

On-Body Interaction: Armed and Dangerous

Chris Harrison^{1,2} Shilpa Ramamurthy³ Scott E. Hudson^{1,2}

¹Human-Computer Interaction Institute, ²Heinz College Center for the Future of Work, ³Computer Science Department
Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh PA 15213
chris.harrison@cs.cmu.edu, shilpar@andrew.cmu.edu, scott.hudson@cs.cmu.edu

ABSTRACT

Recent technological advances in input sensing, as well as ultra-small projectors, have opened up new opportunities for interaction – the use of the body itself as both an input and output platform. Such *on-body interfaces* offer new interactive possibilities, and the promise of access to computation, communication and information literally in the palm of our hands. The unique context of on-body interaction allows us to take advantage of extra dimensions of input our bodies naturally afford us. In this paper, we consider how the arms and hands can be used to enhance on-body interactions, which is typically finger input centric. To explore this opportunity, we developed *Armura*, a novel interactive on-body system, supporting both input and graphical output. Using this platform as a vehicle for exploration, we prototyped many applications and interactions. This helped to confirm chief use modalities, identify fruitful interaction approaches, and in general, better understand how interfaces operate on the body. We highlight the most compelling techniques we uncovered. Further, this paper is the first to consider and prototype how conventional interaction issues, such as cursor control and clutching, apply to the on-body domain. Finally, we bring to light several new and unique interaction techniques.

ACM Classification: H.5.2 [Information interfaces and presentation]: User Interfaces - Graphical user interfaces; Input devices and strategies.

Keywords: Computer vision, projectors, depth camera, vision-based input, free-space gestures, sensing, Arumura.

INTRODUCTION

Very recently, there has been a perfect storm of technological advances that has culminated in the emergence of a new interaction modality: *on-body interfaces* [16,17,32,38]. That is, using the human body as both an input and output platform. The first of these advances is the incredible computational power now able to be wielded in small, mobile devices. Despite their diminutive size, they are able to perform tasks of creation, communication and computation with unprecedented ease. The emergence of pico-projectors has

also been influential. Recently released laser pico-projectors can render graphics at different depths and onto irregular surfaces without loss of focus, enabling placement, for example, at oblique angles. This capability, along with their extreme portability, has enabled mobile on-body projection [16,17,26].

The third ingredient is vision-based tracking. Systems have, for some time, offered the possibility of tracking the human body and recognizing gestures [3,7,11,24,35,40,45]. Importantly, this can be achieved without the need to directly instrument the user with sensors or tracking elements (e.g., data gloves [36] or sensor-embedded fabrics [30]). More recently, using e.g., structured light or time-of-flight sensing, depth cameras can acquire both image data and a depth map. This alleviates segmentation challenges and further increases sensing fidelity and recognition capabilities [16, 18]. For example, Microsoft’s Kinect Xbox accessory, costing \$150 retail, captures an RGB+Z image at 30FPS. Not only can these sensors be deployed in a fixed setup (e.g., conference room) [43], but are now sufficiently compact that they can be worn on the body [16].

The availability of these three elements opens the door to a range of possible *on-body interfaces*. In this paper, we consider how the arms (and hands) can be used to expand the interactive power and design space. Although most computing interfaces are primarily finger-driven, the unique context of *on-body interaction* allows us to take advantage of extra degrees of freedom our bodies naturally afford us.

The contributions of this work are three fold. Foremost, we developed a novel real-time system called *Armura*, which can segment, track, and classify a variety of arm and hand gestures. Our system can also project graphical elements onto the user, allowing for interactive on-body applications. Secondly, we used *Armura* to identify key use modalities, powerful interactions, and potential applications. Our explorations revealed that several “desktop class” interactions are portable to and relevant in the on-body domain. Further, several new interaction techniques were discovered, many unique to on-body computing.

THE ARMS AS AN I/O PLATFORM

As the old colloquialism “like the back of your hand” suggests, we are intimately familiar with our own bodies. Indeed, it is the only “device” we receive training with from birth and every waking moment thereafter. Because of this, we are incredibly dexterous; our kinesthetic senses allow us to rapidly and accurately position our body and limbs, even

Copyright © 2012 by the Association for Computing Machinery, Inc. Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions Dept, ACM Inc., fax +1 (212) 869-0481 or e-mail permissions@acm.org.

TEI 2012, Kingston, Ontario, Canada, February 19 – 22, 2012.
© 2012 ACM 978-1-4503-1174-8/12/0002 \$10.00

with our eyes closed. We also develop finely tuned muscle memory and hand-eye coordination, all of which can be immediately leveraged in digital interaction tasks [8,21].

Consider, for a moment, the arm as a computing I/O device. Unlike all other devices we use, we cannot choose to modify its form – we can augment it, but not change it. This simultaneously means its capabilities are fixed – humans (in general) are a known quantity. For example, there are only so many ways we can move our arms, hand and fingers – defined by set of musculoskeletal constraints. Additionally, we only have so much surface area for graphical output. Since the I/O capabilities are somewhat predefined for us, the question then becomes, what can we do with the arms?

