A Line-based Palm-top Detector for Mobile Augmented Reality

Haruhisa Kato
KDDI R&D Laboratories Inc.
2-1-15 Ohara Fujimino-shi
Saitama, Japan
hkato@kddilabs.jp

Akio Yoneyama
KDDI R&D Laboratories Inc.
2-1-15 Ohara Fujimino-shi
Saitama, Japan
yoneyama@kddilabs.jp

ABSTRACT
We propose a marker-less Augmented Reality (AR) application based on a realtime hand posture estimation technique for smartphones. A conventional marker-less AR system does not have sufficient accuracy and speed in the detection of a mobile device. This paper presents a fast hand posture estimation algorithm based on a combination of feature points and feature lines that consist of the boundary of fingers. The proposed method realized rendering of virtual 3D models on a hand over 12 frames per second (fps) on a smartphone. Simulation results show that we can archive about 73% complexity reductions and be more accurate than the conventional method.

Categories and Subject Descriptors
H.5.1 [Information Interfaces and Presentation]: Multimedia Information Systems—Artificial, augmented, and virtual realities; I.3.7 [Computing Methodologies]: Computer Graphics—Three-dimensional Graphics and Realism; I.4.8 [Computing Methodologies]: Image Processing and Computer Vision—Scene Analysis

General Terms
Experimentation

Keywords
marker-less augmented reality, hand posture estimation, homography matrix

1. INTRODUCTION
Augmented reality (AR) is one of the user interfaces to support a user by integrating virtual information into the real world. A smartphone is a suitable platform for AR systems because of its mobility and the included camera, screen and various sensors [2,3].

In mobile AR applications, the computer vision technique estimates the position and posture of artificial tags (markers) that consist of a predefined shape and size for ease of recognition. Marker-based AR is popular due its accuracy and robustness [1,5]. However, it is necessary to setup unnatural markers beforehand.

In this paper, we present a marker-less AR system for a user’s closed hand in order to eliminate the need for markers. A hand is a ubiquitous user interface for all kinds of applications in the real world. Our work allows a user to intuitively control virtual information on the palm.

The rest of this paper is organized as follows. The overall architecture of the system and details of our approach are described in Section 3. In Section 4, we implement and evaluate the system. Conclusions and future work are provided in Section 5.

2. RELATED WORK
To dispense with the marker, previous studies on marker-less ARs reported the recognition of common objects such as the hand. For example, the conventional hand-based method [7] detects the skin color region first and finds parts of big curvatures from the region as fingertips. However, the method needs much computation for an ellipse fitting fingertip detection. Reduction of computation is necessary to realize real time mobile AR. In addition, the distance between the camera and the hand are significantly limited. Specifically, the fingertips should be close enough to look like an ellipse. Meanwhile, the hand should be far enough away so that all fingers are inside the camera view.

Another conventional method [6] is to estimate the posture of the palm by finding the convex hull that contains the entire skin area. Using color information, hand posture estimation is several times faster than the conventional method [7]. In addition, the method [6] can be estimated even if the fingers do not look like an ellipse. However, the problem is that recognition accuracy depends on skin color extraction. In general, a color-based extraction method is sensitive to background and lighting conditions. Moreover, conventional methods [6,7] fail to recognize whether one fingertip goes out of camera view since these methods estimate the position using coordinates for all fingertips. A general smartphone camera does not have a sufficiently wide angle to capture the whole hand. A user should keep the proper distance between the smartphone and the hand, so it deteriorates usability significantly. For example, when the user gets closer to see detailed information as superimposed on the hand, conventional methods cannot estimate whether
fingertips go outside the camera view. Since the fingertip is detected as the big curvature or vertices of a convex hull, conventional methods cannot support the closed hand.

In order to solve these problems, we propose a markerless AR with a fast hand posture estimation algorithm for smartphones. The proposed method uses line segments for the hand, thus the hand posture can be estimated correctly even if the fingertips are lost from view. Also, it is possible to estimate accurately using lines connected to the endpoints of the segment, even though the camera is closer to the palm, and the fingertips disappear from camera view.

3. Markerless Palm-Top AR System

3.1 System Overview

The architecture of the system (Fig. 1) consists of three main components: line detection, posture estimation and scene composition. Line detection recognizes the feature line in the top of the palm. Posture estimation calculates the Homography matrix composed of the rotation and translation. Scene composition imports the virtual model and augments it on the palm-top.

