Cyclops: Wearable and Single-Piece Full-Body Gesture Input Devices

Liwei Chan[*] **Chi-Hao Hsieh**[†] **Yi-Ling Chen**[*] **Shuo Yang**[†]
Da-Yuan Huang[*] **Rong-Hao Liang**[*] **Bing-Yu Chen**[*]
National Taiwan University
[*]{liweichan, yilingchenntu, d99944006, rhliang, robin}@ntu.edu.tw
[†]{p121225, yangs}@cmlab.csie.ntu.edu.tw

ABSTRACT

This paper presents Cyclops, a single-piece wearable device that sees its user's whole body postures through an ego-centric view of the user that is obtained through a fisheye lens at the center of the user's body, allowing it to see only the user's limbs and interpret body postures effectively. Unlike currently available body gesture input systems that depend on external cameras or distributed motion sensors across the user's body, Cyclops is a single-piece wearable device that is worn as a pendant or a badge. The main idea proposed in this paper is the observation of limbs from a central location of the body. Owing to the ego-centric view, Cyclops turns posture recognition into a highly controllable computer vision problem. This paper demonstrates a proof-of-concept device and an algorithm for recognizing static and moving bodily gestures based on motion history images (MHI) and a random decision forest (RDF). Four example applications of interactive bodily workout, a mobile racing game that involves hands and feet, a full-body virtual reality system, and interaction with a tangible toy are presented. The experiment on the bodily workout demonstrates that, from a database of 20 body workout gestures that were collected from 20 participants, Cyclops achieved a recognition rate of 79% using MHI and simple template matching, which increased to 92% with the more advanced machine learning approach of RDF.

Author Keywords

Full-body gesture input; posture recognition; single-point wearable devices; ego-centric view

ACM Classification Keywords

H.5.m. Information Interfaces and Presentation (e.g. HCI): Miscellaneous

INTRODUCTION

Body gestural input has emerged as a popular natural user interface and become widely accessible since the release of Microsoft Kinect. Recently, more depth-sensing cameras have enabled reliable body tracking, but they have a limited working distance owing to the cameras' imaging fields-of-view.

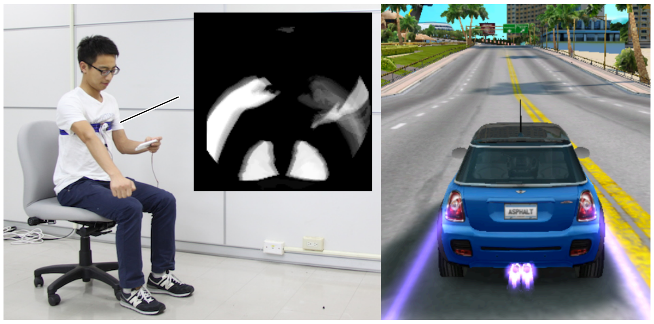

Figure 1. Cyclops is a wearable single-piece bodily gesture input device that captures an ego-centric view of the user. With Cyclops worn at the center of the body, the user plays a racing game on the mobile phone with hand and foot interactions. The user pushes forward an imaginary gear stick on his right to trigger a nitro turbo.

To free users from the location constraints, novel wearable devices that are dedicated to partial body-centric inputs, such as arm [20], foot, palm [6], and finger-based [5] touch or gestural interactions, have been developed and demonstrated. Other research works have developed full-body motion capture by distributing motion sensors [19][15][32] around body parts of the user. However, wearing these sensors is commonly inconvenient, even though they are incorporated into motion-capture suits.

Recent research has developed full-body gestural input using single devices. Smart phones, for example, have been utilized for recognizing coarse-grain activities [18] such as walking, running and sleeping. Allowing for more interactivity, single wireless units that are carried by users [7] or arm-worn accelerometers [25] have been demonstrated for the recognition of motion gestures.

Cyclops

We present Cyclops, a single-piece wearable device that sees a user's bodily postures through an ego-centric view of the user. Figure 1 demonstrates the Cyclops device, and an example of the device's view of the user performing hand and foot interactions in a racing game. This ego-centric view offers two benefits. First, because Cyclops is fixed on a user's body, the ego-centric view presents a registration-free image, allowing body gestural recognition using simple template matching

CHI'15, April 18 - 23 2015, Seoul, Republic of Korea.

ACM 978-1-4503-3145-6/15/04$15.00
http://dx.doi.org/10.1145/2702123.2702464

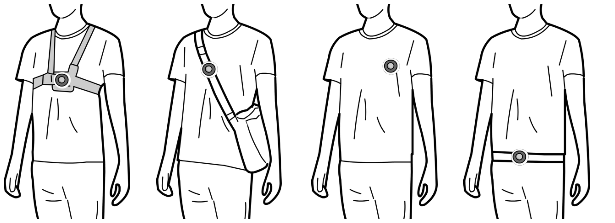

Figure 2. Cyclops can be worn around the middle of the body. Different placements favor head or foot interactions.

and motion history images. Second, a user's limbs enter the ego-centric field of view from outside of it, allowing foreground limb extraction to be more independent of cluttered backgrounds since the limbs are identified as foreground objects that penetrate the outermost rim of the image.

Figure 2 shows different ways to put on Cyclops as a wearable device. Displacement of the device toward the upper or lower body further facilitates head or foot interactions. Users can move the device according to the task at hand.

Contributions

The main contribution of this paper is the concept of full body gestural recognition using a fisheye ego-centric view of the user. To demonstrate the idea, this work (1) presented a proof-of-concept prototype and (2) evaluated the potential of the ego-centric view for full-body gestural recognition using motion history images with template matching and random decision forest methods.

RELATED WORK

This work concerns body-centric interactions that are identified using wearable devices, concentrating on full-body gestural recognition with single-point sensing techniques.

Body Inputs Using Wearable Cameras

Body-mounted cameras for monitoring gestural interactions have been extensively studied. Previous research [22] has demonstrated that locating cameras at various positions on the body enables unique interactions. Head-mounted cameras [24][36][8], for example, yield a mobile interaction space that is under the user's perspectives, but they do not support eye-free interactions. Attaching cameras to the body rather than to the head or close to the eyes, provides more dedicated interaction spaces. For example, attaching cameras to the shoulders [13] or chest [23][12], can turn users' palms into interactive surfaces, and facilitate hand gestural input. A camera on a wrist can be used track hand gestures [38][17]. A camera on a foot, pointing upward [2], provides an interaction space that incorporates the user's upper body. However, none of these developments can be used to sense the motions of feet so none supports the identification of full-body gestures. The goal of this work is to identify full-body motion gestures using single-piece wearable device.

