KSII TRANSACTIONS ON INTERNET AND INFORMATION SYSTEMS VOL. 4, NO. 4, August 2010

Dynamic Manipulation of a Virtual Object in Marker-less AR system Based on Both Human Hands

[1]Junchul Chun and [2]Byungsung Lee
[1]Department of Computer Science, Kyonggi University,
Yui-Dong, Yongtong-Gu, Suwon, Gyeonggi-Do 443-760, South Korea
[e-mail: jcchun@kgu.ac.kr]
[2] Nextеye Co., Venture Digm 512, 162-1 126-1, Pyeongchon Dongan,
Anyang, Kyunggi, 430-070, South Korea
[e-mail: lbs0116@kgu.ac.kr]
*Corresponding author: Junchul Chun

Received May 7, 2010; revised June 9, 2010; accepted June 27, 2010;
published August 27, 2010

Abstract

This paper presents a novel approach to control the augmented reality (AR) objects robustly in a marker-less AR system by fingertip tracking and hand pattern recognition. It is known that one of the promising ways to develop a marker-less AR system is using human's body such as hand or face for replacing traditional fiducial markers. This paper introduces a real-time method to manipulate the overlaid virtual objects dynamically in a marker-less AR system using both hands with a single camera. The left bare hand is considered as a virtual marker in the marker-less AR system and the right hand is used as a hand mouse. To build the marker-less system, we utilize a skin-color model for hand shape detection and curvature-based fingertip detection from an input video image. Using the detected fingertips the camera pose are estimated to overlay virtual objects on the hand coordinate system. In order to manipulate the virtual objects rendered on the marker-less AR system dynamically, a vision-based hand control interface, which exploits the fingertip tracking for the movement of the objects and pattern matching for the hand command initiation, is developed. From the experiments, we can prove that the proposed and developed system can control the objects dynamically in a convenient fashion.

Keywords: Augmented reality, marker-less AR, fingertip tracking, hand detection, skin-color model, hand pattern recognition

---

This work is supported by contents convergence software research center at Kyonggi University funded by GRRC (Geyonggi Regional Research Center) program. A preliminary version [17] of this paper was appeared in ACM ICCS 2009 Nov. 24-26. Seoul, Korea. This extended version includes a marker-less AR system with both hands in a real scene and further analysis and supporting results for dynamic manipulation of a virtual object.

DOI: 10.3837/tiis.2010.08.000

1. Introduction

Augmented Reality (AR) is a technology which allows virtual graphic imagery to exactly overlay physical objects in real time. Recently developed tracking and interaction methods in AR allow users to work with and examine the real physical world, while controlling augmented objects in the system more feasible fashion. In general, AR system requires some indication of where exactly the virtual objects should be augmented. This has been conventionally accomplished by AR markers such as ARTag. However, in recent marker-less AR is considered to be ideal since it does not require the forethought of adding markers to a scene.

Most of vision-based tracking technologies are classified into two classes; feature-based and model based approach [1]. The main idea underlying feature-based methods is to find a correspondence between 2D image features and their 3D world coordinate system. The camera pose can be found from projecting the 3D coordinates of the features into the 2D image coordinate along with minimizing the difference between their corresponding 2D features. Among the related researches, the famous ARToolKit library [2], which utilizes fiducial markers was first introduced. A method to find 3D coordinates of the four corners of a square marker was introduced by Stricker et al. [3]. Tracking algorithms for non-square visual markers such as ring shaped and circular shaped markers were also exploited [4][5]. Model-based methods tracking explicitly use a model of features of tracked objects such as 2D template object or CAD model. Comport et al [6] presented a visual servoing approach to calculate camera pose from a range of model features. In their research, by predicting hidden movement of object and reducing the outlier data, the robustness and performance of the tracking could be improved.

Meanwhile, interaction techniques for AR applications allow end users to contact with virtual objects in an intuitive way. Tangible interface introduced by Ishii [7] and tangible interaction metaphor [8] have become one of the most frequently used AR interaction methods. Tangible AR interactions leads to combining real object input with human gesture interaction and hand gesture interaction methods have been widely investigated from the perspective of computer vision and Augmented Reality [9][10]. Some of the works utilized finger tracking along with hand gesture recognition methods as AR interface in a desktop [11][12] or wearable computing environments [13][14]. Recently, Lee et al [14][15][16] introduced a method to use bare stretched human hand as a distinctive pattern instead of a marker for marker-less AR system. In their work, 6-DOF camera pose was estimated by tracking fingertips and virtual objects are augmented on hand coordinate system. The limitation of Lee's approach is the inspection of the object is restricted because the manipulation of the overlaid object depends on the movement of the hand system itself. Even though studies on the constructing the marker-less AR with hand and hand gesture recognition are done individually, however, studies for combining the both methods in the sense of dynamic manipulation of virtual objects in human hand-based marker-less AR system.

