Clickable Augmented Documents

Sandy Martedi #1, Hideaki Uchiyama #2, Hideo Saito #3

Graduate School on Science and Technology, Keio University
3-14-1 Hiyoshi, Kohoku-ku, Yokohama, Kanagawa, 223-8522, Japan
[1] sandy@hvrl.ics.keio.ac.jp
[2] uchiyama@hvrl.ics.keio.ac.jp
[3] saito@hvrl.ics.keio.ac.jp

***Abstract*—This paper presents an Augmented Reality (AR) system for physical text documents that enable users to click a document. In the system, we track the relative pose between a camera and a document to overlay some virtual contents on the document continuously. In addition, we compute the trajectory of a fingertip based on skin color detection for clicking interaction. By merging a document tracking and an interaction technique, we have developed a novel tangible document system. As an application, we develop an AR dictionary system that overlays the meaning and explanation of words by clicking on a document. In the experiment part, we present the accuracy of the clicking interaction and the robustness of our document tracking method against the occlusion.**

I. Introduction

One of the common frameworks for developing augmented reality (AR) systems has been introduced by Kato et. al [1] and widely known as ARToolKit [2]. The applications using ARToolKit have been explored and developed for several purposes such as for education [3] and arts [4]. The advantage of this technology is the possibility to overlay virtual objects on physical papers dynamically. Generally, this technology is known as a paper-based AR. These days, some researches focus on text documents because the documents can be widely distributed such as newspapers and magazines [5], [6], [7].

As another component for developing AR systems, a user interaction also becomes an important aspect to enable the user to manipulate the virtual contents inside the systems. Koike et al. developed an interactive system called EnhancedDesk manipulated by a finger pointing [8]. However, the environment of the system is limited such that the document should be parallel to the desk and include a fiducial marker to estimate the pose of the document.

To develop a constraint-free document AR system as shown in Figure 1 , we merge our natural feature based document tracking method [7] with a finger pointing method based on a skin color detection. Compared to EnhancedDesk [8], the constraints are relaxed such that the user can select any physical documents without a fiducial marker and hold the document freely. One benefit of our system is the possibility to extend it to many kinds of the applications because we can use various machine-printed documents such as user manuals, textbooks, pamphlets, or academic papers. We can overlay any information on the top of them. Therefore, our system has a potential to be a new interactive media.

MMSP'10, October 4-6, 2010, Saint-Malo, France. 978-1-4244-8112-5/10/$26.00

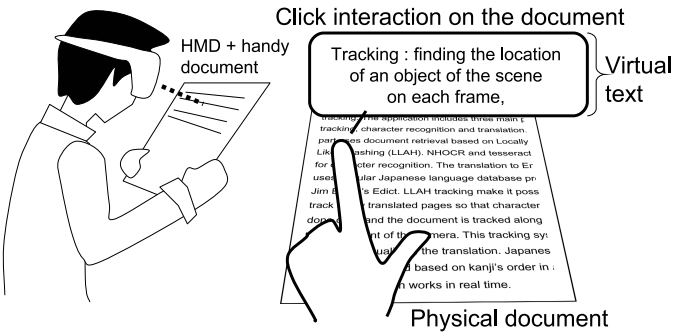


Fig. 1. System overview. The user touches one of the words on a text document to see the virtual contents of the word such as its translation.

The rest of the paper is organized as follows: first, we introduce some related works on augmented reality fields and user interaction techniques in Section II. Next, we describe our system in Section III, and the example of our system in Section IV. In Section V, we evaluate the accuracy of the clicking interaction and the performance of our document tracking.

II. Related Works

For overlaying virtual objects in a real world, we need to estimate a geometrical relationship between a captured image and a real world with a known coordinate system. To achieve this goal, a local keypoint descriptor-based approach such as SIFT [9] is commonly used. It is well-known as being robust against the occlusion. However, the computational cost of SIFT is not applicable for real-time AR systems. To solve this problem, Wagner et al. have reduced the process to compute SIFT by describing a keypoint with a smaller region. They applied their method to a mobile device and proved that it worked in real-time [10].

