Augmented Reality Assisted Photo Positioning for Mobile Devices

Ju-Yi Lin, Chih-Wei Yi and Yu-Chee Tseng
Department of Computer Science, National Chiao Tung University
1001 University Road, Hsinchu City 30010, Taiwan
Emails: {juyi.yi,yctseng}@cs.nctu.edu.tw

Abstract—Recent developments in mobile techniques have enabled a great variety of Location Based Services (LBS). A high positioning accuracy is a fundamental requirement for precision LBS applications, e.g., precise LBS marketing in shopping malls or indoor emergency evacuation services with mobile devices. However, most offerable commercial positioning systems, such as GPS/GNSS and RF-based systems, can not provide positioning accuracy within one meter. In this work, a new positioning approach is proposed for mobile devices, called Photo Positioning, it can provide a high accuracy positioning service. The "positioning" here means to find the location where a photo was taken by investigating the geometric relations between the images of points of interest (POI) in the photo and their location in the real world based on the principle of photo imaging. To implement a photo positioning system, three major components are needed, including a POI database, a method to recognize and locate POIs in a photo, and an algorithm to calculate the position where the photo was taken from the POI information. A positioning algorithm based on the geometric similarity of photo imaging is presented in this work, and a prototype system is developed for Android smartphone platforms. Our experimental results show that the average positioning error of the proposed photo positioning approach can be as low as 74.34 cm.

I. INTRODUCTION

Positioning accuracy plays an important role in Location-Based Services (LBS). For example, as people can become lost in unfamiliar environments, such as shopping malls or parking towers, positioning systems can quickly locate a user’s current position and cooperate with navigation systems to provide routes to desired stores or proper parking lots. However, inaccurate shopping mall navigation systems, for example, may guide customers to incorrect stores, aisles or shelves. This significantly reduces the usefulness of such applications. Most current commercial positioning systems can not provide positioning accurate to within one meter. Those that can may incur high start-up or maintenance costs.

In this work, a new positioning approach called Photo Positioning is proposed, in which the "position" is the location where a photo was taken. This technique can provide a highly accurate positioning service for mobile devices. The concept behind it are demonstrated by the following example. Consider a photo of the Taipei 101 skyscraper. We may guess that the photo was taken in Taipei rather than New York. The dimensions of the image of Taipei 101 can be used as a reference to estimate the distance from the photographer to the building. In addition, if the front side of Taipei 101 is captured, we can further guess that this photo was taken somewhere in front of the building. According to this observation, the location of the photographer can be determined based on the geometric and geographic relations between the points of interest (POI) in the photo. To implement such a photo positioning system, three modules are required, including a POI database, a method to recognize and locate POIs in a photo, and a positioning algorithm that can calculate the position from the geometric information.

Positioning technologies can be categorized into outdoor and indoor systems. The GPS [1] is the most popular outdoor positioning system. Its positioning accuracy ranges from 10 m to 30 m. A GPS receiver determines its position based on the pseudoranges to GPS satellites. Some augmentation systems have been proposed to improve the accuracy of the GPS, such as Assisted GPS (A-GPS), Differential GPS (DGPS) [2] and the Wide Area Augmentation System (WAAS). Although the GPS can provide wide coverage, it does not work well in indoor environments or in poor weather conditions.

The most popular indoor positioning technologies involve RF-based positioning. However, they are limited, lacking a high positioning accuracy and suffering from signal drift problems. Other indoor positioning techniques, e.g. ultrasound, infrared and RFID, have high start-up or maintenance costs. In order to overcome the signal drift issue, inertial measurement units (IMU) have been utilized to detect stepping information from pedestrians. The walking or halting status can be used to relieve the drift problem, and their trajectories can be used to improve the positioning accuracy. Augmented Reality (AR) techniques have been used to enhance positioning systems. For example, in [3] and [4], AR tags were treated as location checkpoints. In [5], an AR user interface was utilized to calculate view angles between POIs. However, the proposed gradient algorithm suffered for the local minima problem.

In this paper, we implement an AR user interface that allows users to recognize POIs from photos and we also propose a similar-triangle-based algorithm to solve the photo positioning problem. The proposed system is unaffected by weather conditions, has high positioning accuracy with low startup and maintenance costs, and only requires g-sensors and m-sensors. In short, it is a practical solution with high positioning accuracy. The rest of this paper is organized as follows. In Section II, we introduce the principle of photo imaging and coordinate systems that will be used in this work. In Section III, the proposed photo positioning system including the concepts, user interface, and system architecture,
is introduced. In Section IV, we present the similar-triangle-based positioning algorithm to calculate the location where a photo was taken. In Section V, the experimental results with a comparison with a view-angle-based algorithm are provided. Finally, Section VI gives our conclusions.