The sum expressive capabilities of the arms are enormous. The shoulder and elbow allow the hands to be translated and rotated in 3D space (six degrees of freedom). The wrist provides a separate mechanism for rotating the hand on two axes. Each finger has one major and two minor knuckles, which allow them to bend and have limited rotation on two axes. Our thumbs, in contrast, have a much greater rotational range and can also bend. In total, there are more than twenty independent degrees of freedom. Combinatorial speaking, our fingers are capable of forming hundreds of poses [23,28]. For reference, American Sign Language (which includes motion) has several thousand signs [39].

We choose to focus on the arms over e.g., the legs or torso, because they serve as our chief appendages for manipulating the physical world. The arms are also an ideal ad hoc display platform given their ready availability and proximity to the head. The skin provides a natural and immediate surface for dynamic digital projection [16,17,26]. Although skin introduces some color distortion, the resolution, frame rate, and overall quality can be high (even on darker skin). Each hand, although small, provides a projection area comparable to that of a mobile device. Line of sight is required, but in practice, is easily accommodated when the arms are active and in front of the user. Furthermore, laser projection (which is focus-free) could one day allow for highly oblique projection, e.g., from a bracelet. Although we focus on graphical output in this work, as that is the predominant output means for computing, on-body interfaces could also readily incorporate auditory or haptic feedback.

RELATED WORK

A key component to the success of mobile computing has been reduction of interaction barriers relative to desktop-class systems. One is able to rapidly take out their smart phone, look up something on Wikipedia, and settle a dinner debate with incredible ease. The larger vision of *on-body* computing is to reduce this interactive viscosity again, by literarily having computing power in the "palm of your hand" and always available [16,17,32,38]. Using current technology, it is already possible to summon capabilities similar to that of a touch screen [16]. It is no longer unfathomable that our hands could serve as a smartphone interface, supported by a button-less and screen-less device containing both sensing and projection capabilities.

Arm-driven and hand-driven input has received attention for decades, and scores of advanced systems are able to detect, track and recognize limbs for a variety of purposes [3,7,11,19,24,35,42,45]. Although many projects have employed handheld projectors [5,6,9,10,31,41], few have taken advantage of the arms as the projection surface [4,29,37,45]. Rarest and most recent, however, are projects that attempt both input and output on the body. It is this unique combination that defines on-body interaction, which is significantly different in operation and experience than gesturing to a remote interface. We now briefly discuss the four most notable projects and their input capabilities.

SixthSense

SixthSense [26] is a proposed pendant-like device, containing a camera and pico-projector (which could communicate to a phone in the pocket or be self contained). Through computer vision, the authors suggest that different objects in the environment could be recognized and fingers and hands (with color markers) tracked for input and gestures. SixthSense touches on a variety of projected interactions, although mostly onto the environment (e.g., walls and objects). Proposed on-arm examples include the dialing of a phone with the fingers and summoning a watch on the wrist with a finger circling motion. Additionally, several static gestures are suggested, including a finger "square" for capturing a photograph.

Skinput

Through bio-acoustic fingerprinting, Skinput can detect and localize finger taps on the skin [17]. An armband contains an array of sensors and a mounted pico-projector, which provides dynamic graphical output on the forearm and hand. Skinput primarily enables finger "click" driven interaction, which can be used to "press" one of many discrete graphical buttons. Although lacking any dimension of continuous input or gestures, a wide variety of interactions are possible. Highlights include menu navigation and playing real-time games, such as Tetris and Frogger.

OmniTouch

Most recent is OmniTouch [16], a shoulder-worn, depth-sensing and projection system that enables interactive multi-touch applications on everyday surfaces, including the body. The system provides capabilities similar to that of a touchscreen: X and Y location of fingers in 2D interfaces and whether they are "clicked" or hovering. Additionally, interaction surfaces are tracked in 3D space such that projected output can be rectified, allowing for graphical interfaces with minimal distortion (e.g., on oblique surfaces). Demo applications include a phone number dialer, multi-touch painting, scrolling menu-based music player, QWERTY keyboard, and map panning/zooming.

LightSpace

There is also great value in fixed infrastructure that can augment spaces with on-body capabilities. This allows any user occupying the room to utilize the features and also eliminates the need for devices to be worn. LightSpace [43] is one such system, utilizing a fixed array of depth cameras

and overhead projectors to augment a room. Interactions enabled by this system are diverse; relevant to this paper is an on-hand “spatial menu”. Users can move their hands over a specific “menu” location, and then move their hands in the Z-axis (up/down) to select from various menu items. Selection is achieved by dwelling for a brief period. The authors also experimented with virtual items, which can be “held” in the hands, and then transferred to other projected surfaces in the environment.

ARMURA

To explore how the arms and hands can be used to supplement on-body interactions, we built *Armura*. Our system is a combination of hardware and software that enables real-time, on-body, graphical interaction. In many ways, it is a realization and extension of the interactions described in SixthSense, Skinput, OmniTouch and LightSpace. As we will see, *Armura* also advances on-body interaction, offering capabilities not seen in previous systems. As such, it provides a unique vehicle for exploration, allowing us to consider and develop several novel arm-driven interfaces.

