Tracking Hands and Objects for an Intelligent Video Production System

Motoyuki OZEKI Masatsugu ITOH Yuichi NAKAMURA Yuichi OHTA
IEMS, University of Tsukuba, 305-8573, Japan
E-mail: ozeki@image.esys.tsukuba.ac.jp

Abstract

We propose a novel method for detecting hands and hand-held objects in desktop manipulation situations. In order to achieve robust tracking under few constraints, we use multiple image sensors, that is, a RGB camera, a stereo camera, and an IR camera. By using these sensors, our system realized robust tracking without the prior knowledge of an object even if there are moving people or objects in the background. We experimentally verified the performance of object tracking by each of the three sensors and evaluated the effectiveness of their integration.

1. Introduction

With the recent progress of multimedia technology, videos and video-based multimedia are widely used for various purposes. There are now great demands for video-based contents, and their indexing, retrieval, and query answering mechanisms. As one effective approach for this purpose, we have been constructing an *intelligent video production system*[3], which automates video capturing, editing, and indexing. The target of our system is to aid in the production of teaching/operating/instruction manuals. Automating video production will greatly contribute to school education, professional training, and lifelong education.

For such videos, objects which appear in a scene usually have important roles. For example, parts are most important when we assemble a machine. Thus, automatic object recognition and tracking can add useful video indices, and the detection results can be used for creating clickable icons.

For this purpose, we propose a robust method for tracking objects which appear in desktop manipulations. For robust object tracking under few constraints, we use multiple image sensors, that is, a RGB camera, a stereo camera, and an IR (infrared) camera. With this method, our system tracks a hand-held object at video-rate even if there are other moving objects or people with skin color tones in the background.

Figure 1. Typical situation

2. Conditions of Object Tracking

The purpose of our system is to capture desktop manipulations, to edit, and to index the obtained videos. One typical situation that we apply our system is as follows:

1. A person holds or points an object in front of his/her body as shown in Figure 1.
2. Referring to the object, the person mentions its name or how to use it; *e.g.*, “This adapter has a connector to attach a monitor cable”.
3. The person then assembles some parts or demonstrates how the object works.

In such a case, it is difficult to provide the complete appearance of an object, since appearance can be easily altered by rotation, deformation, or assembling; *e.g.*, joining parts. Moreover, it is natural that the background may change because of the inclusion of another person or moving object.

Thus, we consider object tracking under the following conditions:

- The system has no prior knowledge about object’s size, color, texture, and so on.
- The background may change at any time during manipulation.

On the other hand, we can naturally assume the following restrictions:

- Most of the important objects are moved or manipulated by human hands.
- The space (volume) in which important objects potentially appear is known on the condition that the work bench is stationary.

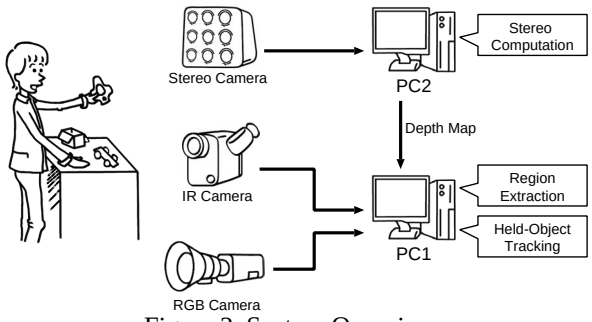


Figure 2. System Overview

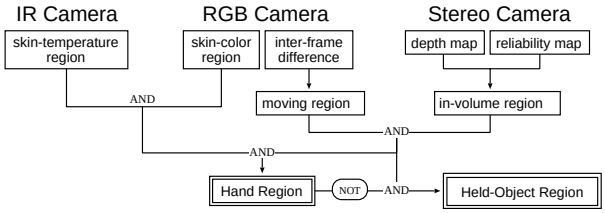


Figure 3. Region Detection and Integration

Even when considering the above two restrictions, the above conditions are still severe. Although a number of studies have been performed regarding hand tracking or object tracking, some of which reported good results, our situation is much more difficult than the situations assumed in those studies. Object rotation or occlusion caused by grasping can easily alter the object's texture, and people moving in the background add serious noise that cannot be easily eliminated.

