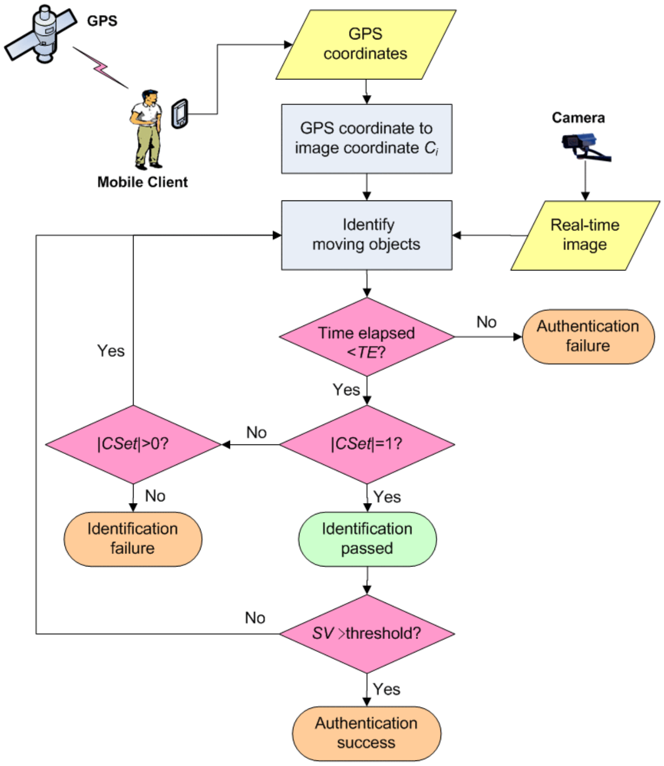


Fig. 4. The process of the key step in the location authentication phase

formed where its center is located at C_i and the radius is *TD*. The whole authentication process must be finished within a valid time period, denoted as *TE*. When the authentication process is started, those mobile objects that are within the candidate area are collected in a candidate set, denoted as *CSet*. The candidate area changes as the received GPS coordinates change. An object in the candidate set is removed when it does not stay in the candidate area. When there is only one object finally in the candidate set before the time expires by *TE*, it means identification success. That is, the corresponding moving object of the mobile client is identified in the camera image. The sum of successive vector differences of the object, denoted as *SV*, is also accumulated. If there is only a few or even one object in the candidate area, the object is easily identified successfully. The *SV* must be larger than a pre-defined threshold to ensure the identified object has sufficient room to move. For example, if the threshold is 720 degrees, it means that the object must turn around at least twice. On the other hand if all the objects are removed from the candidate set, it means that the identification is a failure, i.e., authentication failure. The above process is depicted in Fig. 4. In advance, several typical scenarios of the key step in the location authentication phase are presented in Fig. 5. They are described as follows.

(1) **Moving out**: When an object moves out the candidate area, it is removed from the candidate set. If this object moves back into the candidate area later, it will not be added into the candidate set.

(2) **New object is joined**: When a new object is moved into the candidate area, it is ignored.

(3-1) **Identification failure**: When all the objects in the candidate set are removed, i.e., the candidate set is empty, the identification is a failure, i.e., authentication failure.

(3-2) **Authentication failure** (*current time* − *start time*>*TE*): When the elapsed time is longer than *TE*, the authentication is a failure.

(3-3) **Identification passed**: When there is only one object in the candidate set, the identification is passed.

(4) **Authentication success** (***SV* > *threshold*** **and** ***current time*−*start time*≤*TE***): When the *SV* of the identified object is larger than the threshold and the elapsed time is equal or shorter than *TE*, the authentication is a success. Otherwise, the *SV* is accumulated for checking periodically until one of the scenarios, 3-1, 3-2, and 4, occurs.