The chief feature of our system is the ability to track the location of the arms and hands, as well as recognize their gestural state, in real-time. Additionally, we can simultaneously project coordinated graphical feedback onto the body. We primarily consider a motion- and gesture- driven approach supported by computer vision. On-body input could, however, use a variety of sensing technologies, including bio-acoustics [17], electric field sensing [34] and electromyography (EMG) [33], which provide other dimensions.

Our prototype hardware consists of a ceiling-mounted DLP projector. A mirror is used to project downwards and also expand the interactive area. An infrared camera featuring a wide-angle lens is affixed to this setup, also facing downwards. It is capable of sampling a 640x480 image at 60 FPS. Three infrared illuminators, set apart in a triangle configuration, help to increase contrast (skin is reflective to infrared).

On top of this hardware, we developed a custom Java-based application for detecting, tracking and recognizing arms, hands, and gestures. The heart of the software is a hybrid feature- and template-based gesture recognizer driven by a support vector machine classifier provided by the Weka Machine Learning Toolkit [44]. Our system can identify when a user enters the field of view of the camera, the location of the arms if raised (X and Y; Z can be estimated using hand size, but requires calibration – a moot point with depth cameras), and the state (gesture) of the arms and hands. The later is supported by two independent classifiers, one for

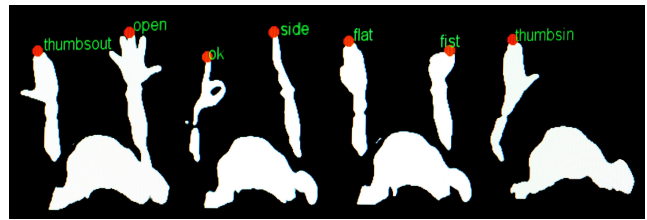


Figure 1. Four users, shown from above, following the segmentation process. Each hand demonstrates one of seven gestures our exploratory system supports.

arm configurations and another for hand gestures. It is assumed the arms and hands are always holding some gesture, even if just flat or loose (e.g., arms by the side, open palm). Rectification between camera and projector space is achieved with a four-point projective transformation.

The screenshot seen in Figure 1 illustrates the result of our user segmentation process. Also pictured are the seven hand gestures our proof-of-concept system supports. The red dot is the estimated top of the hand, used for X/Y position. Although not a comprehensive gesture set, designing around a small range of motions and gestures can also have human advantages in terms of memory and attention load [20], and overall is a good place to begin explorations.

Armura also tracks and classifies nine synergistic “arm-level” gestures, seen in Figure 2. Importantly, hand gestures can only be inferred when the hands are visible and distinct (i.e., not touching other elements). X/Y location is produced using a bounding box of the user (red outline in Figure 2).

In practice, our system performs well, tracking at interactive speeds (see Video Figure) and achieving acceptable classification accuracies (during piloting, ten-fold cross-validation accuracies were in excess of 96% with our seven hand gestures). This proved more than adequate for our exploration of the design space. Most importantly, it served as a good “in-lab” analog (that could be built today) for more complex and/or future technologies that can operate on the go (e.g., with depth cameras and pico-projectors).

CHIEF USE MODALITIES

Over the course of *Armura*’s development, as well as iterative development on a suite of demonstration applications, four chief use modalities became apparent.

Single Arm

The simplest use modality is a single arm (and hand). Although input and output is heavily constrained, it has the important property of leaving the other arm entirely free. This allows the user to engage in another independent task,

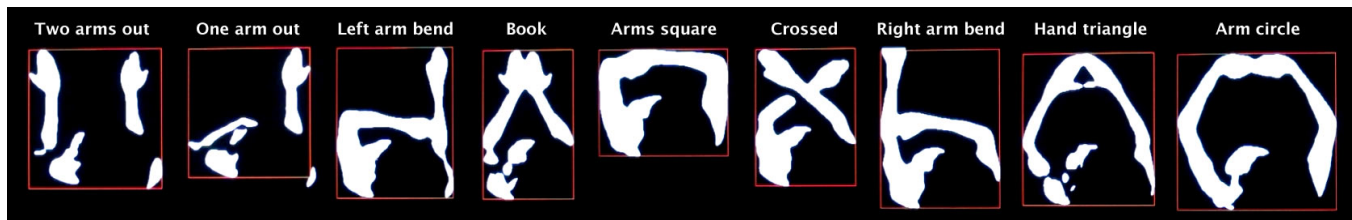


Figure 2. Our exploratory system can also recognize nine synergistic arm gestures.



Figure 3. Two hands can create a book interaction metaphor. “Pages” can be turned by flipping the appropriate hand, which transitions the interface. Time advances from left to right.

such as writing or taking a phone call. It could prove especially useful for accessing information related to the other task, such as looking at ones calendar, getting directions, or checking the time. Anecdotally, the elbows tucked in, hands held in front, and palms up appeared most intuitive to users, and also comfortable (i.e., reduced gorilla arm).

Single Arm + Other Arm for Display

In this modality, one hand is used for input, while the other hand is used for graphics. The input capability of this modality is identical to that of *Single Arm*, since still only one arm is providing input. The graphical output is simply projected on the “display” arm. This simple change, however, has several beneficial implications.

