Vol.15 No.3, 2013

Original Paper

Development of a Head Gesture Interface for a Self-portrait Camera

Shaowei Chu*1 and Jiro Tanaka*1

Abstract – Most existing digital camera user interfaces place little emphasis on self-portrait options. Therefore, it is not always easy to take self-portraits using conventional user interfaces. This paper presents a vision-based head gesture interface for controlling a self-portrait camera that helps users to take self-portraits effectively and efficiently. Intuitive *nodding* and *head-shaking* gestures control the camera zoom in/out on the face, and a *mouth-opening* gesture triggers the camera to take a picture. We evaluated its usability factors (effectiveness, efficiency, and satisfaction) and compared it to a remote control in a user study. The results suggest that our interface is useful for taking self-portrait pictures.

Keywords : digital camera, human computer interaction, user interface, head gestures, image processing

1. Introduction

Taking personal self-portrait photographs has become increasingly popular with the widespread use of online social networking systems (SNS), such as Facebook and Twitter. Self-portraits are taken not only for fun but also to stimulate creativity, log our life, and present ourselves to society [1] [2].

However, there are many problems associated with taking a portrait shot. Conventional methods, such as use of a self-timer, can be tiresome (having to run back and forth to prepare and then pose for the shot), time consuming, and frustrating as many shots may be needed to obtain a satisfactory portrait [3] [4]. Use of a handheld remote control may be a better choice, but the additional device occupies the hand, which limits freedom in terms of the possible postures one can assume and often results in unnatural postures [3] [5]. Conventional interface designs pay only modest attention to user interaction, and largely do not consider user-friendly ways for taking self-portraits.

Therefore, we feel that it is important to develop a user-friendly interface that allows a lone individual to take portrait shots both effectively and efficiently.

Our previous studies have suggested that a vision-based gesture interface may be ideal choice for developing a remote control interface for a self-portrait camera [6] [7]. Such interfaces have a number of advantages: they provide remote control capability that allows a user to interact with a camera while posing in front of it; human gestures contain many meanings that can be mapped to various camera functions; and gesture recognition can be accomplished using an image-processing algorithm on camera live-view image sequences and thus there is no need to modify existing camera hardware.

In a previous study [6], we proposed hand gestures for interacting with cameras. This work attracted some level of interest from the public and the media. However, it is difficult to develop a zooming interface using hand gestures, as the user's hands may go outside the field of view of the camera, especially at high zoom values. Moreover, when using hand gestures a large display is needed to show the live view and the graphical user interface. This limits the portability of the system and obstructs its practical use.

Therefore, we proposed a second technique using head *nodding* and *shaking* gestures as the interface [7]. We found that head gestures work well with small displays and are suitable as the basis of a zooming interface, because the face is unlikely to move outside the field of view; as the focus and most important region in a self-portrait, it is always available for gestures, which can be mapped to camera control functions.

In this study, we enhanced the head gesture interface to develop a practical self-portrait camera, reintroducing the *nodding* and *head shaking* gestures and incorporating facial gestures such as the *mouth-opening* gesture. We conducted a formal user study to examine its usefulness (effectiveness, efficiency,

*1: Department of Computer Science, Graduate School of Systems and Information Engineering, University of Tsukuba

and satisfaction) for controlling a camera. The experiment also compared the proposed head gesture interface with a smartphone remote control. Two types of remote control, button and touch interfaces, were implemented for the comparison. Users were able to use the gesture interface to interact with a camera effectively and felt satisfied with the technique for taking self-portraits. In addition, the gesture interface had a slightly better satisfaction evaluation than the handheld remote control, particularly than touch interface.

A summary of this work's contributions follows.

- A vision-based head-motion gesture interface for cameras is presented. The interface maps *nodding* to zooming in, *head shaking* to zooming out, and a *mouth-opening* gesture to trigger the shutter.
- A prototype was developed and was evaluated by a dozen people who reported high levels of satisfaction.
- We conducted a formal experiment to evaluate the usability of the system compared to a handheld remote control for controlling a camera. Users felt a slightly higher degree of satisfaction with our proposed interface.

2. Related Work

2.1 Self-portrait approaches

There are several typical approaches for taking self-portraits.

The long arm. Most people have taken this kind of self-portrait, which involves holding the camera as far away as possible to take the photograph. This is a very popular technique with tourists. In fact, several camera manufacturers now produce cameras with two LCD screens (such as Samsung DualView camera), one of which is in front so that users using this technique can see themselves when taking self-portraits. Although this method can be fun and records a moment, such pictures are not ideal as they are generally of poor quality; they may distort the face, and it is difficult to keep the hand steady [5].

In contrast, our proposed system can be used on any flat and stable surface. The user can stand a distance from the camera, relax, and perform intuitive gestures to interact with the camera.

The self-timer. Almost all cameras now have a self-timer option. In general, a tripod or a steady surface such as a shelf or table is required to hold the camera. When preparing for the shot typically entails positioning the camera while behind it, perhaps using placeholders such as a potted plant or other object as stand-ins for the user, pressing the button, quickly getting into position, and then waiting for the picture to be taken. This method is time consuming and can be frustrating, i.e., ensuring the correct focus and good composition, as it may require many attempts to obtain a satisfactory portrait [3] [4]. In addition, the user may find it difficult to find the appropriate zoom and then take a picture while (s)he keeps touching the camera to make zooming adjustments.

A better choice would be for the user to be able to interact with the camera remotely while in front of the lens, as in our prototype.

Remote control. Using a remote shutter release controller, it is possible to pose and click without having to run back and forth to the camera. The camera will also be able to focus correctly as the user will already be in front of the lens [5]. In addition, the smartphone remote control may provide many more function controls, such as zoom, parameter settings and live preview. However, if the hands are going to be in the shot, it will be necessary to conceal the handheld remote control [3] [5]. Indeed, the additional device occupying the hand often results in unnatural postures in final photographs.

Our proposed gesture interface can provide remote control without requiring any devices to be held. The user is able to perform intuitive head gestures to trigger camera functions.

Vision-based gesture interface. This technique involves the application of complex image-processing algorithms to the camera live view image sequence and the detection of specific features, actions, and human gestures as control commands to interact with the camera. Successful applications, such as Sony Camera Robot [8], use face-detection techniques to locate people and take pictures automatically. The Casio Motion Shutter [9] also enables users to take pictures using hand motions. Chu [6] proposed using hand gestures to control pan and tilt for composition of the camera view frame.