In the previous work [17], we introduce a preliminary marker-less AR system in a simple background image and show some possibility that the hand gesture can be used for interaction with the virtual model. In this work, in order to provide more feasible and extended manipulations of such a marker-less AR system, we have worked to integrate the marker-less AR with vision-based hand mouse using both hands with a single camera. Especially, we are focusing on a vision-based approach for a real time fingertip tracking in the hand coordinate system and hand gesture recognition of another hand as a hand mouse on desk top environments. Our work also provides a feasible registration method to overlay virtual objects in a real complex scene by constructing a marker-less hand coordinate system rather than using a traditional fudicial marker. The rest of this paper is constructed as follows: In section 2, the overall steps for the proposed system are described. In section 3, the way to build the marker-less AR system is introduced in detail. In section 4, hand gesture recognition with a template matching technique for the vision-based hand mouse is described. In section 5, we show the experimental results based on the proposed approach. Finally in section 6, the benefit of the work and the future works are discussed.

2. Overview of the Proposed Method

The proposed system consists of three major phases as illustrated in **Fig. 1**. In this work, bare stretched human hand plays a role of a marker to register virtual objects in a marker-less AR system therefore the hand detection should be done first from the input image frame. For this purpose, skin-color based hand segmentation is performed to extract hand region in the first phase. Subsequently, fingertips are detected by investigating the curvature of the contour from the segmented hand region. The positions of detected and tracked fingertips in the sequence of the input image frames are used for camera pose estimation. In the second phase, the camera pose is estimated using the positions of the tracked fingertips from the previous stage. At the same time, intrinsic camera parameters of a camera are calibrated using an initial checker board pattern. Such information is used for constructing a hand coordinate system. Final phase includes rendering virtual objects on central position of the hand coordinate system and controlling the objects using a hand mouse which is developed by hand pattern recognition method. Predefined hand patterns are used for translation, rotation and scaling the objects on the AR system.

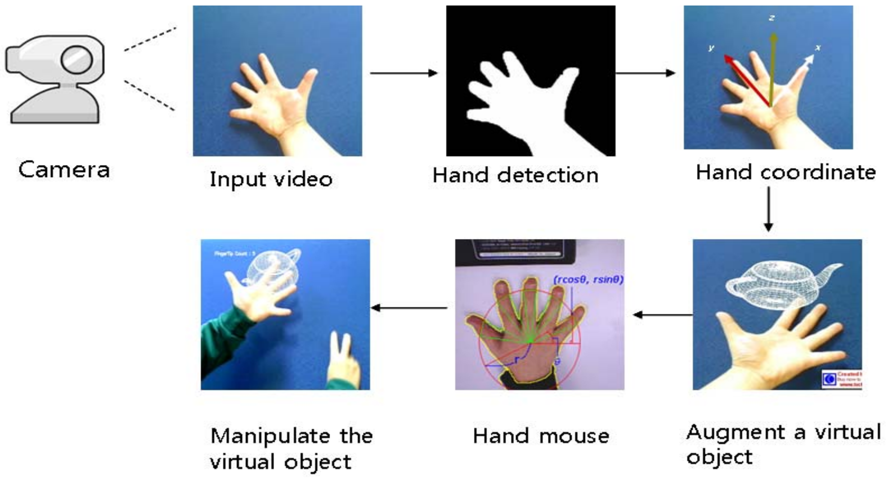


Fig. 1. Overall steps for developing a marker-less AR system and vision-based interface

3. Virtual Hand Coordinate for Marker-less AR

3.1 Hand Detection Method

For hand region detection we utilize YCbCr skin color model which is proven to detect skin region effectively to segmenting hand region. Skin color model has been widely used for hand and face detection because the use of color information can simplify the task of hand localization in complex environments. Mostly the primary components of R, G, B are used for skin segmentation. Other models such as HSI, normalized RGB, YCbCr, YIQ etc. are used for the segmentation of skin-like regions [18][19]. Even though, color information is an efficient tool for identifying skin region if the skin tone color can be properly adapted for different lighting conditions, it has some limitation since the skin color model is sensitive to the light source varies significantly or complex background.

As a skin color model we adopt the YCbCr since it is perceptually uniform and it is similar to the TSL space in terms of the separation of luminance and chrominance. It is known that the chrominance components of the skin color are independent of the luminance component. Thus, the normalized RGB color model is transformed to YCbCr model. Moreover, in this work, by disregarding the luminance component(Y), robustness of skin detection can be obtained in the case of variations in lighting conditions. The model we adopt is to define a region of skin tone pixels using Cb, Cr values from samples of skin color pixels. **Fig. 2** shows the hand region with Cb, Cr values of an input hand image. From the RGB based input hand image, we can easily generate Cb and Cr values using the following equation (1)

$$\begin{aligned} Y &= 0.299 \times R + 0.587 \times G + 0.114 \times B \\ C_b &= -0.16874 \times R - 0.33126 + 0.5 \times B \\ C_r &= 0.5 \times R - 0.41869 \times G - 0.081313 \times B \end{aligned} \tag{1}$$

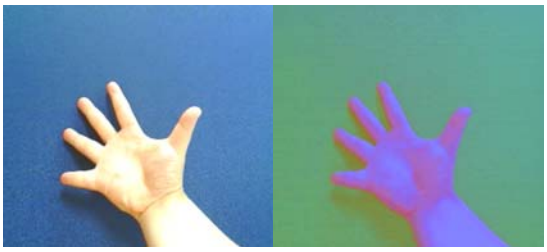


Fig. 2. Input RGB hand image(left) and C_bC_r based image(right)