Foremost, the visual content can now be displayed on a less-dynamic hand, which is easier to view. Not only can this hand be positioned to provide a superior view of the content, but also located in a more comfortable and sustainable position, in case of extended interaction. Equally important, the user no longer has to look at their hand driving the input. Thus, it is free to operate more quickly, at greater distances from the user, and even out of the user’s visual field. In our experiences, this modality is the most comfortable and intuitive to use. In many ways, the input hand operates “off to the side” like a mouse, requiring no visual attention to control. The eyes can remain fixed on the interactive graphical output, which feels very much like a handheld screen.

Two Single Arms

The previous two modalities rely on the expressive power of a single arm for input. Using two arms for input doubles the expressive power (e.g., 2×3 DOF arms = 6 DOF). Note that we do not call this modality “two arms”. Instead we prefer the term “two single arms”, which stresses their individuality. In particular, they simply operate in parallel, gaining no new joint input dimensions. In the next section, we will consider how two arms can be used in conjunction, yielding more synergistic uses.

Bimanual manipulation has been shown to be extremely powerful and intuitive [8,14,19,21,22]. Having extra degrees of freedom available also enables more sophisticated interactions and also provides additional graphical space if needed. The classic 4 DOF action is simultaneously scaling and rotating an image – typically demonstrated on multi-touch surfaces with a two finger “pinch” gesture. This action could be achieved on-body using two hands and a “grasping” gesture. A more practical example might be two-dimensional panning of map with one hand, while the se-

cond hand controls the zoom level and other modal functions, such as a traffic overlay.

Synergistic Use of Both Arms

Our final modality is the use of both arms in a synergistic fashion. They no longer operate as two single arms, but rather as a new, combined entity. Synergistic configurations, which intersect the arms or hands at fixed points, have the potential to reduce the degrees of freedom. For example, in our *arms square* gesture, the arms become rigidly fixed to one another, naturally limiting the degrees of freedom. This reduction in expressiveness of the input space makes synergistic gestures exceptionally well suited for high-level mode switching. Anecdotally, this appears to match peoples’ intuition - initiating an interaction with a coarse arm-level gesture, and then switching to the more dexterous abilities of the hands (and fingers) for manipulation within that state.

Motion constraints in synergistic configurations have some interesting design implications. For example, the *crossed* gesture (Figure 2), defined by the intersection of the two arms at their midpoints, allows for 1 DOF scissor-like movement. The *arm circle* gesture has the interesting property of expansion in one dimension causing the contraction of the other dimension. However, the *book* gesture, which joins the two hands, retains all three axes of motion, as well a unique “page turning” action (Figure 3). Scores of other synergistic examples exist, many with interesting and potentially powerful innate motion constraints.

APPLICATIONS AND INTERACTIONS

A central objective of this work was to explore the design space of arm- and hand-driven interaction. Put plainly: if you have two arms, what on-body interactions are possible?

To explore this area, we drew inspiration from three distinct sources. First and most straightforward, was to look at the four chief existing on-body systems (see related work) and how the arms and hands were utilized in the proposed interactions. Secondly, we identified promising interactions and metaphors found in contemporary computing environments. Finally, the iterative rapid development of many small applications that run on top of *Armura* also proved invaluable - we discuss the most interesting of these. Additionally, these demos also serve to illustrate 1) the immediate feasibility and potential of on-body computing, 2) the fact our system is functional and operates in real-time, and 3) the impressive scope of the surrounding design space.

As noted previously, our system is position- and gesture-centric. It does not capture all possible dimensions of on-

body input (e.g., bend angle of the arms) – this full level of richness will slowly become available with future advances in processing and sensors. However, importantly, our system does afford us the ability to explore continuous and discrete input approaches, as well as interactions that take advantage of both simultaneously. Finally, the interactions we describe are not intended to preclude or replace finger input. Indeed, they can be used in concert with fingers to enhance the overall on-body experience.

Presentation Only

The very simplest interaction is to use the arms as a projection surface with no input capability. That is, the location of the arms adds no expressive power to the interaction - they are simply tracked so as to provide a projection surface. This could be used, to e.g., project a map of a user’s surroundings, which could prove useful to visitors of museums, hospitals and similar. The system could also display context-sensitive information, such as the time, upcoming meetings, and office directories – all of which could be valuable and shown without the need for input.

Modal Gestures

With even the addition of a single gesture (beyond the required presentation gesture for activation), the interaction space expands significantly. In general, this enables a user to fire an event. If the user does not wish to activate this option, the arms can simply be lowered (i.e., acting as a cancel or escape). For example, a *fist* hand gesture could cycle between different modes - each opening of the palm would reveal a different application, e.g., clock, calendar, stocks. Or if running late for a meeting, an *OK* hand gesture could trigger an apologetic email.

With multiple gestures, the design space is very large and the uses diverse. Different gestures could trigger different modes. For example, a bent “watch-on-wrist” arm gesture could render the time on the wrist. Forming a *hand triangle* gesture and moving it up or down could control the dimming of lights in a room. Gestures are also well suited to instantiation of interaction; for example, using a *two arms out* gesture to “summon” interactive capability. To end an interaction, or navigate backwards through a menu, an *arm cross* gesture could be used.