Our proposed system also uses a vision-based gesture interface. We have implemented two important control functions: zooming and triggering the shut-

ter. We mapped *nodding* and *head shaking* gestures to zooming in and out, respectively. In addition, the *mouth-opening* gesture is used to activate the camera shutter trigger.

2.2 Head gesture interface

The vision-based head gesture interface has been studied by several researchers [10] – [18]. One typical gesture recognition scheme presented by Davis [10] uses feature tracking on face region to estimate head motions, and then utilizes a Finite State Machine (FSM) to recognize *nodding* and *shaking* gestures. The author used the gestures for a simple dialog-box agent and acquired yes/no acknowledgement from the user. However, it needed a hardware equipment, IBM PupilCam, to detect the face, thus limiting its practical application. Li [11] modified the recognition scheme and introduced the block motion vectors to perform motion estimation, which reduced computational requirements and enabled the use of a single camera to run on mobile devices. However, it required the user to place his/her head at predefined positions in order to perform the gestures. Tan [12] introduced the use of coordinates of eyes to judge the direction of face's movement. A hidden Markov model (HMM) was trained to perform *head nodding* and *shaking* recognition. The limitation reported by the author is that, by using eye locating method, it was easy to generate wrong results when the user moved his/her head in one direction continuously. Other proposals such as the use of stereo camera [13], or context knowledge [14] to improve the recognition were also reported.

On the other hand, facial gestures, such as mouth-opening and eyebrow-raising, that work together with head gestures, were also studied and proved to be an effective way to trigger commands [16] [17] [18].

Although the head gestures and facial gestures were widely studied, few of the studies addressed the zoom interface by using head gestures. In this work, we present an intuitive function mapping of zoom. Because humans can easily perform the *nodding* and *shaking* gestures continuously, it is appropriate to map them to zooming in/out respectively. This was verified in our user study, as described in Section 5.

We also improved the gesture recognition reported in [10] [11]. We use a safe zone around the face to perform auto-initialization and quickly exclude casual head movements. A simplified motion estimation by using optical-flow for "good features to track" [19] on face is used to achieve good performance and motion tracking accuracy. These improvements offer an effective head gesture interface.

3. Design Goal

Our design goal was to develop a vision-based head-gesture interface for controlling a digital camera to take self-portrait pictures effectively and efficiently. Considering the feedback from previous studies and surveys, we present a summary of the design goals of the current system.

Small size of frontal screen - Unlike conventional vision-based interfaces, which usually have a large display to provide visual feedback, we took mobility into consideration to design a system that would work well with a small screen. Currently, we use a 3.5-inch viewfinder and have introduced several strong gesture patterns that work well with little visual feedback.

Strong motion gesture patterns - Portrait shots do not capture motion, but rather focus on a static posture. Therefore, it is straightforward to use motion gestures as control commands. The gestures should have strong motion patterns that cannot easily be triggered by accident. Moreover, it is better to use small motions that do not disturb the user's eye contact with the lens when performing gestures. Thus, the user can still observe a preview of his/her posture and confirm the status of the camera.

Head gestures only - We believe that head gestures are the best choice for mapping to the zooming function of the camera. To maintain consistency, we did not combine head gestures with hand or body gestures.

Real-time processing - The image-processing procedure for detecting gestures should be fast (at least 30 frames per second, FPS) to guarantee that the system shows smooth video on the viewfinder and so the user can obtain instant feedback from the camera while performing gestures.

We will discuss the validity of our design in the Discussion section on the basis of the results of the user study.

4. Proposed Gesture Interface

The major innovation of our work is that the self-portrait camera is responsive to head and mouth

Fig. 1 Prototype of our self-portrait system. A Canon 60D camera is used to take pictures and an iPhone is used as a viewfinder.

motion gestures when the user is in front of the lens. When the user faces the camera, the camera first detects the user's face and then tracks *nodding*, *head shaking*, and *mouth-opening* gestures. The user can perform these gestures to zoom in, zoom out, and trigger the shutter, respectively, to take a self-portrait photograph.

4.1 Overview

The prototype system (**Fig. 1**) consists of a professional digital single-lens reflex (DSLR) camera, an iPhone, and a personal computer (PC, not shown in the figure). The DSLR camera is used to take pictures, and the iPhone is used as a frontal screen to enable users to see themselves in the live view video. A specific type of tripod or professional tripod can be used to hold the camera. In our current prototype system, a PC is connected to the camera via a USB cable to exchange data with the camera, processing images to detect gestures, and sending previews to the iPhone through a WiFi connection.

To interact with the system, the user faces the camera and performs the head and/or mouth gestures. The face and detected gestures will appear on the live view iPhone screen as a visual aid.

4.2 Introduced interaction gestures

We examined many possible head and facial gestures to identify intuitive motion gestures that could be mapped to two important camera functions, i.e., zooming and shutter trigger. We decided on motion gestures, as static head poses or facial expressions may be easily confused with a user's portrait postures and expressions. In addition, when a user is making a static pose for a photograph, motions can be easily distinguished as command functions. Another important consideration is that the gestures should not be easily triggered by accident. When preparing for a shot, the user may try different creative postures and expressions with lots of movement; therefore, the gestures should be markedly different from whimsical or general movements of the head. In addition, the gestures mapped to camera functions should be intuitive.

Based on the above considerations, we chose *nodding*, *head shaking*, and *mouth-opening* as candidate gestures for controlling camera functions. All three gestures are intuitive, induce little fatigue and can be easily remembered by users [13] [17] [18]. In our earlier implementation, we attempted to detect an *eyebrow-raising* gesture, which has also been discussed previously [16] [18]. However, in a user study, we found that this gesture cannot be recognized well in users with long bangs, especially women, and we therefore abandoned it. We also considered *clockwise/counterclockwise head-tilting* gestures, but experiments showed that it was impractical, as users easily lost eye contact with the screen when performing it, and it was difficult to recognize these gestures. Hence, we also abandoned this gesture.

In the following subsections, we describe our gesture-recognition technique using *nodding*, *head shaking*, and *mouth-opening* gestures.