Usually, these local descriptors require rich textures to extract distinctive keypoints from the textures. As a consequence, these descriptors cannot be applied to text documents because the local regions of the text documents are similar

and not discriminative. To overcome this issue, other types of the descriptors for texts have been proposed by using local geometrical relationship of feature points [5], [6].

Hull et al. have proposed the horizontal connectivity of word lengths as a descriptor [5]. Nakai et al. have proposed another approach called locally likely arrangement hashing (LLAH) [11], which descriptor is composed by the combination of keypoint arrangement. LLAH is applied to several applications such as annotation extraction [6] and document mosaicing [12] because it works in real-time. We have modified LLAH in order to handle a large range of viewpoint by on-line learning of the descriptors [7].

For the user interaction using user fingers, Verdie discusses some common methods for touch interaction and their limitations in computer vision systems [13]. The example of the vision based systems is MirrorTrack that uses a mirror to acquire two views of a finger and its reflection to define a hand touch [14]. There are several types of a clicking method using a single camera such as movement delay [15], thumb appearance [16] and the color difference of a fingernail when a hand touch occurs [17].

The main contribution of this paper is to merge our natural feature based document tracking method [7] with a finger clicking interaction to develop the constraint-free clickable papers. Our system uses only normal text documents with a single camera. Other special equipments such as fiducial markers are not necessary. In addition, we introduce a handheld AR dictionary system as a practical application.

III. System Configuration

Our system utilizes a camera and a display such as a mobile phone or a head mounted display (HMD). A projector can be also applicable for the display. The user prepares any black and white printed text documents.

As a pre-processing step, the reference image of each user's document is registered into our system. The reference image can be generated from digital PDF documents or document images captured from a top view.

When the user begins to use our system, the user sets the camera pose close to the reference image for camera pose initialization. From next following frames, the user can move the camera to arbitrary viewpoints because our system is able to track the pose of the document.

While the document is being tracked, the user can interact with the document by selecting a word on the document as a query to our system. Selection process is done by clicking the word using a finger. Because we use a single camera, we track the movement of the finger to define a click. The flow of our system is illustrated in Figure 2.

A. Database Preparation

The virtual contents of augmentation are prepared and registered into a database as a pre-processing step. The example of the contents is the translation of each word into other languages. In order to overlay the translation of each word, the overlaid position on the document is also registered such as the upper part of the word. The user can register any kinds of contents such as word translations, pictures and movies linked with words on the document.

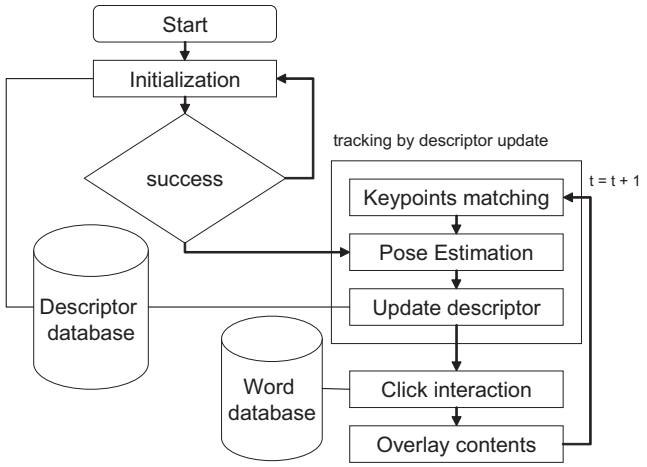


Fig. 2. Flow of our system. Our system contains two databases: keypoints database and word database for virtual contents. The process can be divided into three parts: initialization, tracking and interaction.