Body Inputs Using Wearable Low-Level Sensing

Light-weight wearable devices, rather than cameras, can be used in low-level sensing. To make the body a touchable interface, capacitive sensing is integrated with users' clothing [28]. Scott et al. [31] proposed the recognition of foot gestures using a phone in a pocket. EarPut [21] instrumented the ear as an interactive surface for touch-based interactions using capacitive sensors.

Much research on body input is dedicated to hand-based interactions. Data gloves have been widely studied [35][4] in the field of HCI, and other techniques have been developed. Analyzing the sound that bounces through bones [14][26] allows tapping on the skin to be detected by acoustic sensors that are worn on the arm. SenSkin [27] made the skin into a touch interface by sensing skin deformation that is caused by the application of a force tangentially to the skin. PUB [20] facilitated touch interactions on the forearm by using a distance sensor on the wrist. Touche [30] enabled discrete hand gestures using swept frequency capacitive sensing. Saponas et al. [29] enabled hand gestural interaction by analyzing forearm electromyography. uTrack [6] enabled touch interaction on the palm by magnetic localization. FingerPad [5] used magnetic tracking to turn the index fingertip into a touch interface.

Depending on purpose and application, low-level sensing techniques be implemented on certain parts of the body. However, full-body motion input requires a body sensor network of sensors that are distributed on body parts [15][16], which may be inconvenient for users to put on.

Full-Body Gestures Using Single-point Devices

Activity recognition using single-point devices, such as smart phones, has been widely explored [18][10]. Relevant research has focused on the coarse-grain classification of daily activities such as walking, running, sitting, standing, and sleeping, using inertial sensors in smart phones. Real-time recognition is not their main concern. More recent research, RecoFit [25] has allowed repetitive exercises to be recognized in an interactive manner using a single arm-worn inertial sensor. Humantenna [7] enabled the identification of whole-body motion gestures by treating the human body as an antenna.

A single-point device involves minimal instrumentation. Previous research in this area, however, suffer from focusing on only limited motion gesture sets and an inability to detect static bodily gestures. Owing to its fisheye lens, Cyclops can recognize a rich set of static and moving full-body gestures.

HARDWARE PROTOTYPE

235-Degree Super Fisheye Lens

The main component of Cyclops is an ultra wide-angle camera that sees the user's limbs. Figure 3 displays the three tested types of wide-angle lens. Two are wide-angle lenses from GoPro[1] and Super Fisheye[2], and the other is an omni-directional lens from Omni-Vision[3]. Each lens is positioned at the center of the user's chest, observing him adopting five body postures.

[1] http://www.gopro.com/

[2] http://www.superfisheye.com/

[3] http://www.omnivision.com/

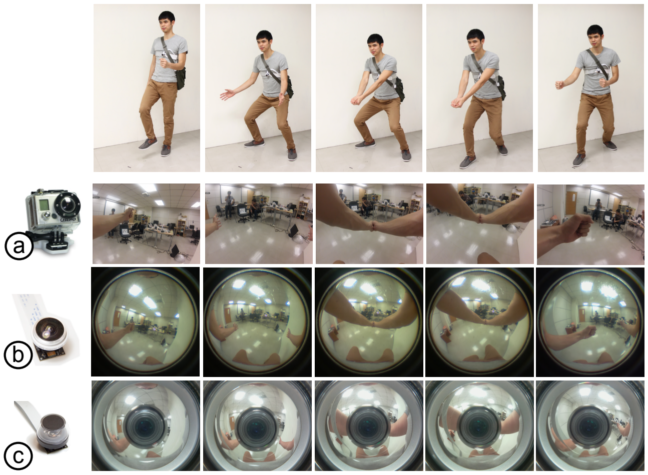

Figure 3. Comparison of three wide-angle camera lenses for observing five bodily gestures. (a) Go-Pro (180 degrees) wearable camera, (b) SuperFisheye (235 degrees), which was used in the Cyclops prototype herein, and (c) Omni-Vision, an omni-directional lens.

GoPro is a wearable imaging device that is equipped with an 180 degrees wide-angle lens, designed to capture the first-person experience of the wearer. The device can see the user's hand gestures but barely observes the user's head and legs. The 180-degree images thus obtained are presented as a reference, to suggest the need for a device with a wider viewing angle to capture full body interactions. Super Fisheye is the fisheye lens with the widest field-of-view that we could obtain commercially: it has a 235 degree field-of-view, and so easily captures the user's head and legs. Lastly, Omni-Vision, a omni-directional lens, has the widest field of vision at the expense of central vision. This omni-directional view clearly observes all limbs, but not those parts of the limbs that enter the central region. Accordingly, the Super Fisheye lens was chosen in the presented implementation owing to its ultra-fisheye field-of-view.

Enabling 235-Degree Field-of-view

The Raspberry Pi NOIR camera module's original field-of-view is about 90 degrees and not wide enough to cover the full 235-degree field-of-vision of the Super Fisheye Lens. Our previous prototype that used an unmodified NOIR camera module with the super fisheye lens achieved a field-of-vision of only 190 degrees. To fully realize the potential of the 235-degree lens, the original lens of the NOIR camera module was replaced with an 110-degree lens, as shown in Figure 4. Specifically, the lens in the NOIR camera module on the left the figure was removed. A 3D-printed ring-shaped connector formed a bridge between the 110-degree lens and the 235-degree lens.

Cyclops Wearable Device

To simplify foreground extraction, Cyclops was implemented with infrared imaging and active infrared illumination. Although color images could also be used to extract bare body limbs, they are more affected by various colors of clothing.

Figure 5 presents the components of our hardware prototype. The super fisheye lens with its 235-degree field of view provides an ego-centric view. Five infrared LEDs were attached around that lens to provide uniform illumination. The Raspberry Pi NOIR camera module captures infrared images which are then streamed wirelessly to a remote PC for further image processing. Notably, the Raspberry Pi NOIR camera module can see both visible and infrared light. To facilitate the extraction of limbs from images, an 850nm filter is added to block visible light and only infrared reflection from foreground objects such as limbs is accepted. A nine-DOF inertial motion sensor (IMU) is utilized to re-orient the fisheye images such that users need not worry about incorrectly putting on the wearable device, such as by putting it on upside down. The orientation data such as pan, pitch and yaw that are obtained by the IMU sensor are also applied in body gestural recognition in the experiments and applications.

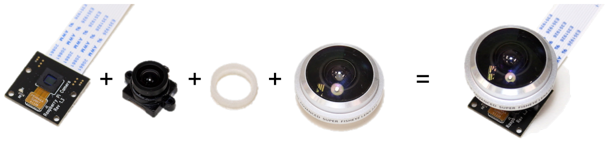

Figure 4. Cyclops' ultra-fisheye field-of-view is realized by firstly replacing the lens in the NOIR camera module with a 110-degree lens, and then bridging it with a 235-degree super fisheye lens.