The threshold values of each chrominance components are derived from the test of 30 hand images. The average values of C_b and C_r obtained through the experiments are as follows:

$$(128 \le C_b \le 170) \cap (73 \le C_r \le 158) \tag{2}$$

With carefully selected threshold range values, a pixel value is classified to the face region if its value meets the range. Subsequently, from the candidate region the exact hand is extracted precisely using the described skin color model. In order to enhance the segmented hand region, morphological operators are applied to the segmentation result. The center of the hand can be extracted from the medial axis of the segmented hand region $I(p)$. Using the distance transform [20], which is used to compute the medial axis of the segmented hand, the single connected hand is acquired. The distance transform (DT) is the transformation that generates a map D whose value in each pixel p of the segmented region O is the smallest distance from this pixel to the background O^c. The image *D*, which is called a distance map, can be defined as follow:

$$D(p) = \min\left\{ d(p,q) \middle| q \in O^c \right\} = \min\left\{ d(p,q) \middle| I(q) = 0 \right\} \tag{3}$$

Fig. 3 illustrates the procedure for hand shape detection.

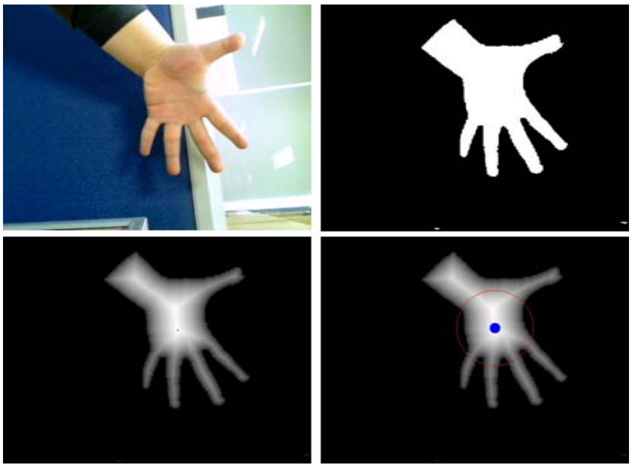


Fig. 3. Process for hand segmentation: (a) an input image (b) segmentation of hand (c) distance transform of the segmentation result (d) central position of the hand

3.2 Fingertip Detection and Tracking

In order to detect fingertips of the segmented hand region a curvature-based feature detection method is adopted. From the contour from the segmented hand, we tried to find pixels which presented peaks along the contour perimeter and regarded them the location of fingertips. At each pixel P_i in the contour of the hand segmentation, the l-curvature (K_i) becomes the angle between two vectors $\vec{v}_1(p_i, p_{i-l})$ and $\vec{v}_2(p_i, p_{i+l})$. The curvature is simply the dot product of two vectors defined as follow:

$$\kappa_l = \frac{\vec{v}_1 \bullet \vec{v}_2}{\|v_1\|\|v_2\|} \tag{4}$$

If the value of the curvature close to 0 will be peaks or valleys of the contour. The direction of the curve is considered to determine the peak values as the points of fingertips. **Fig. 4** shows the accurate fingertips detected and the central position of the hand respectively. The central position of the hand is given as the point whose distance to the closest the contour of the hand is the maximum.

Once fingertips are detected, we can track them by matching newly detected fingertips to the previously tracked fingertips. Optical flow has been known as a robust method to tracking features in the sequence of the images. However, sometimes the optical flow cannot properly cope with noise or occlusion caused by the hand motion during tracking the fingertips [17]. Therefore, we adopt matching algorithm [14] which minimizes displacement of pairs of fingertips over two frames as follow:

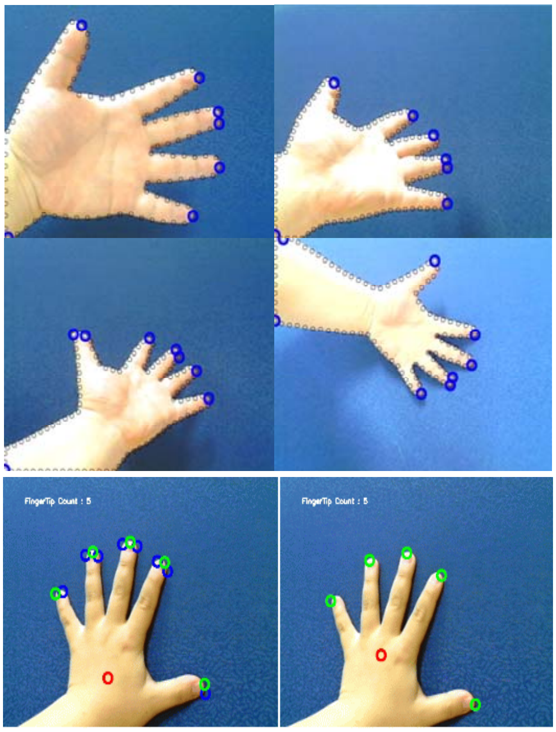


Fig. 4. Detected fingertips and the central position of the hand

$$f_i = \arg \min \sum_{j=0}^{4} \left\| \left(f_{i,j} - C_i \right) - \left(f_{i+1,j} - C_{i+1} \right) \right\| \tag{5}$$

In equation (5) f_i and f_{i+1} represent sets of 5 fingertips at i^{th} and $(i+1)^{th}$ frames, $f_{i,j}$ is the fingertip of j^{th} index in the frame f_i, and C_i is the central position of the hand, respectively. The trajectories of the fingertips can be measured by taking correspondences of detected fingertips between successive image frames in the constraints that there is no abrupt change of hand motion between the sequential images.