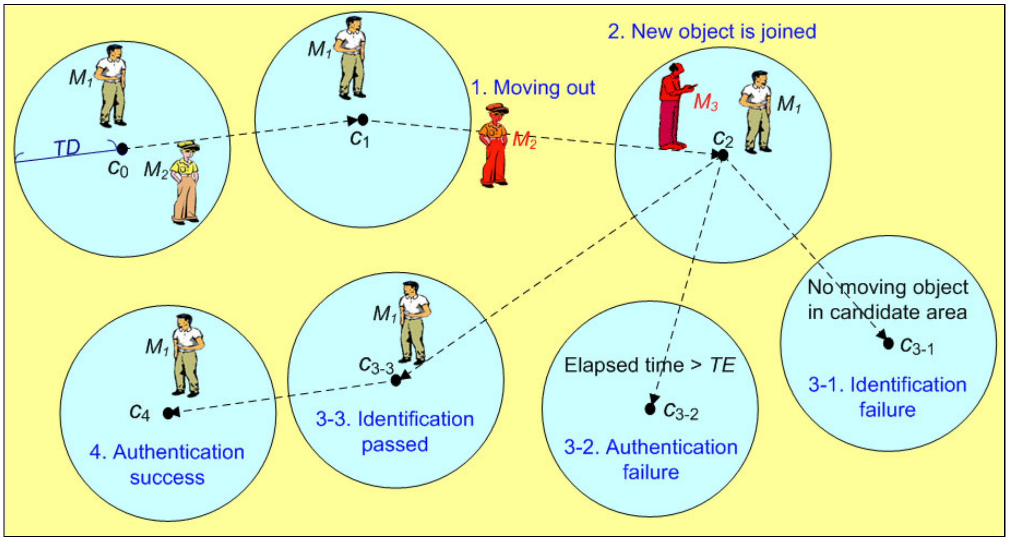

2. New object is joined
1. Moving out
M_1
M_2
M_3
TD
C_0
C_1
C_2
No moving object in candidate area
$C_{3\text{-}1}$
3-1. Identification failure
Elapsed time > TE
$C_{3\text{-}2}$
3-2. Authentication failure
$C_{3\text{-}3}$
3-3. Identification passed
C_4
4. Authentication success

Fig. 5. The typical scenarios of the key step in the location authentication phase

```
Algorithm Location Authentication
Input: none
Output: authenticate result
CSet: candidate set
TD: tolerate distance
ST: starting time
TE: valid time period
SV: the accumulated difference of successive vector
Begin
    Transform received GPS coordinate to the corresponding image coordinate;
    Establish the candidate area according to the image coordinate and TD;
    CSet:= moving objects in the candidate area;
    ST:= current time;
    While (current time −ST)≦TE and |CSet|>0 do
        Transform received GPS coordinate to the corresponding image coordinate;
        Establish the candidate area according to the image coordinate and TD;
        CurSet:= all the moving objects identified in the current candidate area;
        Remove objects in CSet when they are not included in CurSet;
        Compute SV for every objects in CSet;
        If |CSet|==1
            if SV of the only object in CSet >threshold
                return success;
            End If
        End If
    End While
    return failure;
End
```

Fig. 6. The location authentication algorithm

According to the above scenarios, objects are excluded based on the candidate area updated periodically until only one object exists in the candidate set. The object is then checked whether its *SV* is larger than the pre-defined threshold, e.g., 720 degrees. If it is true, the authentication is a success. However, if all the objects are excluded or the elapsed time is longer than the *TE*, the authentication is a failure. A large *SV* and a short *TE* increase the difficulty to authenticate successfully. It also means that an attacker will find it difficult to forge the movement of a mobile client. The design of the *SV* and the *TE* parameters enable the flexibility of VLocAuth which will satisfy the desirable security requirements of location authentication. The algorithm for the key step in the location authentication phase is presented in Fig. 6.

SECURITY ANALYSIS

In order to evaluate the security of VLocAuth, we analyzed for the following attacks, issues, and compared with some previous approaches:

Attack 1: Suppose the attacker can intercept the request message $(UserID, r_U, T, MAC_{K_{CL}}(UserID, r_U, T))$, the attacker cannot pass the authentication of LBS server through replay attack since the message includes a time stamp *T*. In advance, a challenge is answered after the MAC code is verified by the server to ensure the message is not altered during transmission. Only the registered mobile client owns the hash function *h* and random seed to generate the response correctly.

Attack 2: Only the registered mobile client owns the hash function *h* and random seed. Therefore, only the registered mobile client can generate the keys, K_{CL}, K_{CA}, and K_{CS}. The attacker is unable to guess or estimate the correct key values. VLocAuth can resist the man-in-the-middle attack.

Unforgeability issue: In VLocAuth, the ubiquitous camera is responsible for the authentication by matching the GPS coordinates and moving objects in the real-time camera image. Either a legal user or an attacker is unable to forge the GPS coordinates and the movement of the moving object simultaneously to authenticate location successfully. Therefore, VLocAuth meets the requirements of location unforgeability.

Privacy issue: Before the location information is transmitted to the AS, it is symmetrically encrypted by using r_C given by the LBS server and a shared secure code obtained in the registration phase. An attacker is unable to obtain the location information. Besides, the GPS coordinates are associated with a temporal identifier TID_C given by the LBS server. The AS does not know the real identifier requested for location authentication. In addition, the location authentication is handled by AS and CS. The LBS server does not know the location of the mobile client. Accordingly, VLocAuth protects the privacy of mobile clients.