II. PHOTO IMAGING AND COORDINATE SYSTEMS

Below, we introduce photo imaging and mathematical backgrounds that will help us to design algorithms to solve the photo positioning problem.

A. Principle of Photo Imaging

Fig. 1 illustrates an example of photo imaging. Let \( L \) be a positive lens with focal length \( f \). \( c \) is the center of the lens, which is the optical center of \( L \). \( f \) is a focus of \( L \). The distance between \( c \) and \( f \), denoted as \( d_f \), is the focal length of \( L \). The line perpendicular to the lens and passing \( c \) and \( f \), denoted as \( A \), is the optical axis of \( L \). \( O \) to the left of \( L \) is an object, and \( I \) to the right of \( L \) is the image of \( O \). The distance between \( O \) and \( L \) is called the object distance and denoted as \( d_o \); and the distance between \( c \) and \( I \) is called the image distance and denoted as \( d_i \). The angle \( \alpha \) is called the view angle of object \( O \).

![Fig. 1. The relation in photo imaging.](image-url)

The relation among \( f \), \( d_o \) and \( d_i \) can be described by the thin lens equation

\[
\frac{1}{d_o} + \frac{1}{d_i} = \frac{1}{f}. \tag{1}
\]

If we take a photo, \( d_o \) is the distance from the object to the camera lens and \( d_i \) is the distance from the camera lens to the film. From Eq. (1), we have

\[
d_i = \frac{1}{1 - \frac{f}{d_o}}. \tag{2}
\]

Especially, if \( d_o \gg f \), \( d_i \) can be approximated by \( f \). Actually, this is the most case in photographing, and we use \( f \) to approximate \( d_i \) in the following discussion.

As illustrated in Fig. 1, we draw a virtual image \( I' \) in front of the lens at the distance of \( d_i \). \( I \) and \( I' \) are mirroring and reversed. Let \( a \) (and \( a' \), respectively) denote the intersections of \( A \) and \( O \) (and \( A \) and \( I' \), respectively). Let \( b \) and \( b' \) denote the one end of \( O \) and \( I' \) to the same side of \( A \). We have the following similarity relation

\[
\bar{ab} : \bar{bc} : \bar{ca} = \bar{a'b'} : \bar{b'c'} : \bar{ca}'. \tag{3}
\]

\[
\approx \bar{a'b'} : \bar{b'c'} : f. \tag{4}
\]

When a photograph is taken, the image is recorded on a film gauge or converted by a CCD or CMOS device into electronic signal and saved as a file. Traditionally, the most popular film gauges are the 35mm still photography films that are also known as 135 films. The size of 135 films is \( 35\text{mm} \times 24\text{mm} \). Compared to the traditional film, the size of the CCD or CMOS device in a digital camera, usually not known by users, varies from one model to another. It is not so convenient to get the real image distance and the diagonal of the film to calculate the field of view. To get ride of the problem, the concept of the Effective Focal Length (EFL) is introduced to have a standard description. Let \( D_{135} \) denote the diagonal length of 135 films, \( D \) denote the real diagonal length of a non-135 film, e.g., CCD or CMOS, and \( f \) denote the real focal length of the camera lens. The EFL of the lens is given by

\[
f_{EFL} = D_{135} \times \frac{f}{D}. \tag{5}
\]

B. Coordinate Systems

The Earth frame that is used to describe the 3D space within a small range near the Earth’s surface is based on the local tangent plane to the Earth’s surface. In the Earth frame, the x-axis, y-axis and z-axis respectively point to the north, east and center of the Earth. The Earth frame works well as long as the Earth’s surface can be treated as a plane. Let \( \mathbf{x}_E, \mathbf{y}_E, \mathbf{z}_E \) respectively denote the unit vector of the x-axis, y-axis and z-axis in the Earth frame. Then, \( E = \{\mathbf{x}_E, \mathbf{y}_E, \mathbf{z}_E\} \) is the standard base of the Earth frame. For each vector (point) \( \mathbf{v} \), \( [\mathbf{v}]_E = [v_1 \ v_2 \ v_3]^T_E \) denote the coordinate of \( \mathbf{v} \) with respect to the Earth frame.