3.3 Camera Pose Estimation

In order to construct a hand coordinate system, camera pose estimation from the detected fingertips is necessary. For this work, an offline camera calibration is performed by use of a simple planar grid pattern of known size in the field of view. Following figure is a grid pattern we have used to estimate camera pose.

As illustrated in **Fig. 5**, from the grid calibration pattern with a stretched hand internal parameters are estimated. Once the fingertip position is estimated by using the grid pattern along with hand, from the measured fingertip positions relative to the origin corner of the grid pattern, the coordinate system is moved to center of the hand.

Fig. 6 shows the perspective projection of an object from a world coordinate (O) to image plane. In this projection M is a 3D point, m is its projection onto the image plane and C is camera coordinate system, respectively.

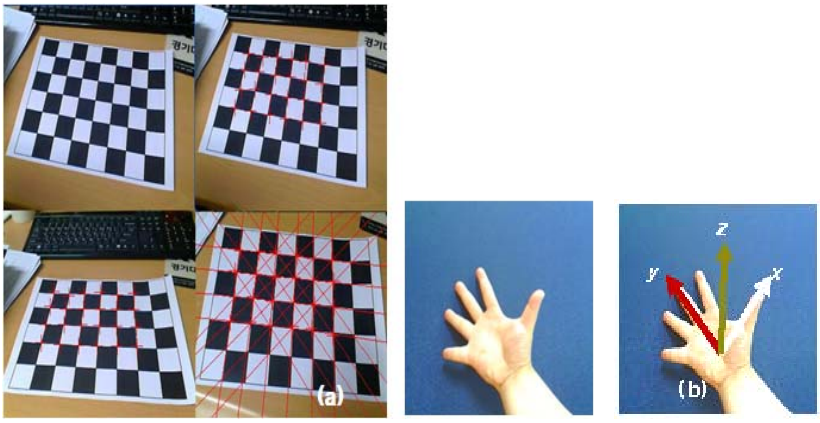


Fig. 5. (a) Grid pattern and hand posture for camera pose estimation and (b) 3D hand coordinate system

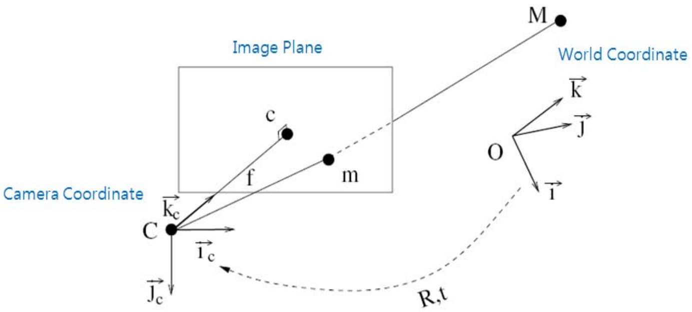


Fig. 6. Perspective projection model

Given a camera pose estimated, the 2D positions $m = (x, y)$ of fingertips detected from input image frame can be unprojected to the 3D world coordinate system by following formula:

$$s\tilde{m} = P\tilde{M} \tag{6}$$

In equation (6), s is a scale factor, $\tilde{m} = (x, y, 1)^T$ and $\tilde{M} = (X, Y, Z, 1)^T$ are the homogeneous coordinates of point m and M, and P is a 3×4 is camera projection matrix. The camera pose projection matrix P can be decomposed as:

$$P = K[R|t] \tag{7}$$

where K is the 3×3 camera calibration matrix which depends on the internal parameters of the camera such as focal length and $[R|t]$ is the 3×4 external parameters, which corresponds to the Euclidean transformation from a world coordinate system to the camera coordinate system. R and t are a 3×3 rotation matrix and a translation vector, respectively. With the hand model and the external camera parameters [22] the camera pose can be estimated during the tracking of fingertips.

4. Vision-based Hand Mouse

4.1 Hand Pattern Recognition Method

As for an interface to interact with the virtual objects registered on the hand coordinated marker-less AR system, a vision-based hand mouse is considered. As an initial stage, to improve the precision of the hand detection, the candidate region of the hand from the input imagery are determined by removing the hand shadow caused by any light variation.

In this work, a vision-based hand mouse is used for a tangible interface for the marker-less AR system. As for the hand pattern recognition, we have defined 5 different types of hand shapes: zoom in, zoom out, rotations with x, y, z respectively as illustrated in **Fig. 7**. The input hand pattern is recognized by using template matching method, which measures the similarity between input hand pattern and predefined template patterns of hand to execute a specific command which manipulates virtual objects on the marker-less AR system.