Confidentially issue: Only the LBS server and registered mobile clients obtain the shared hash function h, MAC function C, and random seed. Therefore, only K_{CL}, K_{CA} and K_{CS} can be generated by them. The same session key is used for decrypting ciphertext. Therefore, VLocAuth can resist the ciphertext-only attack.

Integrity issue: The transmission message is associated with the MAC code in VLocAuth. The LBS server and the registered mobile client only share the function C, of the MAC code. An attacker is unable to intercept and tamper with the message. Therefore, VLocAuth protects the message integrity.

Simplicity issue: Only symmetric encryption algorithm and exclusive-OR operation is used in VLocAuth. Therefore, VLocAuth is simple and suitable for mobile devices with limited computing resources.

Practicability issue: VLocAuth is mainly based on the coming of a UbiCam environment. It is practical and cost-effective. It can also promote the applicability of UbiCam environment.

VLocAuth is compared to the previous approaches, including time-bound based (Sastry *et al*., 2003), constrained channel (Kindberg *et al*., 2002), and GPS-based approaches (Denning *et al*., 1996). The comparison results listed in Table 2 are extended from that presented in the trusted authority (Han and Kim, 2007). The extended part is marked in bold letters. The mark "✓" denotes that the approach satisfies the requirement in the row and "×" denotes that it does not satisfy. The original privacy requirement in Han and Kim (2007) is divided into two ones, privacy against attacker and privacy against service provider. They are used to check the privacy requirement more clearly. The deployment cost is also compared in the table.

For the time-bound based authentication approach proposed by Sastry *et al*. (2003), there is no deployment cost since there is no additional hardware or device needed. However, the location and personal information is not encrypted during transmission. That is, the approach cannot protect the privacy against attackers and service provider. In addition, its covered range is limited since the transmission time is easily influenced by the network traffic.

For the GPS-based authentication method proposed by Denning and Macdoran (1996), it mainly based on the location signature sensor (LSS) to authenticate the location information acquired by the GPS receiver. However, it is vulnerable to replay attack since an attacker can collect transmission data between a receiver and satellites. The deployment cost is high since the devices of users must be equipped LSS, and many fixed hosts equipped LSS must be also installed for location authentication.

For the authentication approach based on constrained channel proposed by Kindberg *et al*. (2002), the deployment cost is high since many third-party devices are needed. It is also inconvenient for users since the access range of the third-party devices is limited. Although this approach can protect the privacy against attackers during transmission, it does not protect the privacy against the service provider. That is, the service provider can know the real-time locations of users. The provider can even analyze users' preferences

Table 2. The comparison of various location authentication approaches

	Approach				
Requirement	[1]Time-bound based	[2]GPS-based	[3]Constrained channel	[4]Trusted Authority	**VLocAuth**
Unforgeability	✓	✓	✓	✓	✓
Privacy against attacker	×	✓	✓	✓	✓
Replay attack	✓	×	✓	✓	✓
Universality	×	✓	×	✓	✓
Covered range	Near	Device specific	3 000 km	No limit	Device specific
Privacy against service provider	×	×	×	×	✓
Deployment cost	**None**	**High**	**High**	**High**	**Middle**

The extended part is marked in bold letters. The mark ✓ denotes that the approach satisfies the requirement in the row and × denotes that it does not satisfy. [1]Sastry *et al*. (2003), [2]Denning and Macdoran (1996), [3]Kindberg *et al*. (2002) and [4]Han and Kim (2007) from the logged location information.

For the approach using trusted authority and proposed by Han and Kim (2007), the identifier of a user is transmitted with its location information while requesting LBS. That is, it cannot protect the privacy against the service provider. Besides, the approach is based on a specific location service architecture defined by the Geopriv Working Group. The defined location sensing method is that the location information of a mobile client can be generated by both the client and the trusted operator. However, such location service architecture and the location sensing method are not available currently. The deployment cost for practicing the proposed approach is high.

With our approach, VLocAuth, the deployment cost is middle since the method utilizes the UbiCam environment currently available and even more getting popular. The authentication is based on the match between the GPS coordinates and moving objects in the real-time camera image. It is impossible to forge GPS coordinates and movement in the camera image. That is, VLocAuth meets the unforgeability requirement. The transmission of location information is encrypted in advance by r_C given by the LBS server and a shared secure code obtained in the registration phase. This method can protect the user privacy against attackers. A temporal identification, TID_C, is also given to the mobile client by the LBS server for location authentication with AS. AS does not know the real identification of the mobile client and the LBS server does not know the location information of the client. Therefore, the method can also protect user privacy against the service provider. The method can also prevent replay attack because of the design of the challenge/response mechanism in the initialization phase.