4.3 Gesture-recognition technique

The proposed gesture-recognition technique is based on face-detection methods that have achieved great success [20]. We use face detection to detect the face region in each frame of the video. To recognize *nodding* and *head shaking* gestures, we first present a safe zone to restrict head motions to exclude unnecessary motions (**Fig. 2**), and apply Lucas-Kanade optical flow tracking [21] to determine the two-dimensional (2D) motions of the head. To recognize the *mouth-opening* gesture, we first conduct empirical estimation to locate the mouth region of the face. The recognition of gestures is made under the assumption that a user's face is directly in front of the camera.

4.3.1 *Nodding* and *head shaking* recognition

To recognize the head gestures, we first define a safe zone as mentioned above. This zone, which is 40% larger than the face region, is initialized based on the face region in the image. Then, in the follow-

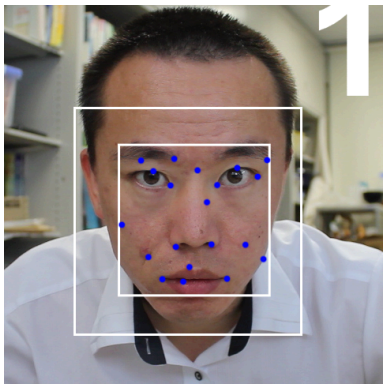


Fig. 2 Face region (inside rectangle), safe zone (outer rectangle), and the feature points extracted are marked with filled circles.

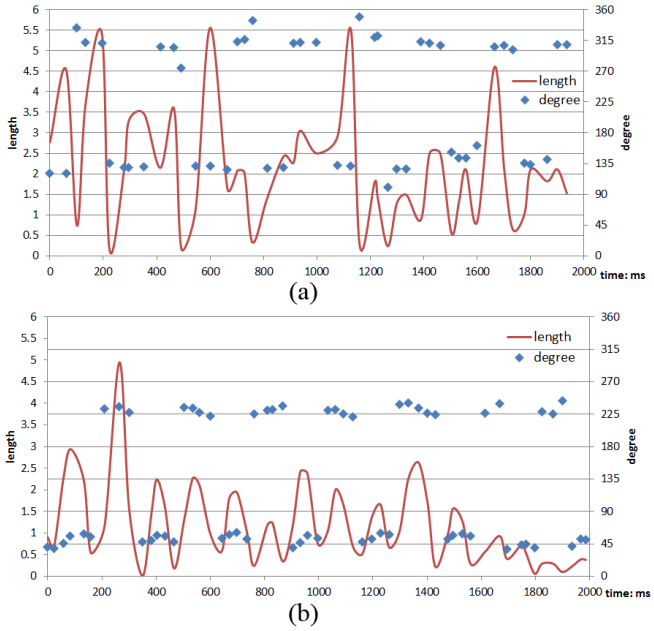


Fig. 3 Nodding (a) and head shaking (b) gesture data.

ing frames, we check whether the face has moved out of the safe zone or not. If so, the zone is reinitialized and a new check will start; if not, the zone remains as it is and the system waits for the next frame and face region to check the head motion. When the face does not move out of the safe zone within a specific period (currently 500 ms), then it is assumed that the head is still and that it is a good time to recognize head gestures.

Within the face region, we extract the image features for tracking head motion. Several feature-extraction algorithms have been reported previously [22] [23], but we chose a fast extraction method derived from the Hessian matrix, and selected the top 30 feature points as good features to track, as defined by Shi and Tomasi (S-T) [19]. After feature extraction (see **Fig. 2**), the Lucas-Kanade [21] method optical flow measurement is conducted for tracking the motion of each feature point. This is one of most precise methods for tracking Shi and Tomasi features in images. The optical flow measurements are feature points in the current frame displacement from the previous frame. The length and direction of motion of each feature point can be determined by calculating each feature's displacement. We calculate the set of feature points within the face region, and calculate the mean length (speed) and direction of movement as the main parameters in each frame.

The general motion data patterns of *nodding* and *head shaking* are shown in **Fig. 3**. These data were collected and recorded from one female graduate student. Important statistical information can be obtained from these data. First, in the *nodding* gesture (**Fig. 3a**), the motion directions are between 90° and 180° during tilting of the head down, and 270° and 360° during tilting of the head up. In contrast, the directions of motion in the head shaking gesture (**Fig. 3b**) are smooth and steady along the 45° and 225° lines. Second, the motion length data change periodically during the head gestures, but remain below a peak value of 6. The time intervals of each action switch, i.e., up-down and left-right movements, were calculated as 122.2 ms and 137.5 ms, respectively. However, we found in a number of experiments that these intervals varied depending on the user. To make the recognition less restricted, we ultimately chose an interval threshold of 500 ms in our application.

Based on these data, we concluded that the *shaking* gesture is a steadier motion than the *nodding* gesture. Hence, we separated the motion region of moving recognition into four regions: right, 10-80° (70° span); left, 190-260° (70° span); up, 260-360° and 0-10° (110° span); and down, 80-190° (110° span). The length of motion must be larger than 0.5 and less than 6.0 in the recognition process.

The timing-based finite state machine (FSM) shown in **Fig. 4** was used to recognize the gestures. The figure shows an example of *head shaking* gesture recognition with a transition chart with two main states: Stationary 1 as the motionless state and Safe

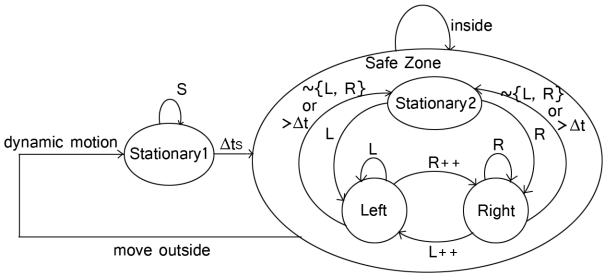


Fig. 4 Finite state machine for recognizing the head shaking gesture.