For camera pose estimation, the reference image of the user's document is registered into the database. We estimate the relative pose between a captured image and the reference image. The reference image can be generated from digital documents such as PDF files. In our example of application (Section IV), the user prepares the reference image by capturing the document from a top view. From the reference image, a collection of descriptors of keypoints is extracted and registered. The database structure can be referred on the our previous paper [7].

B. Initialization

Before tracking the camera pose, the initial camera pose is computed as the initialization step. In the database preparation, the top view image of a document is registered as a reference image. When the user begins to use our system, the user sets a camera pose close to the pose of the reference image. By matching several keypoints in a captured image with those in the reference image, the initial camera pose with respect to the reference is computed. After the initialization succeeded, the tracking of a camera pose starts.

C. Document Tracking

We use document tracking based on LLAH [7]. The method is called tracking by descriptor update. This method can handle detection and tracking on multiple text documents. Moreover this method is also robust against occlusion because non-occluded regions can be still matched effectively. This method is suitable for our system because occlusions by the user's hand occur during the clicking interaction.

To match two consecutive frames, we compute and match the descriptors of each keypoint. Because one keypoint has several descriptors in LLAH, keypoint matching is successfully achieved by selecting the maximum number of the same

descriptors at each keypoint even if the some of the descriptors are different due to the viewpoint change. By dealing with the different descriptors as new descriptors, we updated them for the matching with the next frame image.

D. Clicking Interaction

From our tracking method, word regions can be provided from word extraction because the method uses them for computing descriptors as shown in Figure 3. This can be utilized for realizing our application such that the user selects one of the words as a query by clicking.

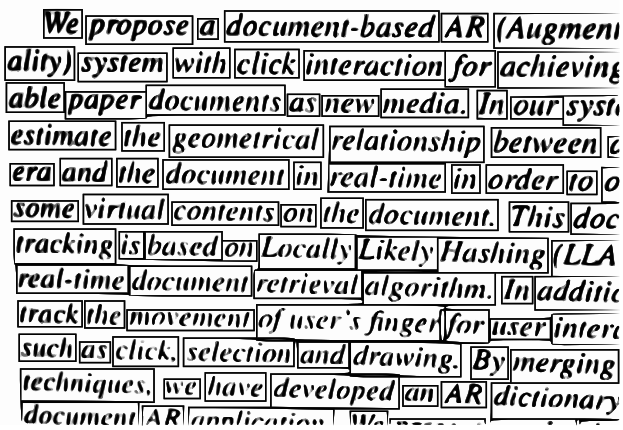


Fig. 3. Word regions. By applying an adaptive thresholding method to a captured document, the word regions can be automatically segmented. This segmented word is utilized for character recognition. In addition, the system can look up the meaning of each word on the word database.

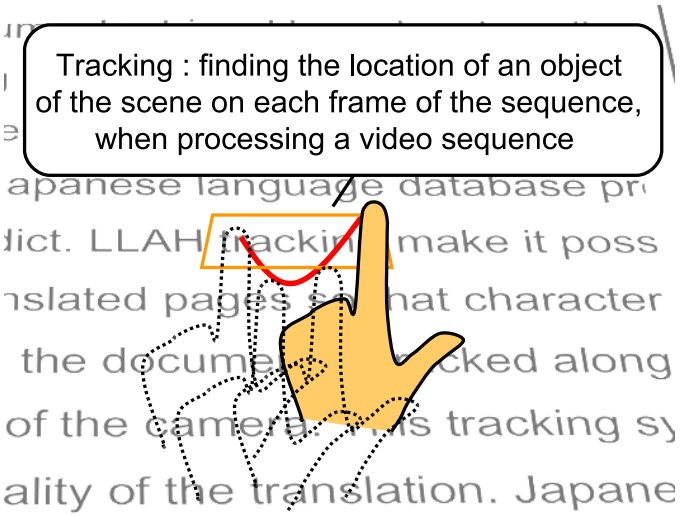
Clicking interaction is performed by a user's finger. The detection of the position of the user's finger is based on a simple color skin detection. First, we detect the area of skin in the captured frame. The detection is performed by scanning the whole image to create a skin mask image. Then, the pixel which has the intensity close to human skin is set to 255 and the other intensity is set to 0. We acquired binary image that resembles the area of human skin.