PERFORMANCE EVALUATION

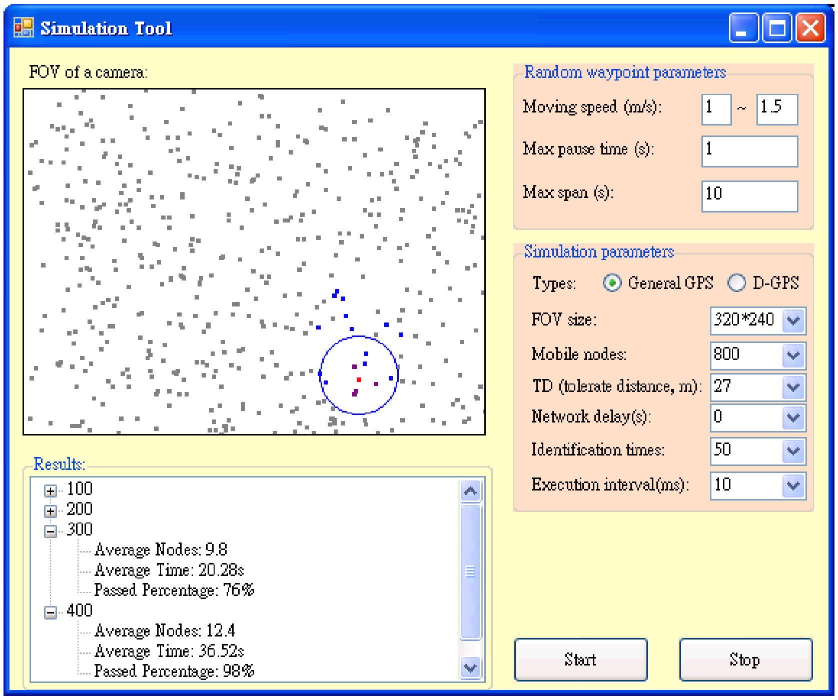
In VLocAuth, the matching between GPS coordinates and moving objects in the real-time camera image is the key step in determining the success or failure of location authentication. The matching process is mainly influenced by the following factors:

- **Type of GPS receiver**: there are mainly two types of GPS receiver according to the positioning error. A positioning error of a general GPS (G-GPS) and differential GPS (D-GPS) receiver is five to 30 meters and zero to five meters, respectively. The smaller the error, the easier it is to match and identify the mobile client target.
- **Number of moving objects in the FOV of a camera**: The smaller the number is, the easier it is to identify the mobile client target.
- **Network delay**: It is the transmission time of a GPS coordinate from a mobile device to CS. The smaller the delay is, the better it is to match GPS coordinate with the moving objects in the real-time camera image.

The above factors are difficult to verify by mathematical proof. Therefore, a simulation tool and a matching tool were implemented separately to evaluate the performance of the key matching step from different ways. They are presented in the following subsections.

The simulation study: The simulation tool is designed to imitate a system developed previously, called GODTA (GPS-based Object Detection and Tracking Approach) (Liao and Chu, 2009). Its screen shot is shown in Fig. 7. The left-upper part is the FOV of a camera. The parameter setting is listed on the right-hand side. The people walking in a square are assumed to be the typical environment for location authentication. It is simulated by using a random waypoint mobility model. The parameters of the model are listed on the upper-right part. They include speed of movement, maximum pause time, and maximum time span. A person, i.e., a node, moves in the range of the speed of movement for a period of time that is less than the maximum time span. Then, it pauses for another period of time that is less than the maximum pause time. A new direction is chosen randomly and repeats the above steps again. The lower-right part of the screen shot shows the simulation parameters. They include the selection of GPS types, the FOV size, and the number of mobile nodes, *TD*, network delays, and identification times (the simulation times). The execution interval is used to control the execution speed of the simulations. The identification of the mobile client is the primary process of location authentication. When the mobile client is identified, its location is authenticated after its *SV* satisfies the threshold within the period limited by *TE*. The checking of *SV* and *TE* is trivial. Therefore, only the identification of the mobile client is considered in the simulation study.

When the "Start" button is pressed, mobile nodes are moving according to a random waypoint mobility model. For every location authentication, a node is chosen randomly as the mobile client. It is marked by the color red as shown in Fig. 7. The candidate area based on the received GPS coordinates



FOV of a camera:

Random waypoint parameters

Moving speed (m/s): 1 ~ 1.5

Max pause time (s): 1

Max span (s): 10

Simulation parameters

Types: ● General GPS ○ D-GPS

FOV size: 320*240

Mobile nodes: 800

TD (tolerate distance, m): 27

Network delay(s): 0

Identification times: 50

Execution interval(ms): 10

Results:

```
⊞ 100
⊞ 200
⊟ 300
    Average Nodes: 9.8
    Average Time: 20.28s
    Passed Percentage: 76%
⊟ 400
    Average Nodes: 12.4
    Average Time: 36.52s
    Passed Percentage: 98%
```