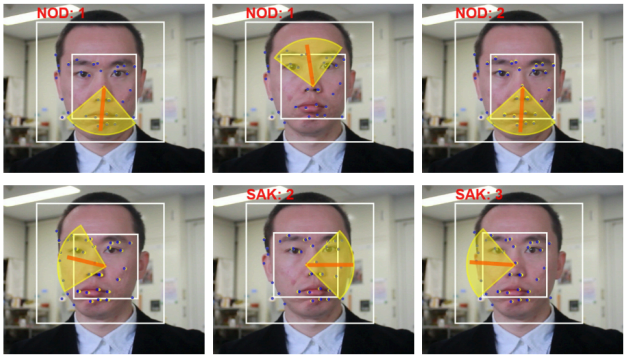


Fig. 5 *Nodding* and *shaking gesture* motions. Images in the top row show two nods; those in the bottom row show three shakes. The fan and the line at middle of the face indicates the direction of motion.

Zone when the face is inside the safe zone within a certain period. The Safe Zone state includes Stationary 2, left motion, and right motion. The transition from left to right or reverse transition adds one factor to the *shaking* count.

Fig. 5 shows an example of the gesture-recognition procedure in image sequences. In each image, the fan shape and the segment line at the middle of the face indicate the direction of motion. The count of nods and shakes recognized in each frame is displayed at the top left of the safe zone ("NOD" represents *nodding*, "SAK" represents *shaking*). The user can see the recognition states while performing gestures.

4.3.2 Recognition of the *mouth-opening* gesture

To recognize the mouth-opening gesture, we first estimated the mouth region related to the rectangle containing the face (**Fig. 6**). Two subjects, one female and the other male, participated in the user study to examine the mouth region related to the face rectangle. We considered both closed and opened mouths, and tested the final recognition accuracy of the *mouth-opening* gesture. We concluded that based on a rectangle using the distances from the eyebrows down to the lower lip and between the outside edges of the two eyes, the mouth width was 50% of the face width, and the height was 50% of the face height.

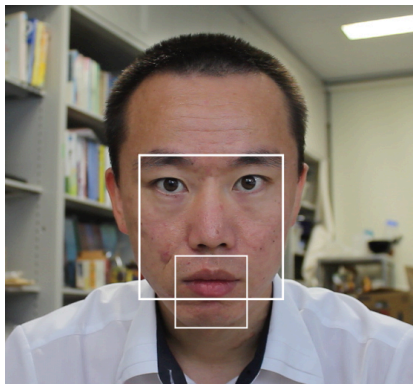


Fig. 6 Estimated mouth region.

In the second step, we manually arranged a matrix of 6×6 tracking points inside the region of the mouth, and applied Lucas-Kanade [21] optical flow to track the points' motions in each video frame (**Fig. 7**). We did not apply dense Horn-Schunck [24] optical flow tracking as used in a previous study [16], as it has low performance. We used a history-recording and accumulation method to analyze the motion data, similar to the previous study [16]. Two 6×6 historical matrixes accompany mouth motions. One detects opening motions, defined as the Down Matrix, which records motions in the downward direction. The other detects closing motions, defined as the Up Matrix, which records motions in the upward direction. The two matrixes are shown together with the mouth image in **Fig. 7**, with the Up Matrix on top and the Down Matrix on the bottom of the mouth image. The strength of detected motions is represented as dark blocks. The images from left to right represent three key frames of the mouth-opening gesture. The motion data are accumulated within 500 ms. When the accumulated data reach the defined threshold, a mouth-opening gesture can be detected.

The strength of matrix data is calculated in each image frame, and the pattern can be seen in the graph at the bottom of the figure. We concluded that when the mouth is opened, the Down Matrix peaks, and when it is closed, the Up Matrix peaks.

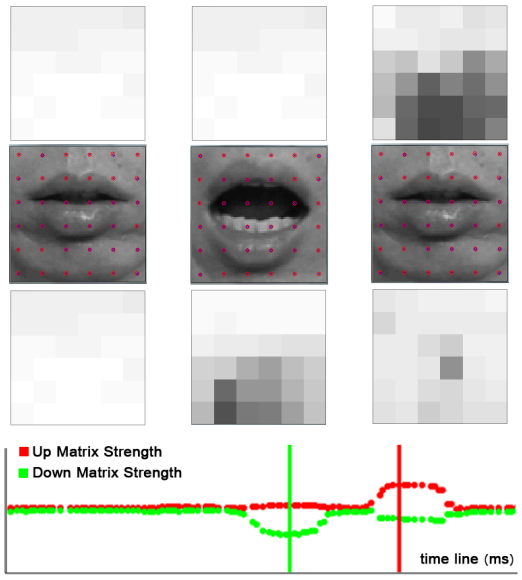


Fig. 7 The *mouth-opening* gesture. The Up Matrix and Down Matrix are shown at the top and bottom of the mouth image, respectively. The graph shows the data strength of the two historical matrixes.

Thus, the *mouth-opening* gesture can be recognized when such data patterns are detected.

4.4 Function mappings

In our previous work [7], we used *continuous nodding* and *shaking* for zooming in and out, respectively. We confirmed that these gestures are intuitive to users. However, the design of a *double nodding* gesture, in which the head nods twice and stops to trigger the shutter can easily be confused with *continuous nodding*. Therefore, in the present study, we mapped the *mouth-opening* gesture to the shutter trigger function instead. This gesture initiates autofocus on the user's face, and triggers the shutter timer to start countdown from 5 to 1, after which the camera takes the picture. If *head shaking* is performed during the countdown, it cancels it.

Regardless of how fast the user performs the gestures, zooming is set at a constant speed of 0.75X per second. We abandoned mapping faster speed gestures to faster zooming, as it made the users nervous and resulted in loss of accuracy.

4.5 Implementation

In the current implementation, we used a professional DSLR camera (Canon 60D) with an 18-55 mm lens. The camera offers a software development kit [25] for developers, including embedded hardware support for face detection. It runs at 30 FPS with 1056×704 video image sequences for live view, and can detect faces at sizes from 58×58 to 513×513 in the image frame.

We developed an iPhone app and used the iPhone screen to show the camera preview. The iPhone preview app runs at 20 FPS with an image size of 480×320 received from the camera in real time. The iPhone can also be used as a handheld remote control. It provides the same functions as the head gesture interface, with zooming and shutter trigger. We used the iPhone as a handheld remote control for comparison with the proposed gesture interface in the user study.

A desktop PC with an Intel Core 2 Q8300 2.5-GHz CPU was used for image processing and gesture recognition. The camera was connected to the desktop PC via a USB cable, and the PC sent the preview to the iPhone through a WiFi connection.