The coordinate system with respect to the AR device (smartphone) is called the AR frame. The origin of the AR frame is at the center of the lens of the device. The x-axis, y-axis and z-axis of the AR frame respectively point to the right, down and front of the device. Here we assume that the z-axis coincides with the central axis of the lens. Let \( \mathbf{x}_{AR}, \mathbf{y}_{AR}, \mathbf{z}_{AR} \) respectively denote the unit vector of the x-axis, y-axis and z-axis in the AR frame. Then, \( AR = \{\mathbf{x}_{AR}, \mathbf{y}_{AR}, \mathbf{z}_{AR}\} \) is the standard base of the AR frame.

Let \( T_{E \rightarrow AR} \) be the coordinate transformation matrix from the Earth frame to the AR frame, and \( T_{AR \rightarrow E} \) vice versa. Note that \( T_{E \rightarrow AR} \) and \( T_{AR \rightarrow E} \) are the inverse matrix of each other. \( T_{E \rightarrow AR} \) can be obtained from the readings of the g-sensor and m-sensor. Let \( \mathbf{g} \) and \( \mathbf{m} \) respectively denote the readings of the g-sensor and m-sensor. Assume the coordinate systems of the embedded sensors are aligned with the AR frame, and \( \mathbf{m} \) points to the north. Besides, in a steady state, \( -\mathbf{g} \) points vertically to the ground. So, we have

\[
\mathbf{I} = T_{AR \rightarrow E} \begin{bmatrix} \mathbf{g} \times (m \times \mathbf{g}) & m \times \mathbf{g} & -\mathbf{g} \end{bmatrix}_{\| \mathbf{g} \times (m \times \mathbf{g}) \|} \begin{bmatrix} \mathbf{m} \times \mathbf{g} & m \times \mathbf{g} & -\mathbf{g} \end{bmatrix}_{\| \mathbf{g} \times (m \times \mathbf{g}) \|} \tag{6}
\]
where $I$ is the identity matrix and $\times$ represent the vector cross product. Hence,

$$\mathbf{T}_{E\rightarrow AR} = \begin{bmatrix} \mathbf{g}(\mathbf{m} \times \mathbf{g}) & \mathbf{m} \times \mathbf{g} & -\mathbf{g} \\ \mathbf{g} \times (\mathbf{m} \times \mathbf{g}) & \mathbf{m} \times \mathbf{g} & -\mathbf{g} \end{bmatrix}, \text{ and}$$

$$\mathbf{T}_{AR\rightarrow E} = \begin{bmatrix} \mathbf{g}(\mathbf{m} \times \mathbf{g}) & \mathbf{m} \times \mathbf{g} & -\mathbf{g} \\ \mathbf{g} \times (\mathbf{m} \times \mathbf{g}) & \mathbf{m} \times \mathbf{g} & -\mathbf{g} \end{bmatrix}^{-1}.$$

By the end of this section, we define the screen frame. The screen coordinate system, called the screen frame, is used to denote the point on a screen. The $x$-axis and $y$-axis respectively point to the right and down of the screen. In this work, we also assume the origin of the screen coordinate is at the center of the screen, that is denoted as $c'$.

### III. Photo Positioning

The "positioning" here means finding the position where a photographer took a picture. The location of a photographer can be estimated by analyzing the POIs in the photo. To implement a positioning system based on the ideas introduced above, we need some supportive information systems for developing new positioning techniques.

- **POI database**: A POI database is needed for recognizing POIs from photos and also for providing the locations of POIs in the real world.
- **POI recognition techniques**: POIs must be recognized from photos. This can be done automatically, manually or even via utilizing tags/codes on POIs. Especially, the coordinate of each POI image in the photo must be known.
- **Positioning algorithms**: From the two information aforementioned, an algorithm is needed to calculate the location where the photographer took the photo.

In this work, we design an AR interface to recognizing POIs in a photo and also finding the positions where the POIs locates. Therefore, the major technical issue remained is how to calculate the location of the photographer. Below, we formally define the photo positioning problem.

**Definition** Assume $n$ POIs are recognized from a given photo. The (3D) location of the $i$-th POI in the real world is represented by $\mathbf{p}_i$, and the (2D) coordinate of the $i$-th POI in the photo is represented by $\mathbf{p}_i'$. Give $(\mathbf{p}_i, \mathbf{p}_i')$ for $i = 1, 2, \ldots, n$, the photo positioning problem is to find the location where a photographer took the photo.