The recognition target in a captured image is a closed hand that is better than an open hand to reduce the impact of the background. In addition, we thought that a line was easier to detect than a point, which does not depend on distance. Many lines can be found in a hand as a palm feature. Even so, hand wrinkles are not suitable as features because they differ greatly in individuals. The proposed method uses finger boundaries of the closed hand as feature lines. At the same time, it also used the end of the border as feature points. However, in order to uniformly calculate feature lines and feature points in the posture estimation process, connecting together the feature points, all features are converted into line segments.

3.2 Line detection

First, the target region is roughly extracted using color information. This process is performed in order to speed up the following processes; region extraction need not be exactly like the conventional method [6]. After reducing the image size, the skin color region is extracted using the Gaussian statistical model [4]. Both skin color and non-skin color are statistically modeled to be robust under all lighting conditions and captured skin colors. The Gaussian statistical models learn skin and non-skin colors independently.

Next, the largest region is selected as a closed hand area (Fig. 2 b). Then, edge detection extracts dark line segments inside the hand area (Figs. 2 c, d). After removing the noise and the short edge, long edges at the neighborhood are integrated. Hough transform extracts the representative edge and endpoints (Fig. 2 e).

Finally, the extracted edges and endpoints are evaluated as to whether they correspond to the characteristics of the palm. Since the boundaries of the fingers are the most dominant lines in the hand, the lines are selected that satisfy the following conditions.

- The line segment is longer than a predetermined threshold.
- Line color is darker than a predetermined threshold.
- Color change is less than a preset threshold in the vicinity of the line.

![Figure 1: The architecture of the markerless palm-top AR system.](image1)

![Figure 2: Processes of Line detection and Posture Estimation.](image2)
• The difference between the slope of the line segments is less than a preset threshold.

3.3 Posture Estimation

The selected lines are related to lines in the palm by utilizing the deviation of the coordinates. Specifically, after determining the center coordinates of each segment, the segments of the thumb are determined as the coordinates to maximize the mean difference. Since the projective transformation of the palm keeps the relative position of the lines, each finger is assigned depending on the distance from the line segment of the thumb. Finally, the Homography matrix is obtained using the line segments. However, the borders of the fingers are almost parallel (Fig. 2 e). When using parallel lines directly, the results tend to converge unstably. Therefore, the proposed method calculates the Homography matrix by deriving a new line segment connecting the endpoints of the different line segments.

The line derivation process eliminates the pair of linear dependences where three line segments share one point.

Proof. Suppose three lines, \( a_0x + b_0y + 1 = 0 \), \( a_1x + b_1y + 1 = 0 \), \( a_2x + b_2y + 1 = 0 \) intersect at one point \((u, v)\),

\[
\begin{align*}
    a_0u + b_0v + 1 &= 0 \quad (1)
    a_1u + b_1v + 1 &= 0 \quad (2)
    a_2u + b_2v + 1 &= 0 \quad (3)
\end{align*}
\]

Replacing the terms before and after,

\[
\begin{align*}
    aa_0 + vb_0 + 1 &= 0 \quad (4)
    a_1 + vb_1 + 1 &= 0 \quad (5)
    a_2 + vb_2 + 1 &= 0 \quad (6)
\end{align*}
\]

This shows that \((a_0, b_0), (a_1, b_1)\) and \((a_2, b_2)\) are collinear points that lie on the single line \(ax + vy + 1 = 0\) in \(ab\) space.

Constraints are less than the unknowns, so the solution is not required to be unique. Thus, the line derived process generates a set where three lines do not intersect at one point. In preliminary experiments that examine the combination of the line segments, we found that the optimal combination is the lines as shown in Fig. 3. This combination was more accurate to use four endpoints with co-linearity.

Fig. 2 f shows the result of inverse transformation using the Homography matrix obtained from the combination in Fig. 3.

The selected line \(l\) and the corresponding line \(l’\) will be represented by the following equation as the coefficient of each line.

\[
l = (a, b, 1)^T
\]

\[
l’ = (a’, b’, 1)^T
\]

Here, \(T\) denotes the transpose operator. The relations with the coordinates \(x = (x, y)^T\) can be expressed as follows.

\[
l’^T x = 0
\]

\[
l''^T x’ = 0
\]

In general, any coordinate \(x\) is mapped to the coordinates \(x’\) in the matrix Homography \(H\). However, for scale-independence, the following equation holds over the coefficient \(s\).

\[
sx’ = Hx
\]

Here, \(H\) is represented by the following \(3 \times 3\) matrix.

\[
H = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix}
\]

(12)