The binary image may contains noise if the object in the background or the environment contains similar color to human skin. We remove this noise and choose the hand area by computing the area of blobs in binary image. Because we assume the largest area captured in the image is the human skin, we choose the largest blobs in binary image as the hand area.

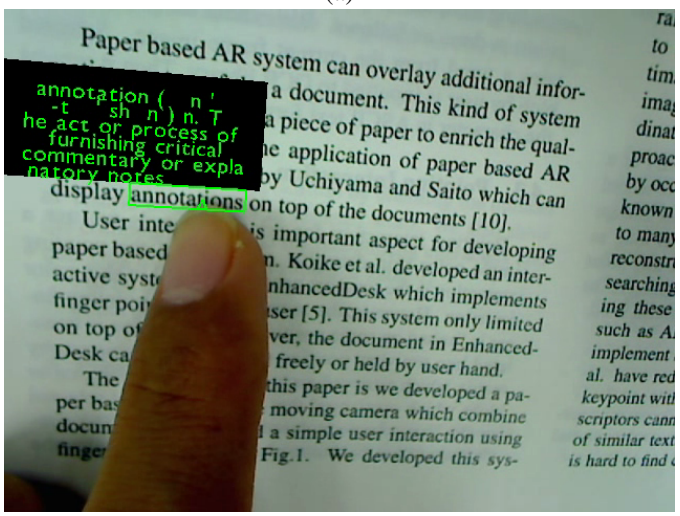
Next process is calculating the position of the fingertip. First, we calculate the center of gravity of hand area by using a contour moment. The hand has a fixed shape and the pointing finger is located farthest from the center of gravity of the shape. We traverse the edge of skin area to get the farthest pixel from the center of gravity. This farthest pixel is used as the fingertip position.

Since AR applications with a single camera depend on only visual appearance, we can not decide when a fingertip touch the document. Instead of utilizing the time when the fingertip touches the document, we use the fingertip trajectory for the click interaction. We define the clicking interaction as a movement of the fingertip that forms **v** shape as shown in Figure 4. When a fingertip moves in a certain angle in **v** pattern and a certain speed, it performs a similar gesture as mouse clicking gesture.

This method has a small computational cost because it only takes into account an angle calculation between two directions of the fingertip. Hence, we can keep our AR system to run in real-time.



(a)



(b)

Fig. 4. Clicking Interaction. (a) There are two options for performing clicking using a finger. First, the user moves the finger along **v** trajectory for clicking. Second, the use taps the document. This tapping action forms **v**. (b) If the clicking of a word is succeeded, the virtual content of the word is retrieved from the word database. In the picture, the user clicks a word "annotation" and the system translate the text and displays its meaning by looking up in the word database.

IV. Application

As the practical example of our system, we develop an AR dictionary system. Suppose there is a document that contains many difficult terms or printed in your non-native language.

By clicking a word, the user can read the overlaid meaning and explanation of each word on the document.

First, the user prepares a text document. It is ideal that the digital version of the document is available as a reference image. If the digital version of the document is not available, the user can capture the document from a nearly top view to register it. When the reference image is registered, we apply an Optical Character Recognition (OCR) method to the document to create the translation of each word as the overlaid contents. We have prepared two modes of the character recognition: English by Tesseract-OCR [18] and Japanese by NHOCR [19]. Before using our system, the user selects one of the mode. After OCR is performed, the translation of each word with its overlaid position is registered into our system.

This virtual contents registration can also be done in a parallel process. Therefore the application can run while the word database is prepared. This parallel database preparation is useful when a new text document is registered. The word database is updated dynamically.

Regarding Japanese documents, the word region cannot be correctly provided from the word extraction because there is no whitespace in its sentence. The whitespace is in fact used as the delimiter on the segmentation process.