Fig. 7. The screen shot of the location authentication simulation tool

Table 3. The parameter settings of the simulations

		Custom setting		
Parameters	Default setting	PP vs. node density by using G-GPS	PP vs. delay	PP and time by using D-GPS
Simulation times	500			
FOV size	320×240 m^2			
Mobile nodes	20~800			
Mobility model	Random waypoint			
Moving speed	1~1.5 m/s			
Pause time	0~1 sec.			
Time span	0~10 sec.			
Network delay		0 sec.	0, 1, 2, 3 sec.	0 sec.
TD (G-GPS)		15, 18, 21, 24, 27 m	24, 27 m	15, 18, 21, 24, 27 m
TD (D-GPS)		-	-	2, 3, 4 m

PP: Pass Percentage

and the *TD* is marked by a blue circle. Initially, all the nodes in the candidate area are added into the candidate set, *CSet*. However, when a node moves out of the candidate area, it is removed from the *CSet* and marked by the color blue. Those nodes marked by the color purple are still included in the *CSet*. The pass or failure of the identification process is determined according to the rules described in the previous section. When all the identification is finished, the result is listed on the lower-left part. The results consist of three items: average nodes, average time, and pass percentage. The average node is the average number of nodes in the initial *CSet*. The average time is the average time to finish the identification process. The passed percentage is the percentage of the mobile clients identified correctly.

Three simulations are designed here. One shows the pass percentage versus various node densities by using a general GPS receiver. Another shows the influence of network delay on the pass percentage. The other shows the pass percentage and average time of identification by using a popular D-GPS receiver.

The parameter settings of three simulations are listed in Table 3. The basic setting is common to the three simulations. The total number of simulations is five hundred. The FOV size is 320×240 m^2. The number of mobile nodes varies from 20 to 800 simulating sparse to dense conditions. The setting of the speed of movement is close to the normal walking speed. Those custom settings are presented below.

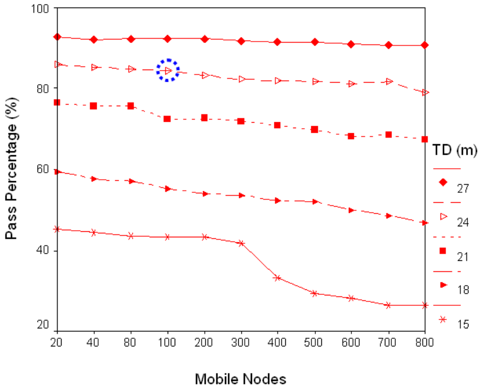


Fig. 8. Pass percentage vs. mobile nodes for G-GPS receiver

Pass percentage vs. node density by using G-GPS receiver: For a G-GPS receiver, this simulation shows the influence of node density and *TD* to the pass percentage. For the custom setting of parameters in this simulation, the network delay is zero seconds for executing the simulation under an ideal situation. The setting of *TD* is from 15 to 27 meters since the positioning error of a G-GPS is about five to 30 meters. The simulation results of pass percentage using various *TD*s are depicted in Fig. 8. When *TD* is only 15, the corresponding pass percentage is decreased quickly to less than 30 percent by increasing the node density. However, when *TD* is 24 or 27 meters, the pass percentage is larger than 80 percent no matter what the node density is. For a normal situation with 100 mobile nodes and the *TD* set to 24 meters, shown by the blue-dashed circle in Fig. 8, the pass percentage is 83.4 percent and the average identification time is 14.26 seconds. The result of such setting is good for practical usage by using G-GPS.

Pass percentage vs. network delay by using G-GPS receiver: There exists the delay while transmitting the GPS coordinates from a mobile device to the CS in the practical environment. Therefore, the custom setting of network delay and *TD* is from zero to three seconds and 24 or 27 meters, respectively, as listed in Table 3. The simulation results are depicted in Fig. 9. According to the results, the delay causes a small decrease in the pass percentage compared with that of zero delay. The decrease is less than five percent. Therefore, a larger *TD* can keep the pass percentage high and reduce the influence of network delay.