Fig. 7. Hand patterns for template matching (from left to right: zoom in, zoom out, rotation by Y-axis, rotation by x-axis and rotation by z-axis)

The Cb, Cr model used for constructing the virtual hand coordinate system is also applied for detecting hand region detection. In order to make the hand pattern recognition system robust against an apparent difference of size, we normalize the size of both hand pattern images and the detected hand in the input video image as 80×80 sized image. Thus the system is robust against the size variation of the hand in hand pattern recognition. Each hand pattern image is made by an average-value of several hand images for a specific hand pattern as shown in **Fig. 8**.

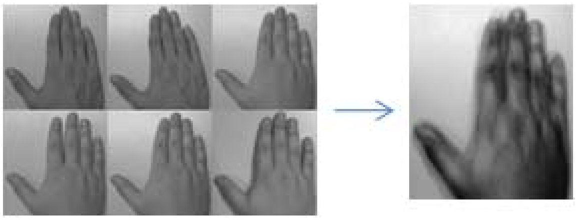


Fig. 8. The generation of a hand pattern image made by averaging 6 different hand images

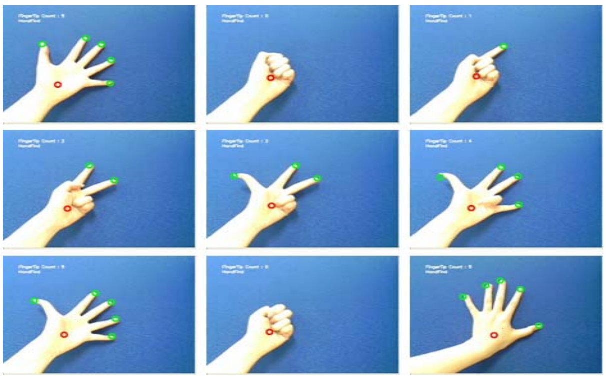
In template matching, the similarity between the registered hand patterns and input hand shape is determined by the Pearson correlation coefficient [23][24] which can be defined by

$$r = \frac{\sum_{x,y} (f(x,y) - f')(t(x,y) - t')}{\sqrt{\sum_{x,y} (f(x,y) - f')^2 \sum_{x,y} (t(x,y) - t')^2}} \quad (8)$$

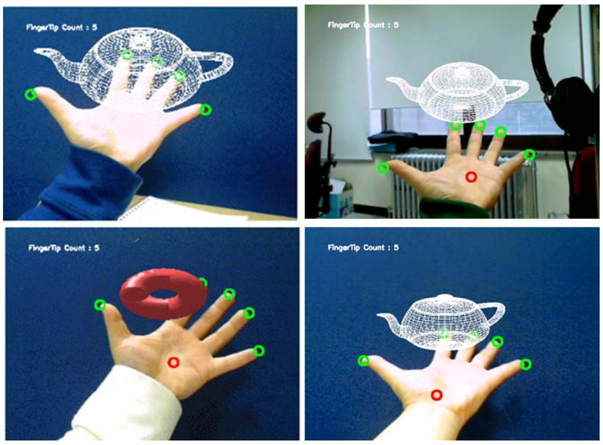
In equation (8), $f(x,y)$ is an input hand region for the hand pattern recognition, f' is the average pixel value of the hand region, $t(x,y)$ is a registered hand pattern image, and t' is the average pixel value of $t(x,y)$. The coefficient (r), which is $-1 \le r \le 1$, approaches to 1 when the two objects have the higher similarity.

5. Experimental Results

Based on the proposed method, we have manipulated some objects. In this experiment, the system is designed to run at 30fps for 640×480 resolutions. The developing environments are 2.50GHz CPU with OpenCV Library. The flowing sequence of the hand images in **Fig. 9(a)** shows a successful result of tracking fingertips. The fingertip tracking is stably successful when the hand is flipped over. Meanwhile, **Fig. 9(b)** also illustrates augmented 3D objects on the bare hand. The registered virtual object is moved according to the varying position of the bare hand.



(a) Tacking fingertips by a matching algorithm



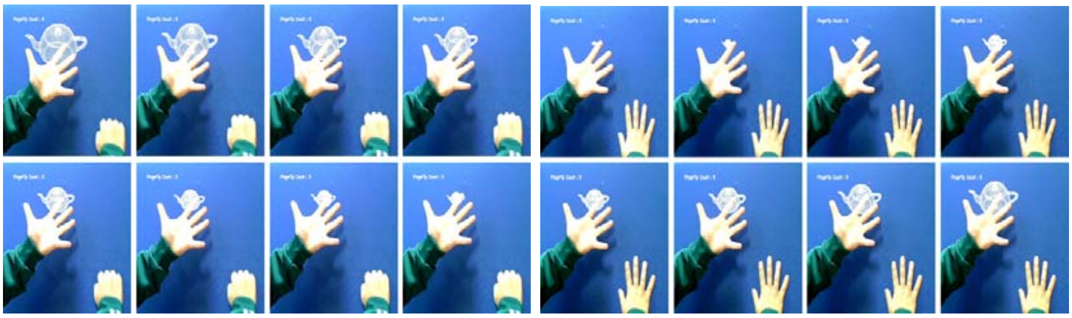
(b) An augmented 3D objects on the hand

Fig. 9. Results of fingertip tracking and an augmented 3D object on the stretched bear hand

Fig. 10 shows the results when three different hand patterns are applied to a 3D teapot. Depending on the hand patterns the augmented object can be manipulated dynamically.