Modal Layout and Control

A special use of gestures (either hands, arms, or synergistic) is to select a preferred layout and control mechanism for an interactive experience. As discussed in *Synergistic Uses of Both Arms*, some configurations have innate motion constraints, providing natural dimensions for input.

Consider the simple example of an office directory. Presenting one arm might render a compact list of names and offices on the hand, which can be scrolled up and down by translating the arm. When the user presents two hands, the secondary hand might display a headshot and bio of the currently selected individual in the list (rendered on the primarily hand). As an alternative to list navigation, a *book* metaphor could be used, allowing the user to turn “page by



Figure 4. Moving the hand to different positions reveals what buildings lay in that direction.

page” (Figure 3) through the directory, or simply to provide more contiguous surface area for projection.

Position Considerations

In addition to gestures, which are discrete, *Armura* can track the arms in free space – giving us three dimensions of continuous input. There are four important ways this positional information can be calculated for on-body systems.

First, the body can serve as an anchor point, making positions relative with respect to e.g., the user’s head [19,26,27]. This has the potential to leverage muscle memory, if for example, we know our calendar is accessible to the left of us or the third menu item is roughly a foot in front of us [25]. Second, one arm can provide a reference point, from which the location of the second arm can be derived (see e.g., [13]); Third, movement can be tracked relative to the arm or hand’s entry position or state change (e.g. switching from a *flat* to *side* hand gesture). Lastly, as shown in Light-Space [43], the interface can be made relative to the environment – a hand over a desk might have a different function than one over the floor.

Modal Positions

If desired, position can be used for modal input. This was the basis for [25], which laid out mobile device functions in space around the user. Looking again at the very simplest of interactions, we could imagine a navigation application that displays what lies ahead based on how the hand is positioned relative to the user or environment. For example, directions to different buildings could be shown when the hand is on the left, right and in front of the user (Figure 4). This requires no further interaction; the user simply drops their hand when finished.

Menuing

Navigation of “menus” is core to the modern computing experience, and allows for graphical browsing and selection of desired functionality. Through the use of gestures, positional tracking, or a combination of the two, on-body computing can readily support intuitive menu navigation, and thus enables a wide variety of interactive applications.

Gestures

Simple menuing can be supported though a small vocabulary of gestures. For example, a yes/no dialog could be answered by flipping the hand palm-side-up or down (Figure 5). With as little as three gestures, navigation of hierarchical menus becomes possible (although not practical for long lists). Indeed, a simple vocabulary of arm and hand gestures

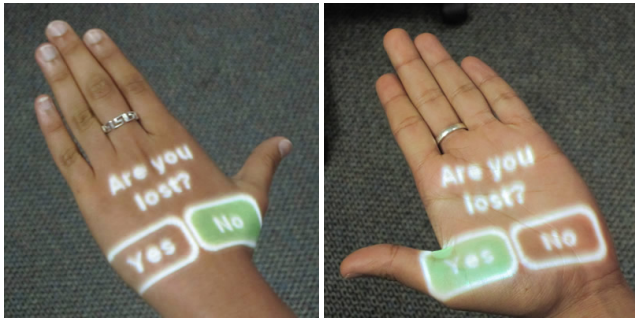


Figure 5. A user can toggle between yes and no by flipping their hand (*thumb in and thumb out* gestures respectively), pointing their thumb at the desired target. To select, the thumb is tucked in (*flat* gesture).

working in concert could provide sufficient richness for a simple, but powerful on-body personal organizer.

Position Tracking

A single axis of movement could be used for continuous manipulation – to, for instance, scroll an arm-projected office directory. As in previous examples, when the interaction is complete, the user can simply drop their arms.

Two or more axes of positional data enable unique interactions. For instance, we can use one dimension for modal control and another for manipulation. As an example, we built a music player interface with five modes: seek, volume, next song, previous song, pause/play. The modes were traversed by moving the hand from left to right. The song position and volume level were manipulated by the relative forwards/backwards motion of the hand (Figure 6). The three binary actions (next song, previous song, pause/play) could be activated by briefly rocking the hand forwards.

Gestures and Positional Tracking

Unsurprisingly, combining gestures and position enables the richest and most intuitive set of interactions. Even a single degree of positional freedom in concert with one “action” gesture allows users to navigate hierarchical list and menus in a practical manner (users can traverse up the hierarchy by using “back” menu items, see Figure 7 left). This immediately enables rich applications, such as an address book, calendar, or music player. With one additional gesture, the “back” menu items become unnecessary; a dedicated gesture can be used instead.

Crossing Gestures

Skinput and OmniTouch were primarily finger “click” driven, necessary for driving their button-centric interfaces. It is also possible to use a specific gesture to serve as an analog to a mouse-down action, with pinching appearing most popular [13,26,42]. However, it is possible to support on-body selection without “clicks” or discrete gestures though the use of crossing gestures [1,2]. As seen in Figure 8, a list of actions is rendered on the arm; selection occurs by swiping the free arm through a desired target.