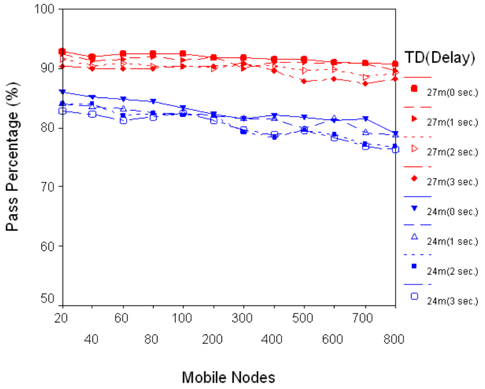


Fig. 9. Pass percentage vs. network delay for mobile nodes with G-GPS receiver

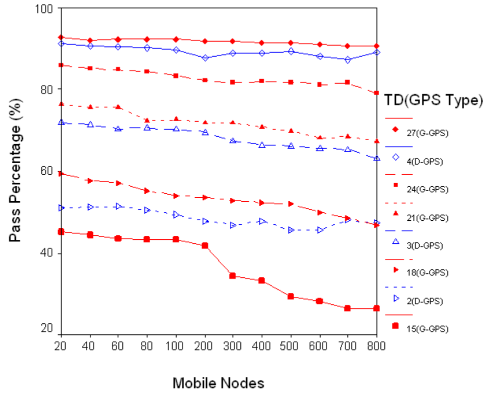


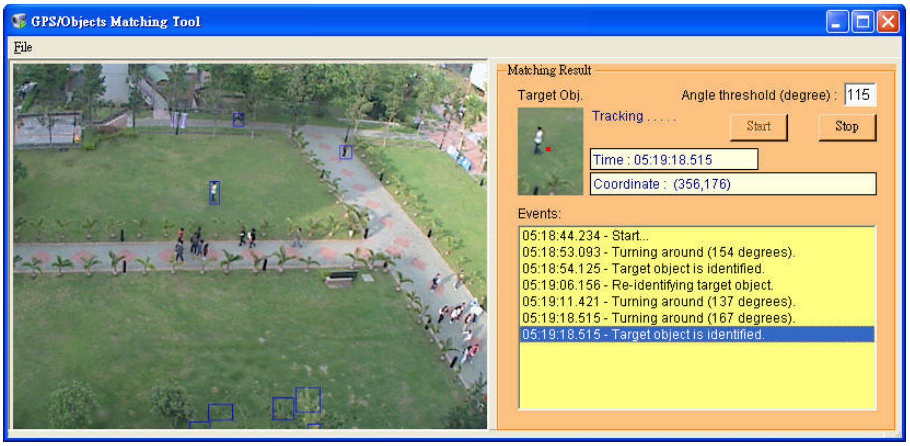
Fig. 10. Pass percentage vs. mobile nodes with G-GPS and D-GPS receivers

Pass percentage and identification time by using D-GPS receiver: In the previous simulations, a G-GPS receiver is used show the pass percentage under various node densities and *TD*s, The pass percentage and identification time of a D-GPS receiver are measured in advance in this simulation. The parameter settings are listed in Table 3. The parameter, *TD* (D-GPS), is set to two, three, or four meters since the positioning error of a D-GPS receiver is zero to five meters. The simulation results are depicted in Fig. 10. When *TD* equals three meters, the pass percentage is about 68 percent. When *TD* equals four meters, the pass percentage is increased quickly to 90 percent. It is similar to that of a G-GPS receiver with 27 meters *TD*.

The time needed to identify the target mobile node for two types of GPS receivers is quite different. The results of the average identification times are listed in Table 4. According to the results, the identification time of a G-GPS receiver is increased by the increase of mobile nodes and *TD*. Conversely, for D-GPS receiver, the identification time is shortened to less than one second by a small *TD*. To

Table 4. Identification time vs. mobile nodes with G-GPS and D-GPS receivers

GPS Type	*TD* (G-GPS)					*TD* (D-GPS)		
Nodes	15	18	21	24	27	2	3	4
20	0.4	1.5	1.6	3.4	6.2	0.01	0.008	0.02
40	0.7	2.3	3.6	6.5	12.3	0.01	0.05	0.07
60	1.3	3.3	5.6	7.8	16.0	0.003	0.02	0.07
80	1.6	3.4	7.3	12.5	19.0	0.01	0.09	0.09
100	2.2	3.8	9.3	14.3	21.6	0.012	0.04	0.21
200	4.2	9.2	13.8	24.3	33.3	0.016	0.1	0.25
300	4.7	10.4	17.8	27.2	38.9	0.055	0.18	0.45
400	6.1	14.5	20.8	29.9	44.7	0.062	0.18	0.57
500	6.9	15.2	24.3	35.4	50.4	0.061	0.2	0.63
600	8.1	17.3	24.0	36.6	54.6	0.07	0.23	0.88
700	8.4	16.4	25.6	39.7	57.8	0.06	0.35	0.8
800	8.1	17.7	26.4	41.3	58.9	0.038	0.37	1



Angle threshold (degree) : 115

Tracking . . . . .