Thus, we propose a new method for object tracking that utilized multiple image sensors: an ordinary RGB camera, an IR camera, and a stereo camera.

3. Using Multiple Image Sensors

The overview of our system is shown in Figure 2. The system detects the following regions based on information provided by three sensors:

RGB Camera: the *skin-color* regions and the *moving* regions.

Infrared Camera: the *skin-temperature* regions, which are regions with an intensity corresponding to skin temperature, *i.e.*, around 34°C.

Stereo Camera: the *in-volume* regions, which are regions in the volume in which hands and related objects appear.

By integrating the above regions, the *hand* regions and the *held-object* region are detected based on the following principle.

$$\text{hand region} = \text{in-volume region} \wedge \text{moving region} \wedge \text{skin-temperature region} \wedge \text{skin-color region} \quad (1)$$

$$\text{held-object region} = \text{in-volume region} \wedge \text{moving region} \wedge \neg\text{hand region} \quad (2)$$

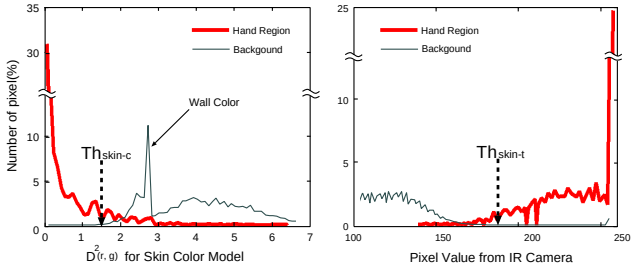


Figure 4. $D^2(r, g)$ for skin color model (left) and Pixel values from IR camera (right)

Figure 3 shows the outline of the above process. Once a held-object region is extracted, we can register the texture, *i.e.*, the appearance, of the object. This texture can be used for detecting the object after the person releases it.

4. Process for Each Image Sensor

Before describing the whole system, we discuss problems concerning model construction, parameter tuning for each sensor, and sensor integration.

Process for RGB Camera

We created a skin color model by gathering the statistics regarding pixels showing skin color, and determined their distribution parameters. This method is based on Kondou's research [1], which determined that the distribution of Japanese face color taken from TV broadcasts is compactly modeled on the rg-plane[1].

First, the skin color regions are manually extracted, and the mean value and the covariance matrix Σ are calculated. Their actual values are as follows:

$$mean(\bar{r}, \bar{g}) = (0.437773, 0.334845)$$

$$\Sigma = \begin{pmatrix} 0.003915 & -0.000230 \\ -0.000230 & 0.000935 \end{pmatrix}$$

Square of Maharanobis distance $D^2(r, g)$ from skin color is calculated, and from this the skin-color region is extracted.

$$D^2(r, g) = \begin{pmatrix} r - \bar{r} \\ g - \bar{g} \end{pmatrix}^T \Sigma^{-1} \begin{pmatrix} r - \bar{r} \\ g - \bar{g} \end{pmatrix}$$

The graph on the left in Figure 4 shows the statistics obtained from a typical image in our environment. $D^2(r, g)$ values in real skin regions and those in the background are plotted. Considering those statistics, we determined a threshold value of $Th_{\text{skin-c}} = 1.5$.

Process for Infrared Camera

Our IR camera captures infrared light with a wavelength between 7 and 14μm, which covers the dominant wavelength that a human body emits. We checked the pixel

[1] A normalized color space. $r \equiv \frac{R}{R+G+B}$, $g \equiv \frac{G}{R+G+B}$.

values in the real hand region and those in a typical background, and determined the threshold for extracting the skin-temperature region.

In our experiments, a threshold $Th_{\text{skin-t}}$ of around 180 well separates those regions as shown by the graph on the right in Figure 4. Since the actual pixel value depends on iris, focus, and other camera parameters, the threshold must be adjusted if those parameters are changed.

Process for Stereo Camera

As mentioned above, we assume that the space (volume) where hands and related objects appear is known to the system. In our experiments, we assumed that the width, height, and depth of the volume are 2.5m, 2m, and, 0.5m, respectively. These can be changed according to the spatial arrangement of the workspace and the camera position.