We used the OpenCV library [26], which provides an implementation of Lucas-Kanade optical flow tracking to estimate head motions. The application program was written in C++. The performance of the implemented gesture-recognition algorithm is summarized in **Table 1**. Gesture recognition had the same performance at different face sizes in the image, as we scaled the face to a standard size of 120×120.

Table 1 Performance of gesture recognition

	Process time (ms)
Nodding and shaking	1.2
Mouth-opening	0.5

The prototype system was constructed in our laboratory and used for the experiments described in the following section (**Fig. 8**).

5. User Study

A user study was performed to evaluate the usability of the proposed interface. Although usability evaluations have been discussed in many previous reports [27] [28], we followed the ISO 9241 usability definition [29] and collected data on three distinct aspects of the proposed interface: effectiveness, efficiency, and user satisfaction. We also compared the gesture interface with a handheld remote control.

Effectiveness. This measures the accuracy and completeness with which users can achieve specified goals in particular environments. We arranged con-

Fig. 8 Experimental environment and apparatus.

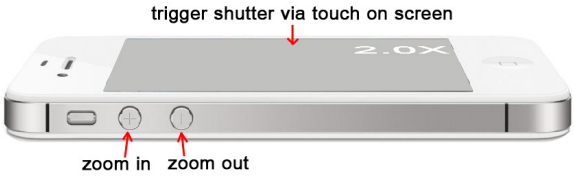


Fig. 9 iPhone remote control: two buttons to control zooming, touch on the screen to take shots.

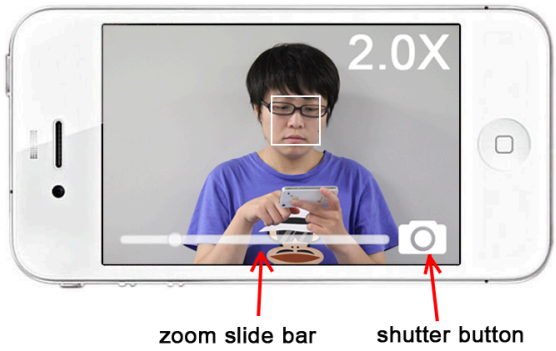


Fig. 10 iPhone remote control. The screen shows the camera preview and any camera information. There are two user interfaces: a shutter button that triggers the camera to take shots, and a zoom slide bar to control zooming.

secutive tasks to allow the user to perform the gestures many times to complete three types of shots: full-body, close-up, and upper-body shots. We evaluated the task completeness and the gesture recognition rate. We also compared the results to the handheld remote control when the users completed the same tasks.

Efficiency. This reflects the resources expended in relation to the accuracy and completeness of the goals achieved. We observed how many times participants actually performed the gestures to complete the three shots, and the time to completion.

User satisfaction. This is the comfort and acceptability of the system to its users and other people affected by its use. We used subjective assessment expressed on a five-point Likert scale.

5.1 Apparatus

We set up the experiment indoors under standard daylight conditions. We did not investigate the robustness of the algorithm to lighting conditions, as it depends on the implementation of face detection and Lucas-Kanade optical flow, which have been examined in previous studies [30] [31]. The camera was placed on a tripod. The experimental setup is shown in **Fig. 8**.

The iPhone was attached to the top of the camera. Two types of remote control were implemented: a button remote control and a touch remote control. The button remote control represents the traditional method, which provides users with eyes-free operation. **Fig. 9** shows the application where the two physical buttons of iPhone were used to control zooming, and where the touchscreen was used to trigger the camera shutter. On the other hand, the touch-based interface (see **Fig. 10**) provides the live preview on screen and the graphical user interface. The graphical user interface on the touchscreen includes two distinct functions: a shutter button icon that triggers the camera to take a picture, and a zoom slide bar to control zooming.

5.2 Participants

We recruited 12 subjects (6 female, 6 male) aged 23-30 years (mean = 25.6, SD = 1.6). Each participant was paid 1000 yen to participate in the study, which took 20-40 min. All participants were anonymous volunteers who saw our recruitment announcement in the international researchers' mailing list in the city of Tsukuba.

5.3 Task and procedure

Each experimental session involved one participant and consisted of four phases: 1) The experimenter introduced the system and allowed the participant to become familiar with it; 2) the participant completed consecutive tasks to measure the effectiveness and efficiency of the interface; 3) the subject completed a questionnaire as a subjective assessment of the system; and 4) user gestures were recorded to calculate the recognition accuracy and for future improvement.

In the first phase, the experimenter explained and demonstrated the two basic functions: zooming and shutter triggering. As this was likely to be the first time that the participants had experienced such an interface, each participant was shown how to use it and given time to become familiar with the system. The use of the iPhone remote control was introduced at the same time.

In the second phase, participants used both the gesture interface and a handheld remote control to complete three consecutive tasks, namely taking full-body, close-up, and upper-body portraits (**Fig. 11**). The participants stood in the same position during the tests, but were allowed to explore the head gesture interface or handheld remote control to zoom and take several shots. The three types of photographs are illustrated in **Fig. 11** and summarized below.

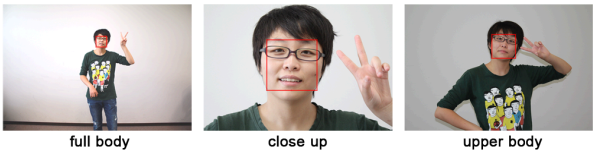


Fig. 11 Three types of photograph: full-body, close-up, and upper body.

Full-body shot. No zoom required. Perform shutter trigger command after standing in the correct position (size of face region from 60×60 to 90×90 in the image).

Close-up shot. The participant zooms the camera from full-body to close-up (size of face region from 260×260 to 340×340 in the image) and takes the picture. This task tests the zooming in function.

Upper body shot. The participant zooms out from close-up to upper body (size of face region from 120×120 to 180×180 in the image) to take a picture. This task tests the zooming out function.

The two techniques (i.e., head gesture interface and using the remote control) were arranged under the same conditions as outlined above. To avoid bias in the experiment, half of the participants used the handheld remote control first, and the other half used the head gesture interface first. The interaction gestures and time to completion of each of the three tasks were automatically recorded. The portraits were saved on the camera and PC.