Time : 05:19:18.515

Coordinate : (356,176)

Events:

```
05:18:44.234 - Start...
05:18:53.093 - Turning around (154 degrees).
05:18:54.125 - Target object is identified.
05:19:06.156 - Re-identifying target object.
05:19:11.421 - Turning around (137 degrees).
05:19:18.515 - Turning around (167 degrees).
05:19:18.515 - Target object is identified.
```

Fig. 11. The screen shot of the matching tool

sum up, when the number of nodes are less than 100, the time taken by a G-GPS receiver is less than 21.6 seconds. It is acceptable for the location authentication. However, when the number of nodes is high, a D-GPS receiver is suitable for fulfilling the need to shorten the authentication time.

The empirical study: The key matching step of VLocAuth is mainly based on the image understanding technique which may be time-consuming. Therefore, a matching tool was implemented to match the GPS coordinates and moving objects on the real-time camera image. It is used to evaluate the performance of the key matching step in the practical environment. The screen shot of the tool is shown in Fig. 11. The real-time camera image is captured continuously and displayed on the left-hand side. The subtraction of two consecutive images is used to detect the foreground moving objects. Initially, those objects are collected in a candidate set (*CSet*) when its size is within a predefined range and marked by a blue rectangle as shown in the figure. A client tool was also implemented to transmit the GPS coordinates of the target object every second to the matching tool. When the target object is moving and turning around, it causes a larger angle of two successive moving vectors computed from the GPS coordinates. The angle change of the corresponding image coordinates should be similar, too. When the vector angle of GPS coordinates is larger than a threshold set on the upper-right corner of the figure, the matching tool removes all the objects without such angle change from *CSet*. The above process is repeated until only one object is left in the *CSet*. The image of the last object is displayed on the upper-right corner. If the object is the designated target, it means the target object is identified successfully. Then, the target object is re-identified again and the time is listed in the event list on the lower-right corner of the figure. For the example shown in the figure, the re-identification is started on 5:19:06. The target

Table 5. The identification time of the matching tool in practical environment

	Identification time of 86 tests					
No. moving objects	1~15	16~30	31~45	46~60	61~75	76~83
1~10 objects (83 tests)	9	17	9	13	8	10
	10	7	9	11	11	14
	15	10	17	12	10	12
	12	11	6	13	11	10
	11	6	15	10	15	12
	17	8	9	9	13	13
	9	21	12	12	10	9
	8	6	15	13	9	10
	13	13	17	13	14	
	13	10	10	11	13	
	7	14	8	12	7	
	11	8	12	15	11	
	12	15	15	12	12	
	16	16	11	9	13	
	8	6	13	10	12	
21~30 objects (1 test)	5					
41~50 objects (2 tests)	9					
	13					
Minimum time (sec) = 5						
Maximum time(sec)= 17						
Average time(sec)= 11.37						

object is identified after two turning around events on 5:19:11 and 5:19:18. The corresponding vector angles are 137 and 167 degrees that are larger than the threshold, 115 degrees. The matching tool spent 12 seconds (5:19:06~5:19:18) to identify the target object.

Two performance values are measured using the matching tool. One is the computing time of image understanding technique. The other is the identification time. The main image understanding technique used in the key matching step is moving object detection. For the measurement of the computing time, two consecutive images are processed by a series of steps for detecting moving objects, including convert to gray scale, subtraction, binarization, dilatation, extraction the BLOB (binary large object) data, i.e., the moving objects. The above steps are repeated 100 times on an AMD Athlon 64X2 dual core 5600+ processor 2.9 GHz desktop computer. The average computing time is 138 and 37 milliseconds for one 640×480 and 320×240 image, respectively. It shows that the image understanding is quite efficient. Especially, when many clients request for authenticating their locations and they are within the FOV of the same camera, the moving objects are only needed to be detected once for every image. Then, the information of the moving objects can be used to match with the GPS coordinates of individual clients. Such matching time is not included in the computing time since it is very fast with respect to the moving object detection.

For the measurement of the identification time, a target object is re-identified for 86 times by using the matching tool. The results of the identification time are listed in Table 5. They are classified based on the number of moving objects. Most of identifications are performed on one to 10 objects. The time is not changed obviously when the number of moving objects is increased to 50. The identification time is ranging from five to 17 seconds and the average time is 11.37 seconds. The results are similar to the simulation results of the G-GPS listed in Table 4, i.e., 12.3 seconds for 40 nodes.