(a) Rotation by x-axis

(b) Rotation by y-axis



(c)Resizing the object (zoom out)

(d) Resizing the object (zoom in)

Fig. 10. Manipulation of a 3D teapot by hand pattern recognition

In order to prove the usability of the developed marker-less AR system, the system is also tested in a conventional situation such as objects with complex background. As illustrated in **Fig. 11**, the 3D object is accurately overlaid on the bare hand and manipulated in a convenient fashion even though the background of the image is complicate.

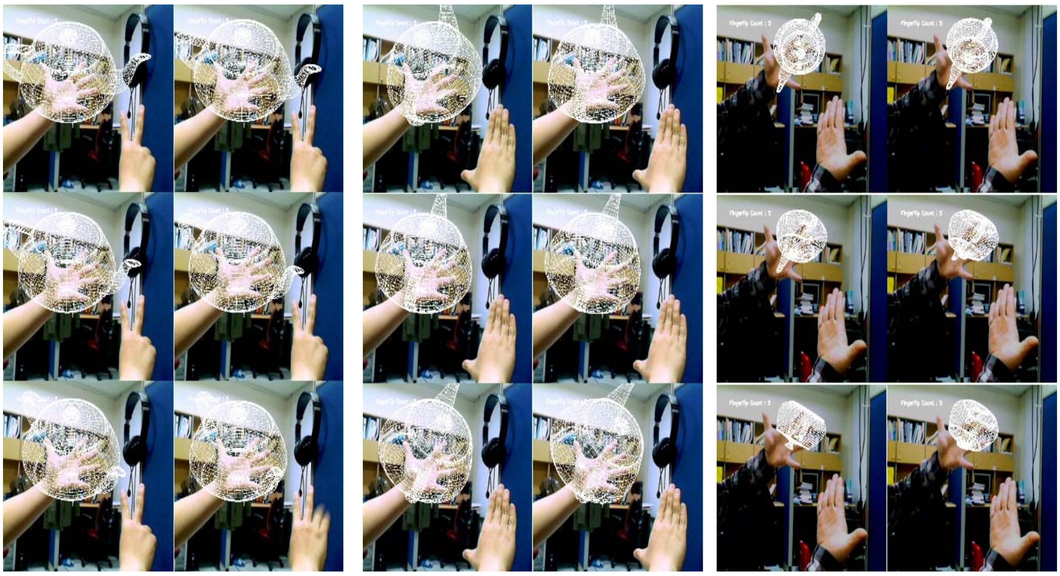


Fig. 11. Manipulation of a 3D teapot with complex background (from top left, clockwise, translation, zoom in, zoom out, rotation by z-axis, x-axis, and y-axis respectively

We also proved that the proposed system is reliable by comparing the moving trajectories of the 3D reconstructed human liver [25] which is rendered on human hand and a fudicial marker as illustrated in **Fig. 12**. By translating the 3D object on each hand coordinate system and the fudicial marker and tracing the central position of origin of hand coordinate system and that of the fudicial marker while the positions of the markers are changed, the trajectories of the object can be acquired. We can find the (x, y) positions of each trajectory are fairly similar each other when the z position is set to constant. From this experiment the developed system can be used for camera pos estimation without losing significant accuracy compared to a state-of-the-art fudicial marker system.