By the absence of whitespaces, several characters are detected as one character. Therefore the user may use a highlight marker pen to select a word in a physical document as illustrated in Figure 5. When the user clicks the highlighted word, the meaning of the word is displayed.

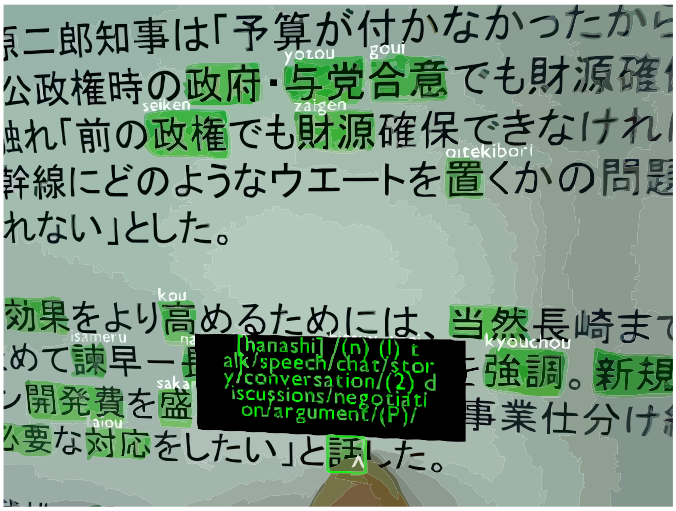


Fig. 5. Word selection for Japanese documents. Because the Japanese word regions cannot be extracted in the document tracking, the user highlights words in Japanese characters region for a query by using a physical marker pen. The system utilizes the NHOCR library to translates it into ASCII characters. The system then lookup the meaning in the word database and displays it.

V. Evaluation

We have tested our system using a desktop PC with specifications: Intel Quad core, 4GB RAM and 640×480 pixel camera. Our application is coded in C++ with OpenCV [20]. The optimized size of the characters in a document is selected beforehand.

We evaluated the accuracy of the clicking interaction in our system. Several test conditions were given by tilting or rotating document which is located 25 cm away from camera. We also placed document 16 cm and 40 cm away from camera without tilting camera. Each test takes 20 times clicking on different words. Clicking is correct if its position is inside a boundary of word that the user wants to select.

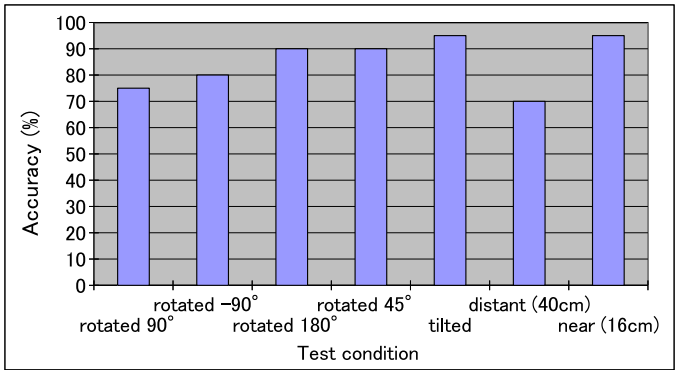


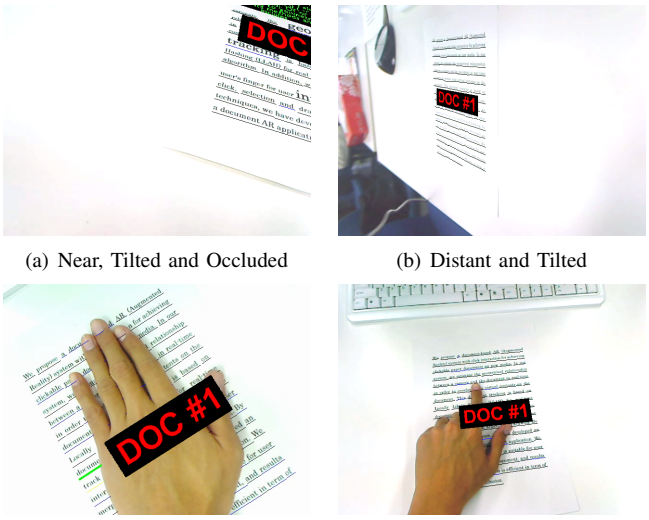
Fig. 6. Evaluation of clicking. The accuracy in each conditions are more than 70%. The accuracy is 85% in average. Most of the error occurs when the finger occludes the area of the selected word. Hence the click preciseness decreases.