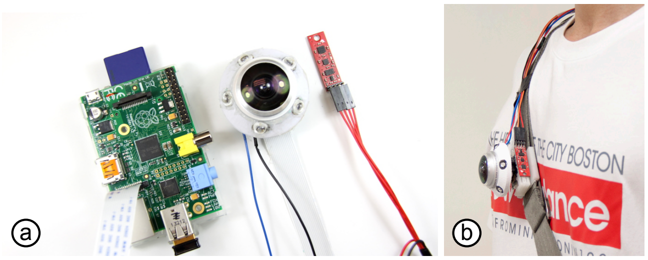

Figure 5. The Cyclops hardware comprises a super fisheye lens, an infrared illuminator around the lens, a Raspberry Pi NOIR camera module, and a 9-DOF inertial motion sensor. The Raspberry Pi module can stream the camera frames wirelessly at interactive speed.

The resulting prototype is packed with a 3D printed case, measuring about two inches in every dimension (width, height and length), excluding the Raspberry Pi and mobile power supply. The flexible data wire that connects the camera module to the Raspberry Pi is 90 cm long, which is long enough to allow the Raspberry Pi module and a mobile power supply to be stored in a fanny pack that is worn by the user.

Raspberry Pi streams images from the camera wirelessly over WiFi using the GStreamer Multimedia Framework. GStreamer[4] provides real-time streaming with an average delay of less than 300ms. In the implementation, a remote laptop receives approximately 20 frames per second from Cyclops, and the streaming delay is acceptable for our applications that run at an interactive speed.

RECOGNIZING BODILY GESTURES

Despite extensive research in this area, gesture recognition remains a challenging problem to solve in computer vision.

[4] GStreamer: http://gstreamer.freedesktop.org/

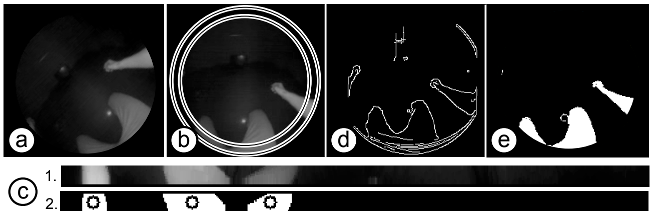

Figure 6. Image processing of the proposed algorithm for labeling components in the image as limbs. Accordingly, the limbs are extracted as a foreground image where they enter the fish-eye image. Notice that the source image in (a) is re-oriented using the information that is provided by the IMU.

Owing to its ego-centric image acquisition pipeline, Cyclops benefits from a sequence of *registration-free* input images from which the essential foreground objects (i.e. limbs) can be reliably identified using basic image processing techniques. Information about the motion of the user can then be further encoded into temporal templates in the form of a low-resolution, grayscale image. This work demonstrates that such compact representation of human actions is very effective for gestural recognition and the corresponding image classification problem can be effectively solved by standard pattern recognition and machine learning methods.

Based on the similar approach proposed in [37], we also take advantage of motion history image (MHI) and random decision forest (RDF) to identify stationary and moving bodily gestures. In the following, the process of foreground extraction will firstly be described. Based on this identification of the foreground limbs, two types of MHI [3] will be generated to represent the temporal motion in a single image. Bodily gestures will be shown to be recognized reasonably well using MHIs with straightforward template matching, and that the recognition rate can be increased by exploiting the RDF-based method [33].

Foreground Extraction

Figure 6 illustrates our image pre-processing pipeline. The basic idea is to extract foreground images from where the limbs enter the ego-centric view at the edge of the fisheye image. This strategy avoids dealing with non-limb foregrounds in the central part of the image. Firstly, the source images (Figure 6a) are re-oriented using the information that is provided by the IMU. Then, as highlighted in Figure 6b, the process is begun from a circular strip at the edge of the fisheye image. Figure 6c displays a straightened version of the circular strip. Then, Otsu thresholding is applied to the strip image to extract the foreground regions that potentially contain limbs. To separately identify limbs that overlap in the image, a vertical erosion along the length of the strip is performed to remove weak connections. The resulting connected components are further processed as follows.

From the geometric center of each component in the strip image, its corresponding position in the source image (Figure 6b) is identified, and used as a seed from which a foreground region is grown by performing image flooding. The Canny edge map of the source image, as shown in Figure 6d, is utilized to block the flooding operation. Figure 6e presents the overall foreground image that is obtained by aggregating all foreground regions. In simple cases, favorable results are obtained even in a cluttered environment. In complex cases, however, incorporating depth information from the infra-red image [11] or using a time-of-flight depth camera may be helpful. The foreground masks are used to compute the MHIs discussed in the next subsection.

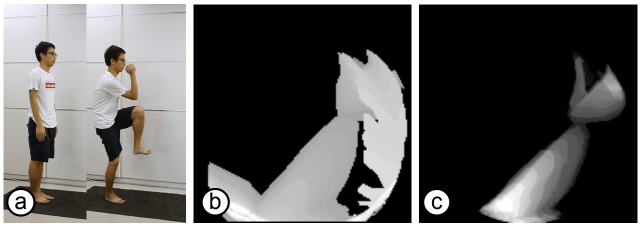

Figure 7. Examples of dMHI and iMHI that are generated from a sequence of images of a user performing a workout.

Motion History Image

Briefly, MHI is an image template in which non-zero pixels simultaneously record the spatial and temporal aspects of motion. A larger intensity value indicates more recent motion and intensity decays over time. MHI and many of its variations [1] have been extensively investigated in the field of action recognition. One key factor that affects the performance in MHI is image registration, which constitutes a difficult problem in the field of computer vision.

Like [37], we adopt two types of MHI in our implementation - the difference-MHI (dMHI) and the integral-MHI (iMHI). Specifically, for each incoming frame I_t at time t in an image sequence, we first resize I_t into a lower resolution of 75×75 and then compute the foreground mask I_t^m with the aforementioned foreground extraction algorithm. The dMHI is obtained by setting the pixels within I_t^m to 255 and decaying the rest pixels with a constant. To form the iMHI, foreground masks are summed up by $\sum_{i=0}^{k-1} w_{k-1-i} I_{t-i}^m$, where $w_i = \frac{2i}{k(k-1)}$ are the weighting coefficients, k is the number of past frames kept in the history, and i is the frame index. Figure 7 shows an example of dMHI and iMHI generated from a sequence of the user performing a workout gesture (right hand and left leg crunching while standing).