Objects in this volume can be detected by using the depth map obtained by the stereo camera. A problem in this step concerns the noise caused by homogeneous regions, periodic textures, and occlusion. In order to solve this problem, the reliability map provided by the stereo camera is used. The sharpness of the peak in disparity computation is evaluated, and a sharp peak generates a larger value in the reliability map[4].

For each pixel, the system uses the depth value only if its reliability is higher than the threshold. This simple operation works well for typical indoor scenes.

5. Integration for Multiple Image Sensors

Prior to actual region extraction and tracking, we need geometric compensation and synchronization among three images.

Geometric Compensation

A calibration board is placed on the worktable. Markers, which are visible from all cameras, are attached to the board. Based on the markers' locations, the projection parameters which map the IR image or the depth map to the RGB image are computed using the following quadratic model.

$$\begin{pmatrix} x_1 & \dots & x_n \\ y_1 & \dots & y_n \\ 1 & \dots & 1 \end{pmatrix} = M_{3\times 5} \begin{pmatrix} u_1 & \dots & u_n \\ {u_1}^2 & \dots & {u_n}^2 \\ v_1 & \dots & v_n \\ {v_1}^2 & \dots & {v_n}^2 \\ 1 & \dots & 1 \end{pmatrix}$$

where $(x_i,\ y_i)$ represents the marker position in the RGB image, and (u_i, v_i) is the marker position in the IR image or in the depth map. Although the IR camera has heavy radial distortion, 25 markers are sufficient to calculate the above parameters.

Synchronization

As shown in Figure 2, images from the RGB camera and images from the IR camera are captured into PC1, and images from the stereo camera are captured by PC2. The depth map images obtained at PC2 are transmitted to PC1 through the ethernet, and the region extraction process and the integration process are executed at PC1. To compensate for the latency of stereo computation and transmission time, the captured time is attached to each image. The depth map image captured at the nearest time is used with the other two images.

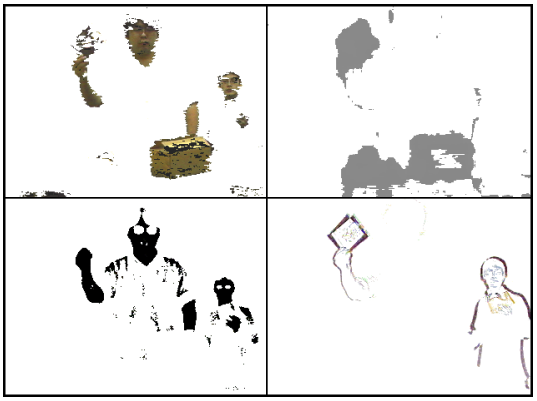


Figure 5. Detected regions (upper-left: skin-color region, lower-left: skin-temperature region, upper-right: in-volume region, lower-right: moving region)

Region Detection and Tracking

As shown in Figure 3, the hand regions are detected by taking logical AND operation of the four regions as shown in equation 1. The extracted hand region candidates are labeled after region expansion-contraction. Then, at most two regions whose areas are larger than the threshold are registered as hand regions. The held-object region is detected by equation 2. Through the position smoothing by the Kalman filter, the final position of the object is estimated. By repeating the above process at video rate, the estimated position of a held object is obtained in every frame.

The detected regions are shown in Figure 5, and examples of the tracking results are shown in Figure 6. As we can see in these figures, the held object is well detected and tracked even when the intermediate result by each sensor contains much noise. The skin-color region detection that is often used for detecting hands is not satisfactory as we can see in Figure 5. By combining the depth and temperature information, we can easily eliminate the region of the skin-color box and the region of a moving person in the background.

6. Experimental Results

Tracking Performance

The specifications of the three image sensors and PCs are shown in Table1. As shown in Figure 6, we evaluated our system in two situations. Scene A is a simple scene in which one person is holding and moving an object. Scene B is a more complicated scene containing multiple objects on the worktable and with another person walking behind.