The result of this experiment is shown in Figure 6. The accuracy of the clicking interaction is 85% in average. One important issue regarding clicking interaction is the preciseness. A fingertip in contrast with a touch pen pointer, it has a dull shape so that it is difficult to choose smaller target in the paper. When the document is placed 40 cm from the camera, the fingertip area is bigger than the boundary of selected word. In this case more than one word area are selected. Moreover the user can not see whether the right word is selected. As a result, the preciseness of clicking decreases. One common solution of this problem is making the selection object bigger than the size of fingertip.

From the results of clicking evaluation, we can conclude that clicking interaction using trajectory is effective against the change of camera pose.

We also tested the performance of document tracking respect to the influence of the occlusion, which was not much discussed in the tracking by descriptor update paper[7]. In this experiment, we tried to overlay virtual text ”DOC” followed by the document number. We placed the document in several conditions: tilted, occluded, 50 cm or 10 cm away from camera.

In our experiments, it is robust to camera position and angle and hand occlusion. Because the document tracking method works well, we can see overlaid text even the document is placed 10 cm away from camera and tilted as shown in Figure 7(a). The overlaid text is displayed properly when the document is located 50 cm away from the camera and tilted (Figure 7(b)) or occluded by hand (Figure 7(c) and 7(d)).



(a) Near, Tilted and Occluded (b) Distant and Tilted

(c) Hand Occluded (d) Distant and hand occluded

Fig. 7. Experiment Results. These results prove that our document tracking method is robust to occlusion and rotated cases. When the camera is located around 50cm from the document, the tracking method still works properly.

Thanks to the robustness to the occlusion, we can add a clicking interaction on the physical document. We can also add some user interface components such as a button and its action so that all of interaction can be embedded inside the physical document. Therefore in our system, the interaction with the keyboard and mouse is no longer necessary and it makes our system handy.

VI. Conclusion and Future Works

We presented a document-based Augmented Reality (AR) system with a clicking interaction for achieving a clickable document as a new tangible interface for a multimedia application. We estimated the pose of a document in real-time in order to overlay some virtual contents on top of a document using LLAH for real-time document retrieval algorithm. We use trajectory of fingertip for defining a clicking interaction. We performed experiments and showed some results that our document-based AR system with a trajectory clicking interaction is efficient in term of speed and robust to occlusions. We have developed an AR dictionary application that displays the meaning of word in a document and shown that our system can be a new interactive multimedia application.

In near future, by using our system, it is possible to build many AR applications that uses document instead of fiducial markers. Digital contents of multimedia application can be extracted and produced by using physical document. However, to implement and build such a general purpose system for every document is a challenging research. A physical document can be printed in color and it may contain images. Our current document tracking only handles a monochrome text document. Our next goal is handling color and image in a physical document.

Interaction is also a difficult issue. Problem regarding precision is necessary to be solved. The fingertip detection which takes into account human skin color will lead a problem due to skin color variation. Therefore we need to improve the fingertip detection and define the more accurate clicking interaction. Moreover a richer interaction is also necessary to increase the usability of the document AR system.

Acknowledgement

This work is supported in part by a Grant-in-Aid for the Global Center of Excellence for high-Level Global Cooperation for Leading-Edge Platform on Access Spaces from the Ministry of Education, Culture, Sport, Science, and Technology in Japan.