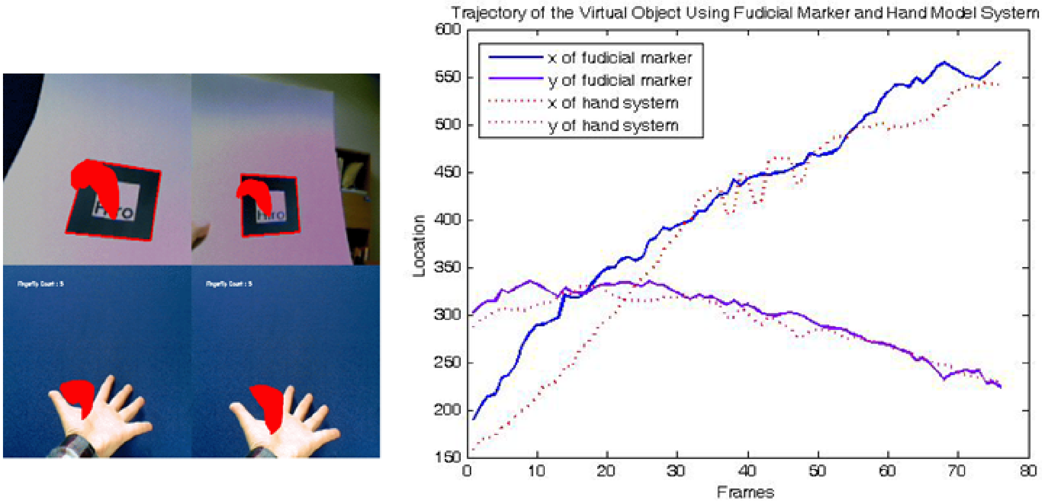
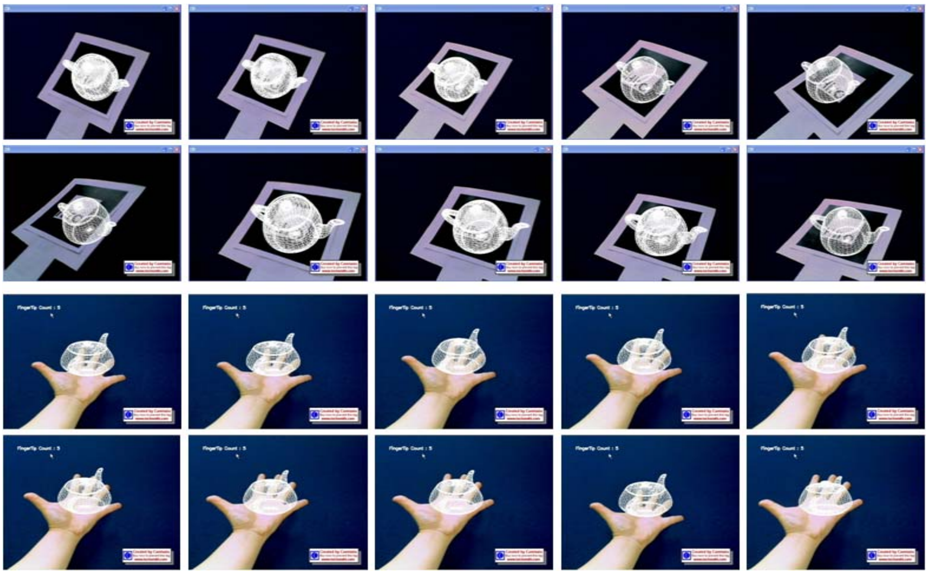


Fig. 12. Moving trajectories of a virtual human liver rendered on a traditional marker and human hand

As illustrated in the sequential images in **Fig. 13(a)**, the manipulation of the virtual object based on both a state-of-the-art ARTag marker-based and the proposed system is stable and accurate. **Fig. 13(b)** also shows sequential video frames during the dynamic manipulating a virtual teapot by using different hand patterns successfully. When the hand region is successfully detected, there are no mismatched cases in the hand pattern recognition.



(a) The manipulation of a virtual object based on the marker-based and the marker-less system

(b) The manipulation of a virtual object using various hand patterns in the proposed marker-less AR system

Fig. 13. Experimental results: (a) the manipulation of the ARTag-based and marker-less AR system and (b) the dynamic manipulation of a virtual object based on hand pattern variation

5. Conclusion and Future Work

In this work we introduced a vision-based hand mouse to manipulate virtual objects in a marker-less AR system. In this work, in order to provide more feasible and effective manipulations in a marker-less AR system, we integrate the marker-less AR system with vision-based hand pattern recognition. Human's stretched left hand played a role of the marker and the right hand is developed as a vision-based hand mouse. To develop the hand coordinate system, YCbCr skin-color based hand segmentation and curvature-based fingertip detection methods are used. In hand detection by disregarding the luminance component from YCbCr using only chrominance components the robustness of the skin detection can be obtained against the light variation. The camera pose estimated from the tracked fingertips are used to register virtual objects on the marker-less 3D hand coordinate system. Moreover, to manipulate the movement of the virtual objects dynamically, the hand mouse which exploits hand pattern recognition is developed. The manipulation of the objects is performed with an event-handling mechanism. From the experiments, we can show that the developed hand mouse interface controls the objects in a feasible fashion.

In order to improve the effectiveness of dynamic interaction in the developed marker-less AR system, first of all the system should provide a mechanism that copes with self-occlusion problem during flipping the hand. In currently developed system, the self-occlusion caused by flipping hand can cause the losing track of fingertips and missing the rendered virtual object temporally until the fingertips are detected again. In addition, 3D hand model recognition rather than using 2D hand images and haptic-based interaction between user and the virtual object are necessary for the completeness of the system.

References

[1] M. Pressigout and E. Marchand, " Hybrid tracking algorithm for planar and non-planar structures subject to illumination change," in *ISMAR'06*, pp. 52-56, 2006.

[2] H. Kato and M. Billinghurst, "Marker tracking and HMD calibration for a video-based augmented reality conferencing system" in *IWAR'99*, pp. 85-94, 1999.

[3] D. Stricker, D. Klinker, and D. Reiners, "A fast and robust line-based optical tracker for augmented reality application," in *IWAR'98, pp. 31-46*, 1998.

[4] Y. Cho, J. Lee, and U. Neumann, "A multi-ring fiducial system and an intensity-invariant detection method for scalable augment reality," in *IWAR'98*, pp. 147-156, 1998.