Random Decision Forest for Gesture Classification

RDF is a generic and powerful learning algorithm that has been used with much success in computer vision and medical image analysis applications [9]. One of the most notable examples is the Kinect body tracker [33]. Briefly, a forest is an ensemble of T decision trees, each comprising internal and leaf nodes. Each internal node is associated with a *split function* f_θ and a threshold τ. At test time, for a data point $\mathbf{x}$ arriving at an internal node, its corresponding *feature* is evaluated using f_θ and compared to τ. Based on the comparison result, $\mathbf{x}$ is assigned to either the left child or the right child of the current node. The above operation is repeated until $\mathbf{x}$ reaches a leaf node, where a predictor function is pre-learned and stored. The final classification of $\mathbf{x}$ is made by aggregating the results of all trees in the forest. In the following, we describe two different approaches of applying RDF classifiers to accomplish gesture recognition with Cyclops.

Standard RDF Classifier

The motion representation of MHI discussed above contains crucial information for gesture recognition and the framework of RDF-based classifiers [33, 37] can be readily applied to accomplish this task. Following [37], to classify a pixel $\mathbf{x}$ in a motion template image, we leverage the intensity differences of MHIs as features, as formulated below,

$$f_{\mathbf{u},\mathbf{v}}(I,\mathbf{x}) = I(\mathbf{x}+\mathbf{u}) - I(\mathbf{x}+\mathbf{v}), \tag{1}$$

where $\theta = (\mathbf{u}, \mathbf{v})$ are offset vectors relative to $\mathbf{x}$ and $I(\cdot)$ denotes the intensity value at a specific pixel location of a given MHI. The intuition behind this feature selection function is that it enables f_θ to learn the spatial extent and configuration of body parts in the motion signature image. At training time, randomly generated offset vectors are evaluated to obtain the feature values by Equation (1). For each internal node, the pair of $\mathbf{u}, \mathbf{v}$ and a corresponding threshold that best separates the labeled training data are kept. More details on applying RDF classifier to gesture recognition are referred to [33, 37].

Multi-layered RDF Classifier

In [34, 11], multi-layered decision forests are proposed to accomplish gesture recognition or depth estimation on mobile devices.The basic idea behind these approaches is to exploit expert forests trained for a particular task (e.g., roughly classify an image into several levels of quantized depth) and only those images corresponding to a particular class are forwarded to a second forest, trained only on examples from this class. Each of the forests then needs to model less variation and hence can be comparatively well performed.

Since Cyclops aims to handle full-body gestures, it is thus possible for the gesture recognizer to deal with rich and diverse gesture types. However, building a training database that covers all gestural variations to train a single classifier is a non-trivial but essential task for any machine learning algorithms. Inspired by [34, 11], we also propose a multi-layered architecture of RDFs with Cyclops. Differing from previous methods, which focus on a single source of input data (e.g., images), our observation is that additional orientation data captured while actions are being performed are potentially useful to facilitate the task of gesture classification.

Continuous orientation signatures are particularly useful when dealing with a specially purposed gesture set, such as a body workout. These gestures are carefully designed and very likely to yield distinct patterns of body orientation across different classes. For example, the orientation signatures of the standing gestures beginning with horizontal tilting differ from those of the face-down gestures beginning with a series of downward tilting. With the built-in IMU module, it is thus very simple to establish *two*-layered RDFs for Cyclops.

1. First layer: before training, the gesture set needs to be roughly divided into N categories according to their orientation patterns. For each motion category, we collect a set of orientation signatures for training. Specifically, *pitch* and *yaw* angles from n consecutive timestamps are concatenated into a vector $\mathbf{o} \in \mathbb{R}^{2n}$. During training, one dimension of $\mathbf{o}$ is randomly selected as a feature. In other words, f_θ is equivalent to an axis-aligned hyperplane in the feature space that separates the incoming data into two disjoint sets.

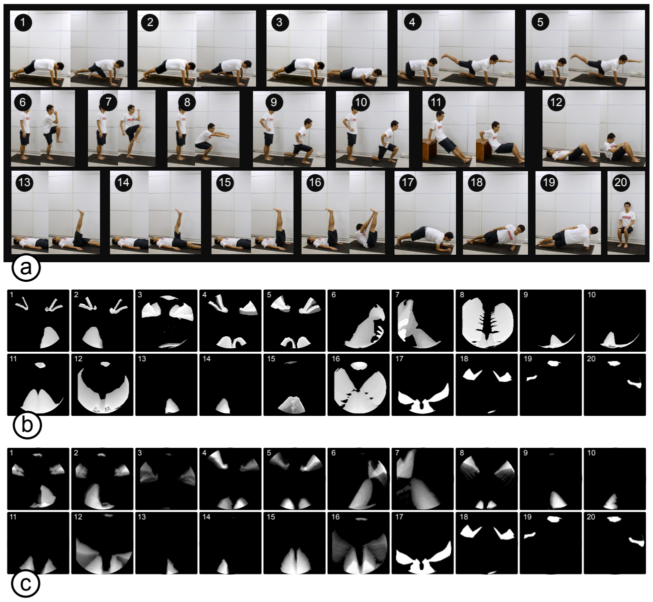

Figure 8. Four types of workout exercises. (a) A total of 20 captured moving and stationary bodily gestures. (b) and (c) shows examples of dMHIs and iMHIs, respectively, of bodily gestures in workout, as observed by Cyclops.

2. Second layer: we train N standard RDF classifiers for each motion category with the training data limited to the MHIs belonging to a certain category.

At test time, an orientation signature $\mathbf{o}'$ is firstly classified by the *single* first-layer forest. Once the motion category of $\mathbf{o}'$ is determined, the corresponding MHI I' is forwarded to the corresponding second-layer forest, by which the final gesture type is solely determined.

SYSTEM EVALUATION

To evaluate the performance of Cyclops, we chose to conduct a study on a body workout data set consisting of a wide variety of full-body gestures. In the following, we describe the experimental settings, data preparation procedures, and evaluation results.

Experimental Settings

The study included 20 exercises, which comprised both moving exercises and stationary postures, and were categorized into four types. As presented in Figure 8a, three of the four types of exercises involved motion; these comprised face-down (1-5), six standing (6-11), and five lying-down (12-16) exercises. The last group of exercises involved four stationary postures (17-20).

In a pilot study, we observed that loose clothing may occasionally block the Cyclops' view. Additionally, the images

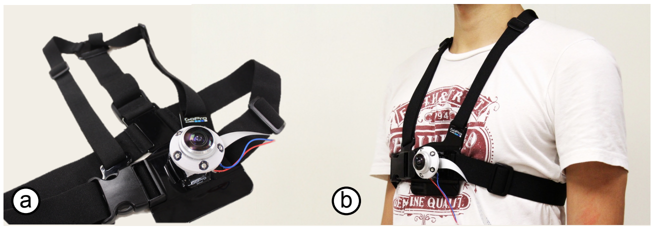

Figure 9. The Cyclops device is set firmly on the chest mount harness to fit a wide range of users.