References

[1] H. Kato and M. Billinghurst, "Marker tracking and hmd calibration for a video-based augmented reality conferencing system," *Augmented Reality, International Workshop on*, vol. 0, p. 85, 1999.

[2] "ARToolkit," http://www.hitl.washington.edu/artoolkit/. [Online]. Available: http://www.hitl.washington.edu/artoolkit/

[3] O. Bergig, N. Hagbi, J. El-Sana, and M. Billinghurst, "In-place 3d sketching for authoring and augmenting mechanical systems," in *ISMAR '09: Proceedings of the 2009 8th IEEE International Symposium on Mixed and Augmented Reality*. Washington, DC, USA: IEEE Computer Society, 2009, pp. 87–94.

[4] C. Scherrer, J. Pilet, P. Fua, and V. Lepetit, "The haunted book," in *ISMAR '08: Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality*. Washington, DC, USA: IEEE Computer Society, 2008, pp. 163–164.

[5] J. Hull, B. Erol, J. Graham, Q. Ke, H. Kishi, J. Moraleda, and D. Van Olst, "Paper-based augmented reality," in *Proc. ICAT*, 2007, pp. 205–209.

[6] T. Nakai, M. Iwamura, and K. Kise, "Accuracy improvement and objective evaluation of annotation extraction from printed documents," in *Proc. DAS*, 2008, pp. 329–336.

[7] H. Uchiyama and H. Saito, "Augmenting text document by on-line learning of local arrangement of keypoints," in *Proc. ISMAR*, 2009, pp. 95–98.

[8] H. Koike, Y. Sato, and Y. Kobayashi, "Integrating paper and digital information on enhanceddesk: a method for realtime finger tracking on an augmented desk system," *TOCHI*, vol. 8, pp. 307–322, 2001.

[9] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *IJCV*, vol. 60, pp. 91–110, 2004.

[10] D. Wagner, G. Reitmayr, A. Mulloni, T. Drummond, and D. Schmalstieg, "Pose tracking from natural features on mobile phones," in *Proc. ISMAR*, 2008, pp. 125–134.

[11] T. Nakai, K. Kise, and M. Iwamura, "Camera based document image retrieval with more time and memory efficient LLAH," in *Proc. CBDAR*, 2007, pp. 21–28.

[12] ——, "Camera-based document image mosaicing using LLAH," in *Proc. SPIE*, 2009.

[13] Y. Verdie, "Evolution of hand tracking algorithms to mirrortrack," Vision Interfaces and Systems Laboratory, Tech. Rep., November 2008. [Online]. Available: http://vislab.cs.vt.edu/ vislab/wiki/images/2/26/ReviewMirrorTrack.pdf

[14] P.-K. Chung, B. Fang, R. W. Ehrich, and F. Quek, "Mirrortrack," in *Proc. AIPR*, 2008, pp. 1–5.

[15] A. Sanghi, H. Arora, K. Gupta, and V. B. Vats, "A fingertip detection and tracking system as a virtual mouse, a signature input device and an application selector," in *Proc. ICCDCS*, 2008, pp. 1–4.

[16] S. Kumar and J. Segen, "Gesture based 3d man-machine interaction using a single camera," in *Proc. ICMCS*, 1999, p. 9630.

[17] N. Sugita, D. Iwai, and K. Sato, "Touch sensing by image analysis of fingernail," in *Proc. of SICE*, 2008, pp. 1520–1525.

[18] Tesseract-OCR, http://code.google.com/p/tesseract-ocr/. [Online]. Available: http://code.google.com/p/tesseract-ocr/

[19] NHOCR, http://code.google.com/p/nhocr/. [Online]. Available: http://code.google.com/p/nhocr/

[20] "OpenCV," http://sourceforge.net/projects/opencvlibrary/. [Online]. Available: http://sourceforge.net/projects/opencvlibrary/