CONCLUSION AND FUTURE WORK

The previous approaches, including time-bound based, constrained channel, and GPS-based, have some drawbacks, such as deployment cost is high, cannot protect privacy against attacker or service provider, or cannot prevent replay attack. Therefore, a vision-based location authentication approach, called VLocAuth, is proposed in this paper. The real-time camera image is incorporated into the location authentication mechanism for improving the drawbacks of previous approaches. The coming of the UbiCam environment is helpful in increasing the feasibility of VLocAuth. VLocAuth not only achieves location authentication but also protects the privacy of users against attacker and service provider. For the simulation study of the key step in location authentication phase, the simulation results show that the pass percentage of a general GPS (G-GPS) receiver can be up to 90 percent and the identification time is less than one second for a differential GPS (D-GPS) receiver. The results of the matching tool also show that VLocAuth is able to handle many requests simultaneously without influenced seriously by the image understanding technique. And the average identification time, 11.37 seconds, is acceptable in the practical environment. Therefore, VLocAuth is a novel ideal by utilizing the UbiCam environment on location authentication. It also raises some issues to be addressed for further development of location authentication. For example, the location exposure is usually determined by services, temporal, or spatial information. VLocAuth can be refined to control of location exposure. In addition, it will be extended to indoor environments so as to become a total solution for location authentication in the future.

ACKNOWLEDGEMENT

This project was financially supported by the National Science Council under Grant No.: NSC 97-2221-E-324-043.

REFERENCES

Al-Muhtadi, J., A. Ranganathan, R. Campbell and M.D. Mickunas, 2002. A flexible, privacy-preserving authentication framework for ubiquitous computing environments. In: Proceedings of the 22nd International Conference on Distributed Computing Systems Workshops (ICDCSW'02), 2-5 July 2002, IEEE Computer Society, pp: 771-776.

Denning, D.E. and P.F. Macdoran, 1996. Location-based authentication: Grounding cyberspace for better security. Comp. Fraud & Secur., 12-16, Feb. 1996, http://portal.acm.org/citation.cfm?id=275749.

Dierks, T. and C. Allen, 1999. Transport layer security. http://www.ietf.org/html.charters/tls-charter.html.

Durresi, A., V. Paruchuri, M. Durresi and L. Barolli, 2007. Secure spatial authentication using cell phones. In: Proceeding of the 2nd International Conference on Availability, Reliability and Security (ARES 2007), April 2007, Washington, DC, USA, pp: 543-549.

GpsGate, 2008. Run many GPS applications using one GPS! Franson Technology AB, http://franson.com/gpsgate/.

Han, K., S. Lee, K. Kim and S.R. Ine, 2004. On the design of secure DRM in ubiquitous environment. In: Proceeding of the KIISC Youngnam Branch Workshop, Kyungil Univ., Kyungsan, Feb. 2, 2004, http://koasas.kaist.ac.kr/bitstream/10203/15741/1/drm%5B1%5D.pdf

Han, K. and K. Kim, 2007. Enhancing privacy and authentication for location based service using trusted authority. In: Proceedings of the 2nd Joint Workshop on Information Security (JWIS2007), Waseda University, Tokyo Japan, Aug. 6-7, 2007, http://citeseerx.ksu.edu.sa/viewdoc/summary?doi=10.1.1.101.6304.

Kent, S. and R Atkinson, 1998. Security architecture for the internet protocol. Nov. 1998, http://www.ietf.org/rfc/rfc2401.txt.

Kindberg, T., K. Zhang and N. Shankar, 2002. Context authentication using constrained channels. In: Proceedings of the 4th IEEE Workshop on Mobile Computing Systems and Applications (WMCSA), June 14-21, USA., pp: 14 2002, http://portal.acm.org/citation.cfm?id=837553.

Liao, H.C. and Y.H. Chao, 2008. A new data encryption algorithm based on the location of mobile users, Info. Tech. J., 7(1): 63-69.

Liao, H.C., and P.T. Chu, 2009. A novel visual tracking approach incorporating global positioning system in a ubiquitous camera environment, Info. Tech. J., 8(4): 465-475.

Murakami, T. and J.S. Ke, 2006. Ubiquitous network society: emerging e-business opportunities. Global Business Dialogue on Electronic Commerce, http://www.gbd-e.org/ig/uns/UbiquitousSocietyVisionRecommendation_Nov06.pdf

Ren, K., L. Wenjing, K. Kwangio and R. Deng, 2006. A novel privacy preserving authentication and access control scheme for pervasive computing environments. IEEE Trans. on Veh. Tech., 55(4): 1373-1384.

Ren, K. and L. Wenjing, 2007. Privacy-enhanced, attack-resilient access control in pervasive computing environments with optional context authentication capability. Mobile Net. and App., 12(1): 79-92.

Sastry, N., U. Shankar and D. Wagner, 2003. Secure verification of location claims. In: Proceedings of the 2nd ACM Workshop on Wireless Security, September 2003, San Diego, CA, USA., pp: 1-10.

Scott, L. and D.E. Denning, 2003. Using GPS to enhance data security: Geo-Encryption. GPS World, 14:40-49, 2003.

Tsai, R.Y., 1987. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. IEEE J. of Rob. and Automat., 3(4): 323-344.