Defining Axes of Input

The arms and hands can also be synergistically used as reference points, bounding features, or to delineate an area for

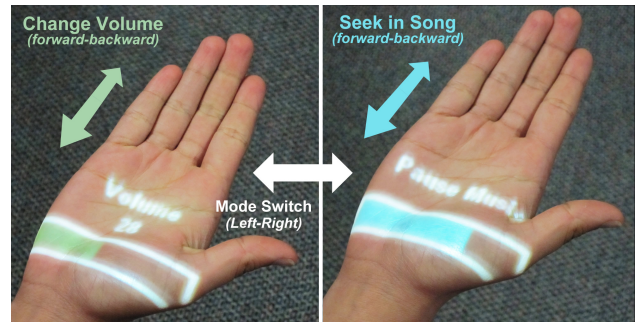


Figure 6. Moving the hand left-to-right toggles between different audio modes (X-axis). The level can be manipulated by rocking the hand forwards and backwards (Y-axis).

interaction. Such a method is described in [13], which uses an ‘L’ gesture to provide a two-dimensional plane for drawing in free space. The interaction illustrated in Figure 8 offers another example: an extended arm is used to provide an intuitive linear axis of input movement, which also serves as a large graphical canvas for navigating a list. The other hand can move along this axis for selection.

Cursor Control

As demonstrated by the venerable mouse, X/Y movement and a few buttons, combined with appropriate graphical feedback, enables endless interaction possibilities. The hands can be readily digitized to provide such functionality, enabling control of on-body “point-and-click” interfaces.

To demonstrate cursor control, but in a more fitting domain, we built a digital tattoo painting application (Figure 7, right). The left arm serves as the canvas (no input, tracking only). The right hand controls three modes. The first is the nominal mode, where movement of the right hand correspondingly moves a brush cursor on the opposing arm (using a control-device gain of roughly 1:5 for precision painting). If the user forms a “brush holding” gesture (Figure 1, *OK*), the cursor paints as it moves. Finally, closing the thumb (Figure 1, *flat*) on the right hand switches to brush selection mode. The user can select a paint color by moving their hands forward and backwards, while brush thickness is controlled by left-and-right motion. Visual feedback is provided with an enlarged “brush preview” rendered on the right hand.

Peephole / Spatially-Aware Displays

The arms, and especially the hands, can be used like a peephole or lens [12,15,46]. This has been shown to be a power-



Figure 7. Left: Hierarchical menu navigation on the hand. Right: demonstration tattoo painting application.

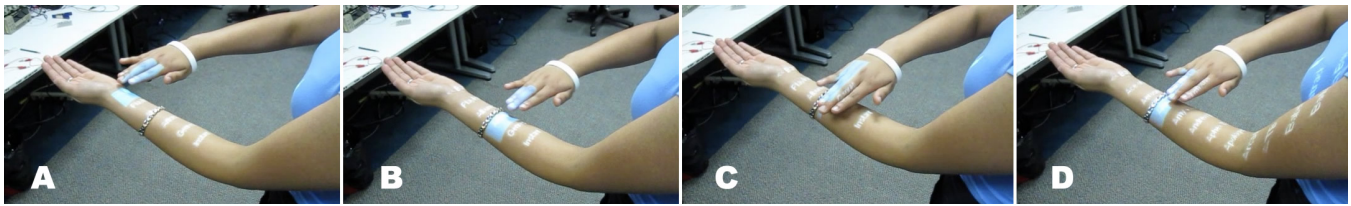


Figure 8. A linear menu rendered on the left arm. Sliding the right arm forwards and backwards along this axis changes the selection (A & B, blue highlight). Selection is performed with a crossing gesture (C), which advances the interface (D).

ful metaphor for small displays - for all practical purposes, we can treat the hand as a small display. Jumping back to an earlier example, peephole displays could allow users to not only see a map of their current surroundings, but also move their hands to view other areas (Figure 9).

As a spatially-aware display, the selection of an appropriate anchor point is important (see *Position Considerations*). Using, for example, the body for relative positioning [27], could allow the hands to show an “X-ray view” inside our bodies. Or, if anchored to the environment or an object [43], such as a map, the hand could act as an additional layer, providing supplementary geo-spatial information.

Control-Device Gain and Clutching

Unlike, e.g., a scroll wheel on a mouse, the arms have a maximum pan-able distance: one arm’s length. To overcome reach limitations that occur with cursor control and peephole displays, we experimented with two ways to expand the interactive space.

First we tried manipulating the control-device (CD) gain (with the device being one’s own hand). This appeared to work conceptually for an “on-body cursor” like we used in our tattoo application. However, with peephole displays, the effect was somewhat disorienting. For example, if seeing an object on the periphery of their hand, there was a tendency to move the hand to that spot so as to center that item – as they would do when manipulating the real world (which is inherently 1:1). However, a higher CD gain broke this nice physical metaphor, and anecdotally caused overshooting. Overall, we found a CD gain of 1:1 to be most natural and intuitive for peephole displays. In other words, when the hand is moved 5 cm, the graphics also track 5 cm. Our second approach is clutching via hand gestures. This effectively expands the interactive space and can also preserve a 1:1 CD gain if desired.