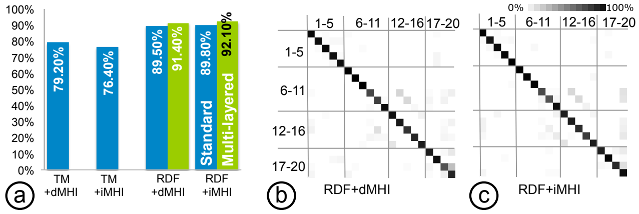

Figure 10. (a) Recognition rates of workout gestures achieved by Leave-one-person-out cross-validation with various algorithms. (b)(c) Confusion matrices of gesture recognition obtained using RDF + dMHI and RDF + iMHI, respectively.

observed by Cyclops from female and male participants differed greatly, mainly because the camera was easily occluded by the chests of female participants, below or above which Cyclops must be worn. Hence, to exclude these unfavorable factors from evaluation, only male participants were recruited and asked to be dressed properly.

Participants

Twenty participants were recruited from our department to perform the workout gesture set. They were aged between 21 and 25 (mean = 24.2, std = 2.95). Their heights (mean = 175.5 cm, std = 5.48 cm), weights (mean = 68.4 kg, std = 10.21 kg), and BMI values (mean = 22.1, std = 2.92) were recorded. All participants were able to perform the entire workout set without any problems.

Training Data Acquisition

At the beginning of this study, every participant received a 10-minute training. The experimenter firstly helped the participants put on the Cyclops device, and then explained the exercises to be performed. To ensure that all participants wore Cyclops properly, a 3D-printed hinge was made to fix the device firmly to a chest-mounted harness as shown in Figure 9. Following the experimenter, the participants had to physically perform dry runs of all of the workout exercises to ensure that they fully understood the details.

During the study, a monitor prompted example videos of each trial exercise to the participants. As soon as the participant was ready to perform the action, he notified the experimenter. A beep sound was then played to indicate that the participant should perform the action; meanwhile both the image and the accompanying orientation (yaw, pitch, and roll) data, provided by the IMU sensor, were recorded. Upon completing an action, the participant had to stay stationary until a second beep was heard, indicating the end of the recording. This procedure simplified the segmentation of the captured data. The above process was repeated for each exercise until the entire workout set was completed. The participant then took a rest before performing a second round of the workout.

Each participant performed two rounds of the entire workout set, and each exercise in the workout set was performed twice. Hence, each participant generated 20 (exercises) × 4 (repetitions) = 80 labeled images (either dMHI or iMHI) and orientation data sequences.

Performance Evaluation

A technical evaluation was carried out to determine the recognition rates achieved by applying 1) basic template matching, 2) standard RDF classifier and 3) multi-layered RDF classifier with both dMHI and iMHI motion signatures. *Leave-one-person-out* cross-validation was performed on the workout dataset to evaluate the performance of each method. For each subject, 1520 training images and 80 test images were utilized to measure the accuracy of the evaluated method to predict the unknown gestures. The obtained recognition rates are then averaged to indicate the overall performance.

Notably, a non-gesture class must be included for a deterministic classifier like RDF. However, the non-gesture class is more difficult to define in this study than in the work of [37], because Cyclops captures full-body gestures. In a body workout exercise, some random bodily motions may even partially coincide with the workout gestures. To limit the uncertainty that could be introduced by an invalid non-gesture class, the non-gesture class was not included in this evaluation.

Performance of Template Matching

Served as a baseline approach for comparison, template matching (TM) is implemented straightforwardly by measuring the distance between two MHI images as the summation of pixel-wise absolute differences of intensity values. For each test image, the label of the MHI image in the training set with the smallest distance is returned as the gesture type. As shown in Figure 10a, the average recognition rates of applying template matching to iMHI and dMHI achieved 76.4% and 79.2%, respectively. This reasonably good performance is accounted for by the registration-free images captured by Cyclops, which considerably simplified the recognition tasks.

Performance of Standard RDF Classification

To obtain the standard RDF classifiers, the following parameters were used to train a three-tree forest for each of the 20 subjects. For each training image, 2000 sample pixels were randomly selected as data points. All the data points were passed to the root node to start the recursive training process. For each intermediate node, 2000 candidate features generated by Equation (1) and 50 candidate thresholds per feature were adopted to determine the best split function. Tree node splitting stops at the maximal level of 19 or the information gain is less than a prescribed threshold in terms of Shannon entropy (we empirically set it to 0.01). In our current C++ implementation, the tree trainer was run on a single core of i-7 3.4 GHz CPU and training a forest took around six hours. We have also tried to accelerate RDF training by concurrently training individual trees with OpenMP. The training time can

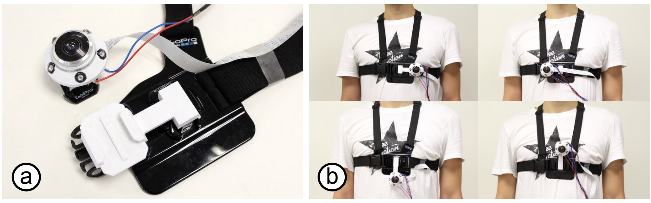

Figure 11. A detachable hinge was designed to offset the placement of the Cyclops device unfavorably.

be further reduced to between 2 and 3 hours. Since the computation of RDF classifiers mainly involves pixel operations, we believe that the computation efficiency can be further significantly enhanced by GPU acceleration.

As shown in Figure 10a, standard RDF classifier increases the recognition rates to nearly 90% for both motion signature representations. Figure 10bc shows the confusion matrices of recognizing workout gestures using dMHI and iMHI, respectively. The gesture classifier worked generally well for most classes. The cases of false recognition mainly involved some face-down and lying-down exercises, which are visually similar to their MHI representations and therefore difficult to be distinguished from each other. In the next evaluation, IMU orientation data will be utilized to deal with such gestures more effectively.

Performance of Multi-layered RDF Classification

Before training, the 20 workout exercises were first divided into three motion categories[5]. The IMU data associated with each category are then used to train the first-layer RDF. Specifically, the following parameters are adopted: number of trees $T = 3$, depth = 10, temporal length of orientation signatures $n = 150$, 100 candidate features, and 20 candidate thresholds per feature. Owing to its simplicity, the first-layer training can typically be completed in just a few seconds. The second-layer RDFs can be obtained just as the standard RDF except that the training data set is reduced for each category.