In the third phase, the subjects completed a questionnaire as mentioned above, and in the fourth phase, the subjects participated in a separate experiment in which they performed the three types of gestures several times. The process was recorded as a video file for offline analysis of the accuracy of gesture recognition. Participation in this phase was voluntary.

5.4 Results

5.4.1 Effectiveness and efficiency

We assessed three factors to determine the effectiveness and efficiency of the proposed gesture interface, i.e., task *completeness*, the *number of gestures* participants performed to complete the tasks, and the *recognition rate* of gestures. We also collected a subjective assessment of effectiveness of the gesture interface compared to the handheld remote control.

The task completeness results are shown in **Table 2**. All of the participants completed the three types of pictures using both the gesture interface and the handheld remote control. We concluded that the head gesture interface is effective for users to control the zooming function and take self-portraits.

Table 2 Completeness

	Full-body	***Close-up***	***Upper body***
Gesture interface	100%	100%	100%
Remote (button)	100%	100%	100%
Remote (touch)	100%	100%	100%

To evaluate the efficiency of the gesture interface we recorded the number of gestures performed by each of the 12 users during the experiment sessions. The results are shown in **Table 3**. The mean values were calculated and the standard deviations are shown in parentheses.

Table 3 Performance for the three types of picture

	Nodding	***Shaking***	***Time***
Full-body to Close-up	10.9 (1.1)	0.8 (1.4)	7.6 s (1.2 s)
Close-up to Upper body	1.4 (1.9)	16.3 (1.7)	7.8 s (2.0 s)

The full-body shot served as the starting point, for which the participants stood in a specified position and performed the *mouth-opening* gesture to take the first picture. Thus, there were no nods or head shakes. After this, participants performed the *nodding* gesture to zoom in and take a close-up shot.

This took 10.9 (SD = 1.1) nods and 7.6 s (SD = 1.2 s) on average to complete the task and take a shot. Sometimes, the participants had to perform the *shaking* gesture to zoom out slightly as an adjustment after zooming in. However, this was rare. The upper body shot, performing the shaking gesture to zoom out from close-up to upper body, took on average 16.3 (SD = 1.7) shakes and 7.8 s (SD = 2.0 s) to complete. In some cases, participants performed the *nodding* gesture to zoom back in as an adjustment.

A comparison with using the remote control with regards to time is shown in **Fig. 12**. Users took about 2 times longer to complete the tasks when using gesture interface. This is generally due to the exactness of gesture recognition and the speed of zoom (0.75X per second). One-way analysis of variance (ANOVA) showed a significant effect ($p < 0.001$) of the three techniques both for close-up and upper body zoom tasks.

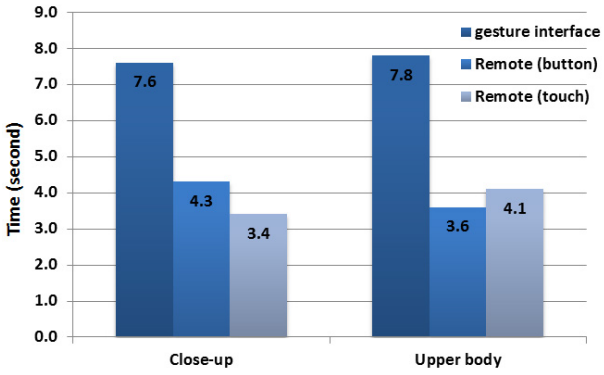


Fig. 12 Time comparison of the three techniques.

To examine gesture recognition accuracy, we collected 118 *nodding*, 126 *shaking*, and 20 *mouth-opening* gestures on AVI video samples from eight participants who agreed to allow video recording. We ran our application to recognize gestures on sample video to calculate the recognition rate. The results, summarized in **Table 4**, indicate good performance and recognition accuracy above 80% for *nodding* and *shaking* gestures. In addition, it achieved 100% accuracy recognizing *mouth-opening* gestures.

Finally, participants were asked to rate their satisfaction with the effectiveness of the proposed gesture interface compared to the handheld remote control on a five-point scale from "very poor" to "excellent". **Table 5** shows the results. We conducted an ANOVA test, but there was no significant effect ($F(2,33) = 0.49$, $p = 0.62$).

Table 4 Accuracy of gesture recognition

	Total	*Recognized*	*Missed*	*Accuracy*
Nodding	118	101	17	85.6%
Shaking	126	102	24	81.0%
Mouth-opening	20	20	0	100%

Table 5 Effectiveness of the interface for taking portraits

Evaluation	*Excellent*	*Good*	*Fair*	*Poor*	*Very poor*
Gesture interface	16.7%	50.0%	33.3%	0.0%	0.0%
Remote (button)	41.7%	33.3%	25.0%	0.0%	0.0%
Remote (touch)	41.7%	33.3%	16.7%	8.3%	0.0%

5.4.2 User satisfaction

We asked the participants to rate their level of satisfaction with the adopted function mappings (**Table 6**). All of the participants agreed that mapping of the *nodding* gesture to zoom in and the *shaking* gesture to zoom out were both intuitive and appropriate. Almost 60% of participants were satisfied with the mapping of the *mouth-opening* gesture to the shutter trigger function. About 40% of users reported a lack of satisfaction with the *mouth-opening* gesture; these users complained that this gesture may conflict with mouth motions when they are talking, or may cause unwanted triggering of the shutter countdown while preparing for the picture. Our application included a cancel function in which the user can perform the *head shaking* gesture to cancel the countdown once it has started. The users evaluated this design as useful when they triggered the shutter by accident.

Table 6 Evaluation of function mappings

Evaluation	*Excellent*	*Good*	*Fair*	*Poor*	*Very poor*
Nodding	83.3%	16.7%	0.0%	0.0%	0.0%
Shaking	75.0%	25.0%	0.0%	0.0%	0.0%
Mouth-opening	41.7%	16.7%	33.3%	8.3%	0.0%

In addition, the users evaluated their satisfaction with regard to freedom and concentration while preparing for a picture when using the proposed interface compared to the handheld remote control (**Table 7**). About 90% of participants reported a

satisfied experience with the gesture interface (ratings of "Excellent" and "Good"), while about 66% participants felt satisfied with the handheld remote control, both the button and touch interfaces. An ANOVA test showed a significant effect, $F(2,33) = 3.79$, $p = 0.03$. Moreover, the post hoc tests by Tukey HSD showed that the gesture interface against touch remote control had a significant effect ($p = 0.03$). However, there were no significant effects of gesture interface vs. button remote control ($p = 0.19$) and button vs. touch remote control ($p = 0.64$).