[5] S. Vogt, A. Khamene, F. Sauer, and H. Niemann, "Single camera tracking of marker clusters: Multiparameter cluster optimization and experimental verification," in *SIMAR'02*, pp. 127-136, 2002.

[6] A. Comport, E. Marchand, and F. Chaumette, "A real-time tracker for markerless augment reality," in *ISMAR'03*, pp. 36-45, 2003.

[7] H. Kato, M. Billinghurst,I. Poupyrev, K. Imamoto, and K. Tachibana, "Virtual object manipulation on a table-top AR environments," in *ISAR'00*, pp. 111-119, 2000.

[8] M. Kolsch and M. Turk, "Fast 2D hand tracking with flocks of features and multi-cue integration," in *CVPRW'04*, pp. 158, 2004.

[9] T. Kurata, T. Okuma, M. Kourogi, and K. Sakaue, "The hand mouse: GMM hand-color classification and mean shift tracking," in *2nd Int'l Workshop on Recognition, Analysis and Tracking of Faces and Gestures in Real-time System*, pp. 119-124, 2001.

[10] W. Lee and J. Park, “Augmented foam: A tangible augmented reality for product design,” in *ISMAR’05*, pp. 106-109, 2005.

[11] J. Letessier and F. Berard, “Visual tracking of bare fingers for interactive surface,” in *Proc. ACM symposium on User interface software and technology*, pp. 119-122, 2004.

[12] W. Mayol, A. Davison, B. Tordoff, N. Molton, and D. Murray, “Interaction between hand and wearable camera in 2D and 3D environments,” in *BMVA’04*, 2004.

[13] J. Terrillon, M. Shirazi, H. Fukamachi, and S. Akamatsu, “Comparative performance of different skin chrominance models and chrominance spaces for the automatic detection of human faces in color images,” in *Proc. 9th Conference on Automatic Face and Gesture Recognition*, pp. 54-61, 2000.

[14] T. Lee and T. Hollerer, “Hand AR: Markerless Inspection of Augmented Reality Objects Using Fingertip Tracking,” in *Proc. IEEE ISWC’07*, pp. 83-90, 2007.

[15] T. Lee and T. Hollerer, “Initializing Markerless Tracking Using a Simple Hand Gesture,” in *Proc. ACM/IEEE ISMAR’07*, pp. 1-2, 2007.

[16] T. Lee and T. Hollerer, “Hybrid Feature Tracking and User Interaction for Markerless Augment Reality,” in *IEEE VR’08 (10th Int’l Conference on Virtual Reality)*, pp. 145-152, 2008.

[17] BS. Lee and JC Chun “Manipulation of virtual objects in marker-less AR system by fingertip tracking and hand gesture recognition,” in *Int’l Conference on Interaction Science*, pp. 1110-1115, 2009.

[18] H. Ishii and B. Ullmer, “Tangible bit: towards seamless interface between people, bits and atoms,” in *ACM CHI’97*, pp. 234-241, 1977.

[19] K. Min and J. Chun, “A nonparametric skin color model for face detection from color images,” in *PDCAT’04*, pp. 115-119 , 2004.

[20] G. Borgefors, “Distance transformations in digital images,” *Computer Vision, Graphics and Image Processing*, vol. 34, pp. 344-371, 1986.

[21] M. Black, “Robust incremental optical flow,” Ph.D thesis, Yale University, 1992.

[22] Z. Zhang, “A flexible new technique for camera calibration,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no.11, pp. 1330-1334, 2000.

[23] C.C Chen and H.T Chu, “Similarity Measurement Between Images,” in *Proc. of the 29th Annual International Computer Software and Applications Conference (COMPSAC'05)*, vol.1, pp. 41-41, 2005.

[24] F. J. Anscombe, "Graphs in Statistical Analysis," *American Statistician*, vol. 27, no. 1, pp. 17–21, 1973.

[25] HJ. Park, DS. Kim, and JC Chun, “An Interactive Method for Volumetric Human Organ Model on Augmented Reality,” in *Proc. of APICT-IST and ICONI 2009*, pp. 383-386, 2009.

Junchul Chun is currently a professor in the department of computer science and a principal investigator in the Graphics and Image Processing Research Laboratory at Kyonggi University, Suwon South Korea. He received a B.S degree from Chung-Ang University majoring in computer science. He also received MS and Ph.D degrees from the department of computer science and engineering at the University of Connecticut, CT., U.S.A, in 1992 and 1995, respectively. He was a visiting research scholar in PRIP (Pattern Recognition and Image Processing) at Michigan State University and University of Colorado at Boulder, 2001 and 2009, respectively. He has been served as an IEEE workshop chair of AITA (Advanced Internet Technology and Application) in 2007 and 2008. His current research fields are 3D face modeling, vision-based animation, and human computer interaction.

Byungsung Lee is a Research Engineer at the Nexteye Co., Anyang, South Korea. He received a BS degree and an MS degree in Computer Science from the Kyonggi University, South Korea, in 2008 and 2010, respectively. His research fields are U-healthcare, vision-based motion tracking, and Augmented Reality.