### A. AR User Interface

To implement a photo positioning system, we have to recognize POIs from a photo and get their coordinates in the screen frame and their locations in the Earth frame. For the sake of practical consideration and to avoid using complicated image recognition techniques, we design an AR user interface to interact with users to learn the required information.

At first, a coarse location information of users, e.g., rough position and heading, is needed in our design to find nearby POIs by querying the POI database. Then, AR objects of nearby POIs, e.g., including their images and names, will be displayed on the screen for users to recognize them from the photo. Users drag the AR objects and then drop them on their images. The positions where the AR objects are dropped indicate the coordinates in the screen frame. Besides, AR objects themselves have their coordinates in the real world. Therefore, we can get all required information by using this AR user interface as illustrated in Fig. 2.

In the snapshot, there are seven POIs marked by black square and labeled from 1 to 7 on the walls. After the photo is taken, the system queries the POI data and displays AR objects (in red) on the touch screen. Then, the user can drag-and-drop those AR objects. After all, he/she presses "Positioning" button to calculate the location of the user.

### B. System Architecture

The proposed system architecture is illustrated in Fig. 3. It contains five modules including the UI module, the POI database module, the Camera module, the Positioning module and the AR matching module. The UI module provides user interface including taking a photo, displaying AR objects on a photo and showing some navigation information. The Camera module is triggered by the UI module to take a photo for positioning. The POI database module can be an offline pre-loaded POI package or an online POI server on the Internet. The Positioning module can provide initial location by coarse positioning function, and it helps to calculate the precise position of user. The AR match module is used to match AR POI objects with POIs in a photo. The UI module is the main process which uses initial location and heading information to query the POI database module to display nearby AR POI objects. At startup of photo positioning, user takes one photo and drag-and-drop the AR POI objects to match the POIs on the photo as Fig. 2. User uses the AR match module to match AR POI object with related POI in a photo. The real coordinate in Earth frame of POI objects are known and the coordinate in AR frame can be obtained from aforementioned drag-and-drop POI interactions. The UI module uses coordinate of POI objects in Earth frame and AR frame to get precise location. The detail of our positioning algorithm will be given in Section IV.
IV. POSITIONING ALGORITHM

The geometry of photo imaging can be interpreted by the simplified model illustrated in Fig. 4. Remind that \( f \) is the EFL, and we assume \( d_l = f \). The screen is virtually placed in front of the lens at a distance of \( f \) and perpendicular to the central axis of the lens. In addition, the central axis of the lens is through \( c' \), the center of the screen, and the \( x \)-axis (and \( y \)-axis, respectively) of the screen is parallel to the \( x \)-axis (and \( y \)-axis, respectively) of the AR frame. Let \( p = \begin{bmatrix} p_x & p_y & p_z \end{bmatrix}^T_E \) denote an object, and \( p' = \begin{bmatrix} p'_x & p'_y \end{bmatrix}^T_S \) denote the image of the object. A similar triangle relation exists the projection between objects and images and can be explicitly written down as

\[
p'_x : p'_y : f = \begin{bmatrix} p_{AR,x} \mid p_{AR,y} \mid p_{AR,z} \end{bmatrix} \quad (7)
\]

Based on the similarity relation, the Photo Positioning Problem can be formulated as below. However, a problem here is that the coordinate of \( p \) is available with respect to the Earth frame, but in Eq. (7), what we need is the coordinate in the AR frame. Based on the coordinate transformation introduced in Subsection II-B, we have

\[
[p]_{AR} = T_{E \rightarrow AR} ( [p]_E - [c]_E) \quad (8)
\]

Problem 1 Photo Positioning problem

**Input**

Given \( f \) and POI pairs \((p_i, p'_i)\) for \( i = 1, 2, \ldots, n \) where \( p_i = \begin{bmatrix} p_{xi} & p_{yi} & p_{zi} \end{bmatrix}^T_E \) is the 3D coordinate of the \( i \)-th POI in the Earth frame, and \( p'_i = \begin{bmatrix} p'_{xi} & p'_{yi} \end{bmatrix}^T_S \) is the 2D coordinate of the image of the \( i \)-th POI on the screen.