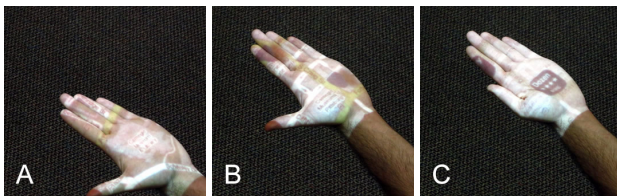


Figure 9. The hand can be used like a peephole display. Translating the hand in two-dimensions also translates the map view (A to B). “Clicking” the thumb (C) switches to an alternate layer; in this example, restaurant reviews.

We built a simple map application to help users locate nearby restaurants, navigated by moving a hand in the X/Y plane (1:1 CD gain). For clutching, we used a *fist* gesture, a natural choice for a “grab” action. The result feels very much like how one would manipulate a large map on a table. When a restaurant of interest is located, the thumb can be tucked in to see metadata (e.g., star rating; Figure 9).

FUTURE WORK

Since work in the area of on-body interaction has only recently begun, there are many avenues for future work. These include exploration of the properties of new sensors and overall sensing modalities. Further, it is not known how much of the full dexterity of arms, hands and fingers can be reliably recognized and how much of it can be usefully brought into an interactive system. There are also ergonomic issues, such as flexibility and fatigue. A more immediate step is to delve deeper into the many issues and avenues uncovered in this work, such as on-body clutching, supported by rigorous user studies. Topics range from the commonplace (how does proprioception affect Fitts’ law performance?) to the amusing (how do people feel about getting their buttons pushed?).

Finally, if one briefly considers the enormous volume of work dedicated to input for small mobile devices, it is not hard to imagine the human form will spark just as much, if not more research. Indeed, the human form is as dramatic a change as the jump from desktop to handheld computing. However, unlike devices of our creation, our bodies have no API and we come in some 6 billion different models. Overall, we hope this paper inspires people to work in this budding domain.

CONCLUSION

We have provided a preliminary exploration of the enormous interaction space of on-body computing. Indeed, not only are conventional “desktop” interactions such as cursor control and crossing gestures relevant, but also new avenues, such as innate motion constraints and dynamic modal layout. The latter are enabled by our unique physiology, which, unlike conventional computing devices, is organic and dynamic. This poses serious sensing and processing challenges, but also rich opportunities for powerful interactions. We also detail *Armura*, our prototype system on which we explored and developed the many applications and interactions described in this work. It also serves to advance the state of the art in on-body interaction, bringing many proposed interactions to reality.

ACKNOWLEDGMENTS

This work was funded in part by a Microsoft Ph.D. Fellowship, and Carnegie Mellon University's SRC Undergraduate Research Opportunities program and the Center for the Future of Work, Heinz College.