Table 1. Image sensors and PCs

PC	CPU	RAM	OS
PC1	P4 1.5GHz	RDRAM 256MB	Linux kernel2.4
PC2	P3 933MHz	SDRAM 256MB	Linux kernel2.2
Intel C++ Compiler for Linux ver5.01			
Sensor	Name	Output image	Vendor
RGB	DXC-9000	320x240 30Hz	Sony
IR	D-EYE10	320x240 30Hz	Nippon Avionics
Stereo	FZ-930	280x200 30Hz	Komatsu

Table 2. Detection and tracking performance

	#Total	#Detection failure	#Tracking failure
Scene A	1350 frames	30 frames (2.2%)	4 frames (0.3%)
Scene B	1350 frames	11 frames (0.8%)	80 frames (5.9%)

Regions were correctly detected for 97% and 93% of the frames in scene A and scene B, respectively. For scene B, tracking failed to a slightly greater degree than in scene A, since the box with skin color and the walking person create misleading regions. However, the rate of tracking failure is still less than 6%, which is difficult to achieve using a single image sensor.

Application Example

We demonstrated one promising application of our system. By combining object tracking with the intelligent video production system[3], we can create a clickable icon signifying a detected object, and link it to information such as the movie clip. Our prototype system does this simultaneously in taking videos: the system directs the cameras to capture a held object; when the person gives any explanation of the object, the system registers its appearance and links it to the movie clip being captured.

An example is shown in Figure 7. In this scene, the person is giving an explanation of a dish for the guest. First, when he held the dish, the system detected it, and the rectangle with the dotted lines shows the location. When he gave the explanation of the dish by speaking "This dish is ....", the system recognized the situation[2], and registered his annotation as information regarding dishes. This step is noted by the red thick lines overlaid on the dotted lines. When the person put the object on the table, the texture and the position of the object were registered, and the captured annotation was linked to the object region.

7. Conclusion

We proposed a novel method for detecting hands and objects held by hands in desktop manipulation situations. By using multiple image sensors, our system realized robust tracking without the system's prior knowledge of an object even if there are moving people or objects in the background or if there are other skin-color regions. We experimentally verified the performance of object tracking using three sensors and evaluated the effectiveness of the sensors' integration.

Our future research will focus on tracking smaller objects, efficient tracking by motion prediction, utilization for video-based multimedia production.

[2]Please refer to [2] for the detection.

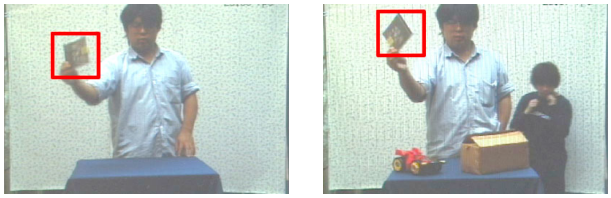
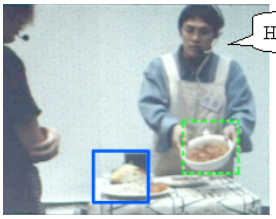


Figure 6. Scene A (left) and Scene B (right)

(a) Left image shows the result of held-object tracking. The person on the right is explaining a dish for a guest.

(b) When he held the dish, the system detected it, and the rectangle with the dotted lines indicates the location.

(c) When he gave the explanation of the dish, the system recognized the situation and registered his annotation as the dish information.



(d) When he put the object on the table, the texture and the position of the object were registered, and the captured annotation was linked to the object region. Then, the system detected a new held-object.

Figure 7. Example of combining our object tracking system with an intelligent video production system

References

[1] H. Koundou, et al. Indexing persons in news video by telop recognision and face matching (In Japanese). *Proc. IEICE Annual Conference*, D-12-190, 1999.

[2] M. Ozeki, Y. Nakamura, and Y. Ohta. An intelligent system for recording presentations (In Japanese). *Proc. 6th IIM*, pages 69–74, 2000.

[3] M. Ozeki, Y. Nakamura, and Y. Ohta. Camerawork for intelligent video production. *Proc. ICME*, pages 41–44, 2001.

[4] O. Yoshimi and H. Yamaguchi. Sharpening of object contours disparity image using coefficient of swelling (In Japanese). *Proc. SSII*, pages 227–230, 2000.