Table 7 Freedom and concentration while preparing for a picture

Evaluation	*Excellent*	*Good*	*Fair*	*Poor*	*Very poor*
Gesture interface	41.7%	50.0%	8.3%	0.0%	0.0%
Remote (button)	16.7%	50.0%	33.3%	0.0%	0.0%
Remote (touch)	0.0%	66.7%	25.0%	8.3%	0.0%

Finally, participants rated their overall satisfaction with the proposed gesture interface compared to the handheld remote control (**Table 8**). All of the participants reported a satisfied experience with the gesture interface, while about 80% and 60% participants felt satisfied with the button remote control, and touch remote control respectively. An ANOVA test showed a significant effect, $F(2,33) = 3.98$, $p = 0.03$. Furthermore, the post hoc tests by Tukey HSD showed that the gesture interface against touch remote control had a significant effect ($p = 0.02$). But there were no significant effects of gesture interface vs. button remote control ($p = 0.17$) and button vs. touch remote control ($p = 0.63$).

Table 8 Impression of proposed interface vs. handheld remote control

Evaluation	*Excellent*	*Good*	*Fair*	*Poor*	*Very poor*
Gesture interface	41.7%	58.3%	0.0%	0.0%	0.0%
Remote (button)	8.3%	75.0%	16.7%	0.0%	0.0%
Remote (touch)	16.7%	41.7%	33.3%	8.3%	0.0%

6. Discussion

6.1 Validity of design

Here, we review the appropriateness of our design goals based on the results of the survey and our observations of the participants' interactions with the self-portrait camera. We also examine whether our implementation satisfied these goals based on our observations of how the participants used the system.

Small size of frontal screen - The participants could interact with the camera with head gestures within a distance of 2 m. Although the participants stood 2 m from the camera and could not see the preview in detail on the 3.5-inch screen, all of them could still determine the zooming status and perform gestures to control zooming in/out from close-up to upper body and take the shots.

Considering strong pattern of motion gestures - Our observations in the present study confirm that the *nodding* and *shaking* motions are strong gesture patterns, which were rarely triggered by accident. However, the *mouth-opening* motion was less distinct and was unexpectedly recognized many times, particularly when participants were speaking or preparing for pictures. However, this gesture is still one of the best candidates for an intuitive gesture, as almost 60% of participants expressed a preference for it.

Head gestures only - The original motivation for introducing a head gesture interface was that the head and face are always available in images when developing vision-based gesture interfaces for zooming. Other gestures, such as hand or body gestures, cannot be used if the camera is zoomed close in. In the user study, we found that head gestures can provide an effective interface for controlling a self-portrait camera, particularly for controlling the zooming function.

Real-time processing - Gesture recognition can be achieved rapidly, within 1.2 ms and 0.5 ms for *nodding* and *shaking*, and *mouth-opening*, respectively. Thus, the algorithm used here can be implemented and will show good performance on a modern digital camera.

The observations and survey results obtained here indicate that the proposed gesture interface fulfills almost all of the requirements set at the beginning of the present study, thus suggesting that it is very effective for the stated application. In general, the users reacted very positively; they derived enjoyment from interacting with the self-portrait camera and were satisfied with the concept of the gesture interface. However, there were several slight problems, and these are discussed in the following subsection.

6.2 Limitations and future work

Our proposed self-portrait system is a "proof-of-concept" implementation, and so there were some hardware limitations. We used a Canon DSLR cam-

era to take portraits, a PC connected to the camera for image processing and gesture recognition, and an iPhone attached to the camera as a frontal screen. These hardware limitations were frustrating for users when considering a practical self-portrait camera. However, we believe that the development of a smart camera or computational camera [32] equipped with a frontal screen and supporting custom programming will soon be developed. We hope to implement the gesture interface described here in such cameras in the future.

While the proposed gesture-recognition algorithm achieved a good level of accuracy, better results are expected with future improvements. The development of methods for extracting facial features to allow localization of the eyes, nose, mouth, and lips for detecting accurate facial motions is also expected in future.

Another limitation of the proposed interface is the inability to map to many more camera functions, such as aperture, shutter speed, ISO, white balance, and so forth. Further studies are required to develop an improved gesture interface to provide access to such camera functions.

7. Conclusions

This paper presented a novel vision-based head gesture interface for controlling a camera that can help users to take self-portraits effectively and efficiently. The user can perform intuitive *nodding* and head *shaking* gestures to control the camera zooming in/out. The user can also perform a *mouth-opening* gesture to trigger the camera shutter to take a picture. Based on the results of a user study, we concluded that the proposed gesture interface has a high degree of usability (effectiveness, efficiency, and satisfaction). In addition, the proposed gesture interface is useful than a handheld remote control, particularly than touch interface. Hence, this proposed gesture interface is a promising concept for the future development of self-portrait cameras.

Reference

[1] Okabe, D., Ito, M., Chipchase, J., et al.: The social uses of purikura: photographing, modding, archiving, and sharing; PICS Workshop Ubicomp 2006, pp.2-5 (2006).

[2] Counts, S. and Fellheimer, E.: Supporting social presence through lightweight photo sharing on and off the desktop; Proceedings of the SIGCHI conference on Human factors in computing systems, pp.599-606 (2004).

[3] 100 seriously cool self-portraits (and tips to shoot your own!), available from: <http://photo.tutsplus.com/articles/inspiration/100-seriously-cool-self-portraits-and-tips-to-shoot-your-own/> (accessed 2013-3-28).

[4] 4 tips for taking gorgeous self-portrait and outfit photos, available from: <http://www.shrimpsaladcircus.com/2012/01/self-portrait-outfit-photography-guide.html> (accessed 2013-3-28).

[5] Taking a great self portrait with your camera, available from: <http://www.squidoo.com/self-portrait-tips> (accessed 2013-3-28).