REFERENCES

1. Accot, J. and Zhai, S. More than dotting the i's -- foundations for crossing-based interfaces. In *Proc. CHI '02*. 73-80.
2. Apitz, G. and Guimbretière, F. CrossY: a crossing-based drawing application. In *Proc. UIST '04*. 3-12.
3. Argyros, A.A., and Lourakis, M.I.A. Vision-based Interpretation of Hand Gestures for Remote Control of a Computer Mouse. In *Proc. ECCV 2006 Workshop on Computer Vision in HCI*. LNCS 3979. 40-51.
4. Barnett, A. The dancing body as a screen: Synchronizing projected motion graphics onto the human form in contemporary dance. *Comput. Entert.* 7, 1 (2009), 1-32.
5. Beardsley, P., Baar, J. V., Raskar, R. and Forlines, C. Interaction using a handheld projector. *IEEE Computer Graphics and Applications*, 25, 1 (2005), 39-43.
6. Blasko, G., Feiner, S. and Coriand, F. Exploring interaction with a simulated wrist-worn projection display. *Proc. ISWC '09*. 2-9.
7. Bolt, R. A. and Herranz, E. 1992. Two-handed gesture in multimodal natural dialog. In *Proc. UIST '92*. 7-14.
8. Buxton, W., and Myers, B. A study in two-handed input. In *Proc. CHI '86*. 321-326.
9. Cao, X. and Balakrishnan, R. Interacting with dynamically defined information spaces using a handheld projector and a pen. In *Proc. UIST '06*. 225-234.
10. Cao, X., Forlines, C. and Balakrishnan, R. Multi-user interaction using handheld projectors. In *Proc. UIST '07*. 43-52.
11. Erol, A., Bebis, G., Nicolescu, M., Boyle, R.D., and Twombly, X. Vision-based hand pose estimation: A review. *Computer Vision and Image Understanding*. 108 (2007), 52-73.
12. Fitzmaurice, G.W. Situated Information Spaces and Spatially Aware Palmtop Computers. *Comm. of the ACM*, 36,7 (1993), 38-49.
13. Gustafson, S., Bierwirth, D., and Baudisch, P. Imaginary Interfaces: Spatial Interaction with Empty Hands and Without Visual Feedback. In *Proc. UIST '10*. 3-12.
14. Guiard, Y., Asymmetric Division of Labor in Human Skilled Bimanual Action: The Kinematic Chain as a Model. *Jour. of Motor Behavior*, 19, 4 (1987), 486-517.
15. Hang, A., Rukzio, E., and Greaves, A. Projector phone: a study of using mobile phones with integrated projector for interaction with maps. In *Proc. MobileHCI '08*. 207-216.
16. Harrison, C., Benko, H., and Wilson, A.D. OmniTouch: wearable multitouch interaction everywhere. In *Proc. UIST '11*. 441-450.
17. Harrison, C., Tan, D., and Morris, D. Skininput: appropriating the body as an input surface. In *Proc. CHI '10*. 453-462.
18. Hilliges, O., Izadi, S., Wilson, A. D., Hodges, S., Garcia-Mendoza, A., and Butz, A. Interactions in the air: adding further depth to interactive tabletops. In *Proc. UIST '09*. 139-148.
19. Hinckley, K., Pausch, R., Goble, J. C., and Kassell, N. F. A survey of design issues in spatial input. *Proc. UIST '94*. 213-222.
20. Hudson, S. E., Harrison, C., Harrison, B. L., and LaMarca, A. Whack gestures: inexact and inattentive interaction with mobile devices. In *Proc. TEI '10*. 109-112.
21. Kabbash, P., Buxton, W., Sellen, A., Two-Handed Input in a Compound Task. In *Proc. CHI '94*. 417-423.
22. Kabbash, P., MacKenzie, I.S., and Buxton, W. Human performance using computer input devices in the preferred and non-preferred hands. In *Proc. CHI '93*. 474-481.
23. Kendon, A. (1988). How Gestures Can Become Like Words. In *Crosscultural Perspectives in Nonverbal Communication*. Toronto, C.J. Hogrefe, 131-141.
24. Laakso, S. and Laakso, M. Design of a body-driven multiplayer game system. *Comput. Entertain.* 4, 4 (2006), 1544-3574.
25. Li, F.C., Dearman, D., and Truong, K.N. Virtual shelves: interactions with orientation aware devices. *Proc. UIST '09*. 125-128.
26. Mistry, P., Maes, P., and Chang, L. WUW - wear Ur world: a wearable gestural interface. In *CHI '09 Ext. Abst.* 4111-4116.
27. Mine, M.R. Brooks, F.P. and Sequin, C.H. Moving objects in space: exploiting proprioception in virtual-environment interaction. In *Proc. SIGGRAPH '97*. 19-26.
28. Mulder, A. (1996). Hand Gestures for HCI, School of Kinesiology, Simon Fraser University, Technical Report 96-1.
29. NTT IT Corp. TenoriPop: <http://tenoripop.com>
30. Post, E.R. and Orth, M. Smart Fabric, or "Wearable Clothing." In *Proc. ISWC '97*. 167-168.
31. Raskar, R., Beardsley, P., van Baar, J., Wang, Y., Dietz, P., Lee, J., Leigh, D., and Willwacher, T. RFIG lamps: interacting with a self-describing world via photosensing wireless tags and projectors. In *Proc. SIGGRAPH '04*. 406-415.
32. Saponas, T. S. Enabling always-available input: through on-body interfaces. In *Proc. CHI '09 Ext. Abst.* 3117-3120.
33. 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. UIST '09*. 167-176.
34. Smith, J., White, T., Dodge, C., Paradiso, J., Gershenfeld, N., and Allport, D. Electric Field Sensing For Graphical Interfaces. *IEEE Comput. Graph. Appl.* 18, 3 (1998), 54-60.
35. Starner T., Pentland A., Weaver, J. Real-Time American Sign Language Recognition Using Desk and Wearable Computer Based Video, *IEEE Trans. on Pattern Analysis and Machine Intelligence*. 20, 12 (1998). 1371-1375.
36. Sturman, D.J. and Zeltzer, D. A Survey of Glove-based Input. *IEEE Comp Graph and Appl*, 14, 1 (1994). 30-39
37. Sugrue, C. 2007. "Delicate Boundaries" (art installation).
38. Tan, D., Morris, D., and Saponas, T. S. 2010. Interfaces on the go. *ACM XRDS*, 16, 4 (2010), 30-34.
39. Valii, C. (ed). (2006). *The Gallaudet Dictionary of American Sign Language*; 1st edition. Gallaudet University Press.
40. Warren, J. (2003). *Unencumbered Full Body Interaction in Video Games*. MFA Design and Technology Thesis, Parsons School of Design.
41. Willis, K. D. and Poupyrev, I. MotionBeam: designing for movement with handheld projectors. In *Proc. CHI '10 Ext. Abst.* 3253-3258.
42. Wilson, A. Robust computer vision-based detection of pinching for one and two-handed gesture input. *Proc. UIST '06*, 255-258.
43. Wilson, A. and Benko, H. Combining Multiple Depth Cameras and Projectors for Interactions On, Above and Between Surfaces. In *Proc. UIST '10*. 273-282.
44. Witten, I.H. and Frank, E. (2005). *Data Mining: Practical machine learning tools and techniques*, 2nd Edition, Morgan Kaufmann, San Francisco.
45. Yamamoto, G. and Sato, K. PALMbit: A PALM Interface with Projector-Camera System. In *Adj. Proc. UbiComp '07*. 276-279.
46. Yee, K. Peephole displays: pen interaction on spatially aware handheld computers. In *Proc. CHI '03*. 1-8.