Assigned to the Eq. (10) and the Eq. (11), the following equation is obtained from the Eq. (9).

\[
s l^T = l'^T H
\]

\[\therefore s l = H l'
\]

Rewriting the vector and matrix elements,

\[
s \begin{pmatrix} a' \\ b' \\ 1 \end{pmatrix} = H \begin{pmatrix} a \\ b \\ 1 \end{pmatrix} = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \begin{pmatrix} a' \\ b' \\ 1 \end{pmatrix}.
\]

(15)

Expanding on each line,

\[
as = h_{11}a' + h_{21}b' + h_{31}
\]

(16)

\[
sb = h_{12}a' + h_{22}b' + h_{32}
\]

(17)

\[
s = h_{13}a' + h_{23}b' + h_{33}.
\]

(18)

Except for the scaling factor \(s\),

\[
a(h_{13}a' + h_{23}b' + h_{33}) = h_{11}a' + h_{21}b' + h_{31}
\]

(19)

\[
b(h_{13}a' + h_{23}b' + h_{33}) = h_{13}a' + h_{23}b' + h_{32}.
\]

(20)

Returning again to the matrix representation,

\[
\begin{pmatrix} -a' & 0 & 0 \\ 0 & a' & -b' \\ 0 & -a' & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = 0. \tag{21}
\]

To simplify the notation of Eq.(21), the left matrix and the vector are represented by \(A\) and \(h\) respectively. The Homography matrix computation is equivalent to solving the following equation.

\[
Ah = 0 \quad (22)
\]
Here, \( \mathbf{0} \) is the \( 2 \times 1 \) vector that consists of all zeros. Since the above equation represents the relationship between a single pair of lines, matrices and vectors will be expanded to accommodate more than one pair. That is, in the case of \( n \) pairs of lines, \( \mathbf{A} \) and \( \mathbf{0} \) are represented as \( 2n \times 9 \) matrix and \( 2n \times 1 \) vectors, respectively. In addition, multiple lines are normalized beforehand. Lines \( l \) and \( l' \) are normalized by \( N \) and \( N' \) matrices, and represented by lines \( L \) and \( L' \), respectively. They are treated as lines \( l \) and \( l' \) again.

\[
L = NL \tag{23}
\]

\[
L' = N'L' \tag{24}
\]

After calculating the matrix \( H_l \) corresponding to the normalized lines \( L \) and \( L' \), the matrix \( H_l \) corresponds to the original lines \( l \) and \( l' \) will be calculated as follows:

\[
H_l = N^{-1}H_lN \tag{25}
\]

In this case, we obtained \( h \) that minimizes the norm of the left-hand side in Eq. (22).

\[
\min ||Ah|| \tag{26}
\]

Minimizing the norm as the sum of squares, \( h \) is the eigenvector corresponding to the minimum eigenvalue of \( A^TA \).

\[
\min h^TA^TAh \tag{27}
\]

4. SIMULATION RESULTS

To compare the proposed method with the conventional method \([7]\), an experiment was performed on a desktop PC (Windows NT6.1 SP1, Core i7 CPU 2.67GHz, 3GB memory). In the experiment, a 3D model was overlaid on a hand in a video recorded beforehand (640x480, 15fps, 363 frames of 408 frames include a hand) as shown in Fig. 2 (a).

Table 1 shows the processing time. The proposed method reduced 73.0\% (= 1 - 6.19 / 22.90) of the time required for the feature detection process. As a result, the operation speed of the entire application is about 3.30 times faster than that of the conventional method.

Table 2 shows the number of the accuracy of posture estimation. The conventional method only lost the hand (11.29\% = 1 - 322 / 363), the proposed method stably detected the palm-top (0.55\% = 1 - 361 / 363). The proposed method realized fast AR application without computationally expensive ellipse fitting and feature extraction.

![Figure 4: a virtual 3D model on the palm-top.](image)

We implemented the proposed method for a smartphone (Sony Ericsson IS11S, Android 2.3) as shown in Fig. 4. The AR application performed at 12.2 fps due to the fast algorithm. Even when there were no fingertips in a frame, the posture estimation was able to detect the hand correctly by using the connected feature lines.

5. CONCLUSIONS

We presented a palm-top AR system using line segments inside the user’s hand. The posture of palm is estimated instead of a marker to establish the three-dimensional coordinate system. Our method is simple but efficient based on the feature lines that connected feature points. The simulation results showed that the proposed method was 3.3 times faster and more accurate than the conventional method. In future work, the robustness of the system will be improved to different lighting condition and background.

6. REFERENCES