[6] Chu, S. and Tanaka, J.: Hand gesture for taking self portrait; Proceedings of 14th International Conference on Human-Computer Interaction (HCI International 2011), Human-Computer Interaction, Part II, LNCS 6762, pp.238-247 (2011).

[7] Chu, S. and Tanaka, J.: Head nod and shake gesture interface for a self-portrait camera; The Fifth International Conference on Advances in Computer-Human Interactions (ACHI 2012), pp.112-117, IARIA (2012).

[8] Sony Party Shot, available from: <http://www.sony.jp/cyber-shot/party-shot/> (accessed 2013-3-28).

[9] Casio TRYX camera, available from: <http://www.casio-intl.com/asia-mea/en/dc/ex_tr150/> (accessed 2013-3-28).

[10] Davis, J. and Vaks, S.: A perceptual user interface for recognizing head gesture acknowledgements; Proceedings of the 2001 workshop on Perceptive user interfaces, pp.1-6 (2001).

[11] Li, R., Taskiran, C., and Danielsen, M.: Head pose tracking and gesture detection using block motion vectors on mobile devices; Proceedings of the 4th international conference on mobile technology, applications, and systems and the 1st international symposium on Computer human interaction in mobile technology, pp.572-575 (2007).

[12] Tan, W. and Rong, G.: A real-time head nod and shake detector using HMMs; Expert Systems with Applications, Vol.25, No.3, pp.461-466 (2003).

[13] Morency, L. and Darrell, T.: From conversational tooltips to grounded discourse: head pose tracking in interactive dialog systems; Proceedings of the 6th international conference on Multimodal interfaces, pp.32-37 (2004).

[14] Morency, L., Sidner, C, Lee, C., and Darrell, T.: Head gestures for perceptual interfaces: The role of context in improving recognition; Artificial Intelligence, Vol.171, No.8-9, pp.568-585 (2007).

[15] Yoda, I., Sakaue, K. and Inoue, T.: Development of head gesture interface for electric wheelchair; Proceedings of the 1st international convention on Rehabilitation engineering and assistive technology: in conjunction with 1st Tan Tock Seng Hospital Neurorehabilitation Meeting, pp.77-80 (2007).

[16] Palleja, T., Rubion, E., Teixido, M., et al.: Using the Optical Flow to Implement a Relative Virtual Mouse Controlled by Head Movements; Journal of Universal Computer Science, Vol.14, No.19, pp.3127-3141 (2008).

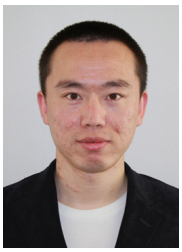
[17] Gizatdinova Y., Spakov O. and Surakka V.: Com-

parison of video-based pointing and selection techniques for hands-free text entry; Proceedings of the International Working Conference on Advanced Visual Interfaces, pp.132-139 (2012).
[18] Grauman, K., Betke, M., Lombardi, J., et al.: Communication via eye blinks and eyebrow raises: Video-based human-computer interfaces; Universal Access in the Information Society, Vol.2, No.5, pp.359-373 (2003).
[19] Shi, J. and Tomasi, T.: Good features to track; Technical Report, Cornell University (1993).
[20] Viola P. and Jones M.: Rapid object detection using a boosted cascade of simple features; Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp.1:511-1:518 (2001).
[21] Baker, S. and Matthews, I.: Lucas-Kanade 20 years on: a unifying framework; International Journal of Computer Vision, Vol.56 No.3, pp.221-255 (2004).
[22] Trujillo, L. and Olague G.: Automated design of image operators that detect interest points; Journal of Evolutionary Computation, Vol.16, No.4, pp.483-507 (2008).
[23] Bay, H., Tuytelaars, T. and Gool, L.: SURF: speeded up robust features; Computer Vision and Image Understanding (CVIU), Vol.110, No.3, pp.346-359 (2008).
[24] Bruhn, A., Weickert, J. and Schnorr, C.: Lucas/Kanade meets Horn/Schunck: combining local and global optic flow methods; International Journal of Computer Vision, Vol.61, No.3, pp.211-231 (2005).
[25] Canon Digital Camera Software Developers Kit (Canon SDK), available from: <http://usa.canon.com/cusa/consumer/standard_display/sdk_homepage/> (accessed 2013-3-28).
[26] Open Source Computer Vision Library (OpenCV), available from: <http://opencv.org/> (accessed 2013-3-28).
[27] Ivory, Y. M. and Hearst, A, M.: The state of the art in automating usability evaluation of user interfaces; ACM Computing Surveys, Vol. 33, No.4, pp.470-516 (2001).
[28] Folstad, A., Law E. and Hornbak K.: Analysis in practical usability evaluation: a survey study; Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems, pp. 2127-2136 (2012).
[29] ISO 9241-11: Ergonomic requirements for office work with visual display terminals (VDTs); Part 11, guidance on usability (1998).
[30] Wang, H., Li, S. and Wang, Y.: Face recognition under varying lighting conditions using self quotient image; Proceedings of the Sixth IEEE international conference on Automatic face and gesture recognition, pp.819-824 (2004).
[31] Molnar, J., Chetverikov, D. and Fazekas, S.: Illumination-robust variational optical flow using cross-correlation; Computer Vision and Image Understanding, Vol.114, No.10, pp.1104-1114 (2010).
[32] Adams, A., Talvala, E. V., Park, S. H., et al.: The Frankencamera: an experimental platform for computational photography; Proceeding ACM SIGGRAPH 2010 papers, pp.29:1-29:12 (2010).

(received Oct. 3, 2012, revised Feb. 28, 2013)

Biography

Shaowei Chu (Student Member)



Shaowei Chu is a PhD candidate in the Department of Computer Science, Graduate School of Systems and Information Engineering, University of Tsukuba. His research interests include human-computer interaction and ubiquitous computing. He received a BSc in computer science from the Liaoning University of Technology in 2006. He received a MSc in computer science from the Communication University of China in 2008.

Jiro Tanaka (Member)

Jiro Tanaka is a Professor of Department of Computer Science, University of Tsukuba. His research interests include ubiquitous computing, interactive programming, and computer-human interaction. He received a BSc and a MSc from University of Tokyo in 1975 and 1977. He received a PhD in computer science from the University of Utah in 1984. He is a member of ACM, IEEE and IPSJ.