Unlike [11], which obtains depth estimates by aggregating results from both depth classification and regression forests, the gesture type is solely determined by the second-layer RDF in our system. Therefore, a highly accurate first-layer classifier is essential. Under the same leave-one-person-out cross validation setting, the first-layer RDF achieved a relatively high average recognition rate of 98%, indicating fairly successful coarse-grained classification. Moreover, we observed a slightly increased average recognition rate for the second-layer MHI-based classification, indicating that gesture classification by categories is simplified when compared with the full workout set. When combined, the two-layer classifiers achieved an overall recognition rate of 92.1% and 91.4%, corresponding to dMHI and iMHI respectively (shown in the green bars in Figure 10a).

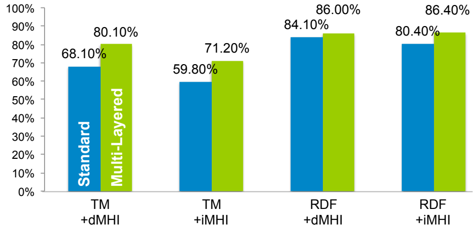

Figure 12. Recognition rates of various gesture recognition algorithms after offsets are applied to Cyclops. Blue/green bars indicate the performance obtained with/without IMU orientation information, respectively.

Applying Offset to Cyclops

In this experiment, the effect of applying an offset to the Cyclops device on recognition performance is investigated.

Test Data Acquisition

To acquire data under offset conditions, four detachable hinges (Figure 11) were fabricated; each introduced a 30-mm offset from the original location in the up, down, left or right direction. Ten participants were recruited from the original participant poolto perform the entire workout set once under all four offset conditions. The offset data sets were used as the test sequences, and the non-offset data sets of the same participants were used to train the RDF classifiers.

Evaluation Results

Figure 12 presents the effects of offsetting the placement of Cyclops on the human body. Since offsetting the Cyclops' camera compromises the characteristics of registration-free images, the fact that the performance of template matching was considerably degraded is unsurprising. RDF classifiers still achieved acceptable performance of above 80%. Notably, the IMU orientation signature is unaffected regardless of how the device is displaced. As a result, two-layered classification consistently improved the overall performance in all test cases.

EXAMPLE APPLICATIONS

The proposed proof-of-concept device is demonstrated with four applications - interactive body workout, mobile racing game, wearable VR gaming and tangible toy interaction.

Interactive Body Workout

Figure 13 shows the interactive body workout application. The user performs workout routines at home while Cyclops monitors progress and provides vocal instructions concerning subsequent actions. For example, Cyclops counts the sit-ups performed by the user; encourages the user to speed up or slow down, and, when the exercise is complete, (Figure 13 a-b), instructs the user to prepare for leg crunches (Figure 13c-f). Static and motion gesture recognitions are based on the iMHI+RDF method, as described in the Evaluation section.

Hand and Foot Interactions in Mobile Gaming

Cyclops enriches gaming on mobile devices by incorporating hand and foot interactions. Figure 14 presents an example of a racing game on mobile phones. The user holds the

[5] The orientation signatures of stationary gestures are similar to the other three types of motion gestures, respectively.

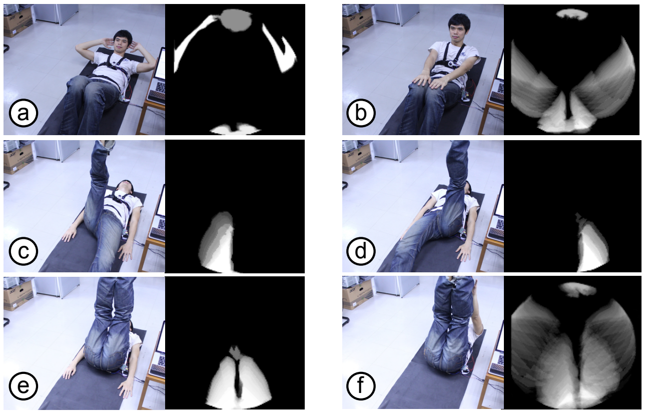

Figure 13. Interactive body workout applications track the user's performing workout exercises, and provide interactive instructions. (a-b) The workout tracker counts the sit-ups and provides suggestions to help the user maintain a good pace. (c-f) The tracker provides hints to the user concerning the next step in a four-step leg crunch exercise.

mobile phone as if holding an steering wheel, and the steering is monitored by the motion sensors of the phone. Moving the phone closer to the body results in an engaged view (Figure 14a), and pushing it away results in a second-person view (Figure 14b). Pushing the right-hand forward as if pushing forward a gear stick triggers nitro turbo (Figure 14c), and pushing the left leg out brakes the car.

The application is implemented with a VNC client on the mobile phone, which displays the racing game that is being run on a remote PC. The same remote PC processes Cyclops' video stream, performs gesture recognition using iMHI+RDF, and feeds back to the racing game with a keyboard simulator.

Wearable VR Snowball Fight Game

Recent developments in virtual reality (VR) have led to immersive gaming with wearable VR headsets such as Oculus Rift. However, most current interaction methods use gamepads, or allow limited body input using external image sensors but they still suffer from the line-of-sight problem.

This application of Cyclops enables the use of omnidirectional body motion gestures in immersive virtual reality gaming. Figure 15 presents a snowball fight game. The user, wearing Cyclops and an Oculus Rift VR headset, plays the game in the first-person view. In the game, enemies appear in all possible directions and throw snowballs toward the player. User interactions include throwing back snowballs by swinging one's arms (Figure 15a), throwing a giant snowball by swinging both arms (Figure 15b), and avoiding a ball by squatting or leaning to the left or the right (Figure 15cd).

The application is implemented with Unity3D and Oculus Rift VR headsets. Cyclops recognizes the player's motion gestures using dMHI+RDF and uses the actions to drive corresponding action scripts in the Unity3D game. The 9-DOF IMU of the Cyclops device tracks the player's orientation and leaning directions.

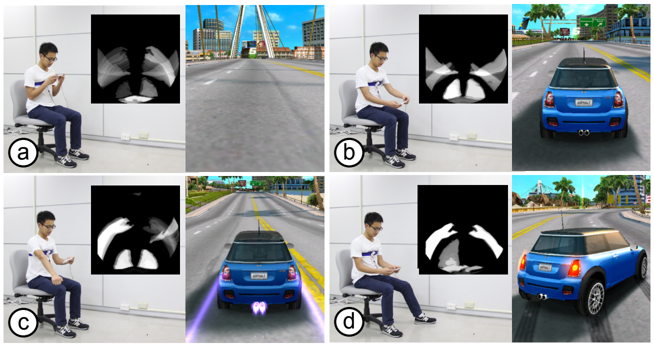

Figure 14. Hand and foot interaction in a mobile racing game. (a) Moving the phone closer to the body provides an engaged view. (b) Pushing the phone away enables a second-person view. Users can (c) push an imaginary gear stick on his right to trigger nitro turbo, and (d) step to the left to brake.