**Output**

Find \( c = \begin{bmatrix} c_x & c_y & c_z \end{bmatrix}^T \) such that \( p'_x : p'_y : f = [p]_{AR,x} : [p]_{AR,y} : [p]_{AR,z} \)

Theoretically, for a non-degenerated case, two POIs can provide enough information to find \( c \). However, more POIs can help to reduce possible errors. But at the same time, contradiction will be the most cases due to measurement errors. Therefore, we suggest applying the *Least Square Method* (LSM) to solve the approximate system.

V. EXPERIMENT

The experiment is implemented on the second floor of a building as illustrated in Fig. 5. The dimensions of the floor print in Fig. 5 is 32.5 m by 19.2 m. In the experiment, 7 POIs are distributed in the environment. In Fig. 5, the POIs are marked by red circles and labeled by \( p_1, p_2, \ldots, p_7 \). The coordinates of POIs in the real world can be found in Fig. 6. Note that the environment is the same as the one illustrated in Fig. 2. 7 photos are taken at the locations labeled by \( t_1, t_2, \ldots, t_7 \) and marked by blue squares. They are called test locations. The distance between two neighboring test locations is 1 meter.

The proposed similar triangle based positioning algorithm, denoted as ST, will be compared with the view angle based positioning algorithm given in [5], denoted as VA. Since VA is a gradient algorithm, the local minima problem is a major concern. The number of local minima in VA decreases as the number of reference POIs increases. An example is illustrated in Fig. 7 and Fig. 8 in which the big starts are POIs and the blue dots are local minimum. We can see the number of local minima decreases significantly.

Since the performance of gradient algorithms may be significantly affected by initial values, we give two classes of initial
Fig. 6. AR-P experiment environment

Fig. 7. Local minima with 3 POIs

Fig. 8. Local minima with 7 POIs

Fig. 9. Average positioning results with different test locations

Fig. 10. Positioning result affected by reference POIs

The positioning errors in meter, and x-axis represents the test locations. For each test locations, 7 POI combinations are used to run the positioning algorithms. The \( t_7 \) is the farthest test location to POI \( p_4 \) and \( t_1 \) is the closest one. The small view angles, such as \( \angle p_3 t_7 p_4 \) and \( \angle p_4 t_7 p_5 \), are easily impacted by the error of user interaction. The positioning error of algorithm VA-G and VA-B are in direct proportion to distance between test location \( t_i \) and POI \( p_4 \), where \( i \) is from 1 to 7. Overall algorithm ST has the best positioning accuracy, and algorithm VA-B has the worst positioning accuracy.

The existence of local minima problem in VA is the major reason. The average positioning errors of ST, VA-G and VA-B are respectively 74.3 cm, 88.9 cm and 246.0 cm.

We also investigate the selection of reference POI combination. The positioning result of distant POI combinations is listed in Fig. 11. For convenience, the POI combination of POI \( p_1 \) and \( p_3 \) is denoted as \( \{1, 3\} \). The POI combinations \( \{1, 2\} \) and \( \{6, 7\} \) at the near plane have better positioning result than POI combinations \( \{3, 4\}, \{3, 5\}, \{4, 5\} \) and \( \{3, 4, 5\} \) at far plane. POI combinations that POIs are closed or nearly in
the same line would affect the positioning result due that the similar relation is used in Similar algorithm. In Fig. 6, the image position of the POI $p_1$, $p_3$, $p_4$, $p_5$, and $p_6$ are similarly aligned with one horizontal line. The positioning error of these closed POI combinations is as illustrated in Fig. 12. POI combinations $\{1, 3\}, \{1, 4\}, \{5, 6\}$ and $\{1, 3, 4\}$ would have poor positioning result compared to other combinations.

VI. CONCLUSIONS

Most current commercial positioning solutions cannot meet the requirement of highly accurate positioning service for LBS applications. In this paper, we propose a Photo Positioning approach based on Geometric Similarity in photo imaging. A prototype system has been developed. The Similar Triangle algorithm does not have local minima problems and have a better average positioning error of 74.34 cm. Our system can be combined with other positioning technique such as the GPS or RF-based positioning with feedback information routing (FIR) in [6] to implement an real-time and highly accurate evacuation system in [7].

ACKNOWLEDGMENT

Thanks for Yaun-Chou Cheng to implement the prototype of AR-Positioning. Thanks for Pei-Liang Shih and Po-Jung Chiu to do related experiments.

REFERENCES