Tangible Toy Interaction

Attaching Cyclops to a stuffed toy turns the toy's body posture into an interactive controller. Figure 16 demonstrates the interaction with a display stuffed toy. A mobile display is embedded in the face of the toy. The Cyclops device tracks the motion gestures of the toy's limbs using iMHI+RDF and triggers facial expressions using the mobile display on the toy's face. Waving its hand causes the toy say 'Hi' with a smiling face (Figure 16a). Moving both of the toy's legs back and forward displays a running face (Figure 16b).

DISCUSSION AND LIMITATIONS

Cyclops is a single-piece wearable device that allows full-body posture inputs. We believe that the ultra-wide-angle view of users' body gestures will pave the way to a new generation of wearable motion capture devices. Therefore, this work seeks to establish the feasibility of this idea by demonstrating a proof-of-concept prototype. In the following, we discuss the main challenges and limitations faced by the current prototype system in two aspects.

Computer vision challenges

Like all existing vision-based projects, Cyclops faces the challenges that are typically faced by computer vision techniques, such as cluttered backgrounds and varying lighting conditions. To mitigate these challenges while validating the benefits of wearable devices that capture full-body gestures, Cyclops was implemented using an infra-red camera with active illumination, and the experiments were conducted in a controlled environment. In the future, Cyclops can be realized with advanced depth sensing techniques, such as time-of-flight distance estimation, to overcome the challenges.

Social acceptance by gender

Previous research [22] has demonstrated that social acceptance of wearable devices differs between genders. Female users were less accepting of Cyclops. In the pilot test, Cyclops inevitably yielded more occluded images owing to its ego-centric view. Female users tended to report feeling uncomfortable with putting the device on the chest. Currently, the prototype is relatively bulky. In the future, Cyclops will

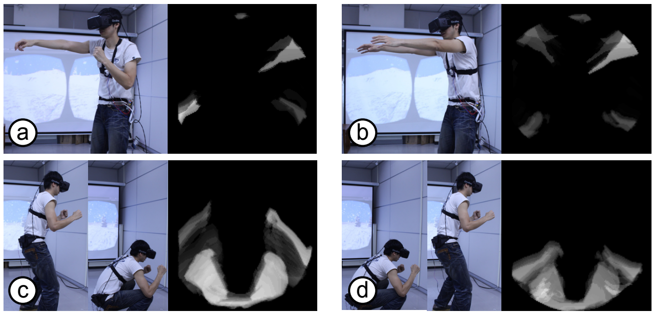

Figure 15. Cyclops supports omni-directional body motion gestures for immersive virtual reality. (a) The user flings a snowball by swinging his right arm; (b) throws a giant ball by swinging both arms, and (c)(d) avoids a ball by squatting or leaning to the left or to the right.

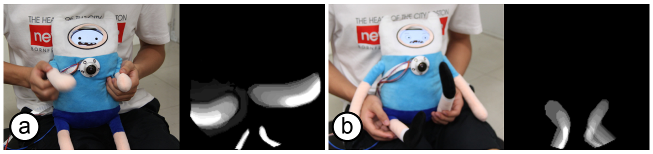

Figure 16. Attaching Cyclops to a stuffed toy enables a user to interact with the whole body of the toy. (a) The toy says 'Hi' with a smiling face when its hands are waved, and (b) displays a running face when its legs are made to run.

be made small enough to fit the wearable forms that are presented in Figure 2.

CONCLUSION

This paper presented a single-piece wearable motion-capture device. The use of Cyclops, as a proof-of-concept device, in full-body posture recognition, is demonstrated. The main contribution is the idea of determining body posture using an ego-centric perspective of the user, using only a single-piece motion-capture device.

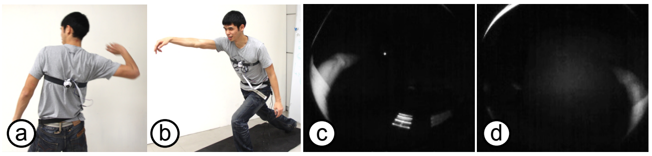

Figure 17. The strengths and limitations of Cyclops arise from its fisheye field-of-view. (a-b) Cyclops fails when it cannot see the motion of the body, such as in ball pitching. (c-d) This problem, however, can be solved by using another Cyclops that is worn on the back of the user.

Cyclops allows many body postures to be observed, owing to its fisheye field-of-view. Like other devices that use cameras, however, Cyclops yields uncertain results concerning postures in which limbs are out-of-sight. Figure 17 shows a user's ball-pitching posture. The pulling back of the pitching hand is invisible to Cyclops. This problem can be solved by adding another Cyclops to the user's back, as displayed in Figure 17. Future works will incorporate into Cyclops a pair of fisheye lenses in a wearable chest mount, yielding a blind spot-free wearable motion capture device.

ACKNOWLEDGMENTS

We gratefully acknowledge the helpful comments of the Associate Chairs and the anonymous reviewers. This paper was partially supported by National Science Council of Taiwan under NSC103-2218-E-002-014 and MOST103-2218-E-002-024-MY3, and also supported by Ministry of Science and Technology, National Taiwan University and Intel Corporation under Grants MOST102-2911-I-002-001, NTU103R7501, and NTU103R104130.

REFERENCES

1. Ahad, M. A. R., Tan, J. K., Kim, H., and Ishikawa, S. Motion history image: Its variants and applications. *Mach. Vision Appl. 23*, 2 (Mar. 2012), 255–281.
2. Bailly, G., Müller, J., Rohs, M., Wigdor, D., and Kratz, S. Shoesense: A new perspective on gestural interaction and wearable applications. In *Proc. ACM CHI '12* (2012), 1239–1248.
3. Bobick, A. F., and Davis, J. W. The recognition of human movement using temporal templates. *IEEE Trans. Pattern Anal. Mach. Intell. 23*, 3 (Mar. 2001), 257–267.
4. Bowman, D. A., Wingrave, C. A., Campbell, J. M., and Ly, V. Q. Using pinch gloves for both natural and abstract interaction. In *Proc. HCI International '01* (2001), 629–633.
5. Chan, L., Liang, R.-H., Tsai, M.-C., Cheng, K.-Y., Su, C.-H., Chen, M. Y., Cheng, W.-H., and Chen, B.-Y. Fingerpad: Private and subtle interaction using fingertips. In *Proc. ACM UIST '13* (2013), 255–260.
6. Chen, K.-Y., Lyons, K., White, S., and Patel, S. utrack: 3d input using two magnetic sensors. In *Proc. ACM UIST '13* (2013), 237–244.
7. Cohn, G., Morris, D., Patel, S., and Tan, D. Humantenna: Using the body as an antenna for real-time whole-body interaction. In *Proc. ACM CHI '12* (2012), 1901–1910.
8. Colaço, A., Kirmani, A., Yang, H. S., Gong, N.-W., Schmandt, C., and Goyal, V. K. Mime: Compact, low power 3d gesture sensing for interaction with head mounted displays. In *Proc. ACM UIST '13* (2013), 227–236.
9. Criminisi, A., and Shotton, J. *Decision Forests for Computer Vision and Medical Image Analysis*. Springer, 2013.
10. Dernbach, S., Das, B., Krishnan, N. C., Thomas, B., and Cook, D. Simple and complex activity recognition through smart phones. In *IEEE IE '02* (June 2012), 214–221.
11. Fanello, S. R., Keskin, C., Izadi, S., Kohli, P., Kim, D., Sweeney, D., Criminisi, A., Shotton, J., Kang, S. B., and Paek, T. Learning to be a depth camera for close-range human capture and interaction. *ACM Trans. Graph. 33*, 4 (July 2014), 86:1–86:11.

12. Gustafson, S., Bierwirth, D., and Baudisch, P. Imaginary interfaces: Spatial interaction with empty hands and without visual feedback. In *Proc. ACM UIST '10* (2010), 3–12.

13. Harrison, C., Benko, H., and Wilson, A. D. Omnitouch: Wearable multitouch interaction everywhere. In *Proc. ACM UIST '11* (2011), 441–450.

14. Harrison, C., Tan, D., and Morris, D. Skinput: Appropriating the body as an input surface. In *Proc. ACM CHI '10* (2010), 453–462.

15. Junker, H., Amft, O., Lukowicz, P., and Tröster, G. Gesture spotting with body-worn inertial sensors to detect user activities. *Pattern Recogn. 41*, 6 (June 2008), 2010–2024.

16. Keally, M., Zhou, G., Xing, G., Wu, J., and Pyles, A. Pbn: Towards practical activity recognition using smartphone-based body sensor networks. In *Proc. ACM SenSys '11* (2011), 246–259.

17. Kim, D., Hilliges, O., Izadi, S., Butler, A. D., Chen, J., Oikonomidis, I., and Olivier, P. Digits: Freehand 3d interactions anywhere using a wrist-worn gloveless sensor. In *Proc. ACM UIST '12* (2012), 167–176.

18. Kwapisz, J. R., Weiss, G. M., and Moore, S. A. Activity recognition using cell phone accelerometers. *SIGKDD Explor. Newsl. 12*, 2 (Mar. 2011), 74–82.

19. Lee, J., and Ha, I. Real-time motion capture for a human body using accelerometers. *Robotica 19*, 6 (Sept. 2001), 601–610.

20. Lin, S.-Y., Su, C.-H., Cheng, K.-Y., Liang, R.-H., Kuo, T.-H., and Chen, B.-Y. Pub - point upon body: Exploring eyes-free interaction and methods on an arm. In *Proc. ACM UIST '11* (2011), 481–488.

21. Lissermann, R., Huber, J., Hadjakos, A., and Mühlhäuser, M. Earput: Augmenting behind-the-ear devices for ear-based interaction. In *Proc. ACM CHI EA '13* (2013), 1323–1328.

22. Mayol-Cuevas, W. W., Tordoff, B. J., and Murray, D. W. On the choice and placement of wearable vision sensors. *Trans. Sys. Man Cyber. Part A 39*, 2 (Mar. 2009), 414–425.

23. Mistry, P., and Maes, P. Sixthsense: A wearable gestural interface. In *Proc. ACM SIGGRAPH ASIA '09 Sketches* (2009), 11:1–11:1.

24. Mistry, P., Maes, P., and Chang, L. Wuw - wear ur world: A wearable gestural interface. In *Proc. ACM CHI EA '09* (2009), 4111–4116.

25. Morris, D., Saponas, T. S., Guillory, A., and Kelner, I. Recofit: Using a wearable sensor to find, recognize, and count repetitive exercises. In *Proc. ACM CHI '14* (2014), 3225–3234.

26. Mujibiya, A., Cao, X., Tan, D. S., Morris, D., Patel, S. N., and Rekimoto, J. The sound of touch: On-body touch and gesture sensing based on transdermal ultrasound propagation. In *Proc. ACM ITS '13* (2013), 189–198.

27. Ogata, M., Sugiura, Y., Makino, Y., Inami, M., and Imai, M. Senskin: Adapting skin as a soft interface. In *Proc. ACM UIST '13* (2013), 539–544.

28. Rekimoto, J. Gesturewrist and gesturepad: Unobtrusive wearable interaction devices. In *Proc. IEEE ISWC '01* (2001), 21–.

29. Saponas, T. S., Tan, D. S., Morris, D., Balakrishnan, R., Turner, J., and Landay, J. A. Enabling always-available input with muscle-computer interfaces. In *Proc. ACM UIST '09* (2009), 167–176.

30. Sato, M., Poupyrev, I., and Harrison, C. Touché: Enhancing touch interaction on humans, screens, liquids, and everyday objects. In *Proc. ACM CHI '12* (2012), 483–492.

31. Scott, J., Dearman, D., Yatani, K., and Truong, K. N. Sensing foot gestures from the pocket. In *Proc. ACM UIST '10* (2010), 199–208.

32. Shiratori, T., Park, H. S., Sigal, L., Sheikh, Y., and Hodgins, J. K. Motion capture from body-mounted cameras. In *Proc. ACM SIGGRAPH '11* (2011), 31:1–31:10.

33. Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., and Blake, A. Real-time human pose recognition in parts from single depth images. In *Proc. IEEE CVPR '11* (2011), 1297–1304.

34. Song, J., Sörös, G., Pece, F., Fanello, S. R., Izadi, S., Keskin, C., and Hilliges, O. In-air gestures around unmodified mobile devices. In *Proc. ACM UIST '14* (2014), 319–329.

35. Sturman, D. J., and Zeltzer, D. A survey of glove-based input. *IEEE Comput. Graph. Appl. 14*, 1 (Jan. 1994), 30–39.

36. Tamaki, E., Miyaki, T., and Rekimoto, J. Brainy hand: An ear-worn hand gesture interaction device. In *Proc. ACM CHI EA '09* (2009), 4255–4260.

37. Taylor, S., Keskin, C., Hilliges, O., Izadi, S., and Helmes, J. Type-hover-swipe in 96 bytes: A motion sensing mechanical keyboard. In *Proc. ACM CHI '14* (2014), 1695–1704.

38. Vardy, A., Robinson, J., and Cheng, L.-T. The wristcam as input device. In *Proc. IEEE ISWC '99* (1999), 199–202.