Digits: Freehand 3D Interactions Anywhere Using a Wrist-Worn Gloveless Sensor

David Kim[1,2], Otmar Hilliges[1], Shahram Izadi[1], Alex Butler[1],
Jiawen Chen[1], Iason Oikonomidis[1,3], Patrick Olivier[2]

[1]Microsoft Research
7 JJ Thomson Ave
Cambridge, UK

[2]Culture Lab
Newcastle University
Newcastle, UK

[3]FORTH
University of Crete
Heraklion, Greece

{b-davidk, otmarh, shahrami, dab, jiawen}@microsoft.com, oikonom@ics.forth.gr, p.l.olivier@ncl.ac.uk

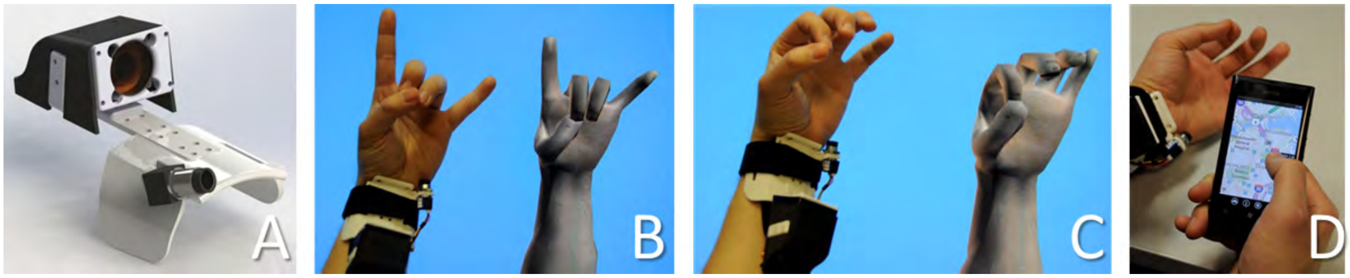

Figure 1: A) Digits is a wrist-worn sensor for freehand 3D interactions on the move. By instrumenting only the wrist, the user's entire hand is left to interact freely without wearing a data glove. B and C) Digits recovers the full 3D pose of a user's hand. D) Spatial interactions using a mobile phone and Digits.

ABSTRACT

Digits is a wrist-worn sensor that recovers the full 3D pose of the user's hand. This enables a variety of freehand interactions on the move. The system targets mobile settings, and is specifically designed to be low-power and easily reproducible using only off-the-shelf hardware. The electronics are self-contained on the user's wrist, but optically image the entirety of the user's hand. This data is processed using a new pipeline that robustly samples key parts of the hand, such as the tips and lower regions of each finger. These sparse samples are fed into new kinematic models that leverage the biomechanical constraints of the hand to recover the 3D pose of the user's hand. The proposed system works without the need for full instrumentation of the hand (for example using data gloves), additional sensors in the environment, or depth cameras which are currently prohibitive for mobile scenarios due to power and form-factor considerations. We demonstrate the utility of Digits for a variety of application scenarios, including 3D spatial interaction with mobile devices, eyes-free interaction on-the-move, and gaming. We conclude with a quantitative and qualitative evaluation of our system, and discussion of strengths, limitations and future work.

ACM Classification Keywords

H.5.2 [Information Interfaces And Presentation]: User Interfaces - Interaction styles;

UIST'12, October 7-10, 2012, Cambridge, Massachusetts, USA.
Copyright 2012 ACM 978-1-4503-1580-7/12/10...$15.00.

Author Keywords

Hand tracking, 3D interaction, mobile, wearables.

INTRODUCTION

Our hands are extremely dexterous, making them the primary mechanism to manipulate and interact with the physical world. Understandably a considerable focus of HCI research has been in transferring such 'natural' hand manipulations into the digital domain. However, current user interfaces rarely leverage the full dexterity of our hands. This is largely due to the technical challenges in sensing the full 3D pose of the hand, with its many degrees-of-freedom (DoF).

Consequently, systems constrain the problem along different dimensions, such as limiting hand tracking to 2D input only [9, 24], focusing on fingertips or other specific parts of the hand [19, 35], only supporting interactions through surfaces and other tangible mediators [5, 10, 11], or supporting a small discrete set of hand gestures [14, 32, 33]. Until recently, real-time 3D hand tracking required full instrumentation of the hand (see [7]). Researchers (particularly in the computer vision community) have begun to demonstrate algorithms for real-time 3D hand pose recovery [27, 31] and their applications for HCI [37].

Whilst sensing the full 3D pose of the hand is becoming more tractable, particularly with the advent of consumer depth cameras, there are certain domains where it still remains a fundamental challenge. The mobile domain is one such example, and the focus of this paper. In mobile settings, computational cost, power consumption, form-factor, everyday use, and self-containment are all key requirements. This makes hand tracking solutions impractical that require the sensor to be embedded in the environment, leverage currently bulky and power inefficient depth cameras, or that require the hands of the user to be fully covered.

This paper explores this very challenge and enables dexterous 3D hand interactions on the move. Our system, called Digits, is a compact form-factor wrist-worn device, built using easily reproducible off-the-shelf hardware, which is orders of magnitude lower power than current depth cameras. The sensor is fully contained on the body, requiring no external sensing infrastructure in the environment, and allows diverse bare hand interactions.

This focus on mobile use does not come at the cost of sensing fidelity. We present efficient processing techniques for robustly detecting features of the hand using the wrist-worn device, and new algorithms, which leverage knowledge regarding the biomechanical constraints of the hand to recover a fully articulated 3D model of the hand.

We leverage the recovered 3D hand pose to recognize diverse hand gestures, including discrete and continuous 3D gestures. We demonstrate interactive scenarios, including: eyes-free spatial interactions on the move, in-air 3D interactions around the periphery of the mobile device, combinations of on surface and above the surface input on mobile phones, controlling large displays from a distance using freehand gestures, and immersive 3D gaming experiences.

Our main contributions can be summarized as follows:

- A wrist-worn system that senses the full 3D hand pose without external sensing infrastructure or full instrumentation of the hand (unlike existing data gloves).
- A real-time signal processing pipeline that robustly tracks points on the user's hand.
- Two new kinematic models that allows for full reconstruction of the hand pose from these sparse points.
- A demonstration of compelling interactive scenarios that Digits enables.
- An evaluation of the feasibility of this approach through preliminary qualitative and quantitative studies.

RELATED WORK

Given their dexterity, sensing the full 3D pose of a user's hand is a technically challenging and active area of research. One approach is to instrument the hand directly by wearing a glove with embedded sensors or markers (see [7] for an overview). Originally designed for AR/VR, such sensors have since been leveraged for mobile scenarios. However, gloves can be cumbersome and uncomfortable to wear, can degrade tactile sensation and limit interaction with capacitive touch sensors, which are now standard on mobile phones.

An approach that avoids user instrumentation is to place a camera in the environment pointed directly at the user's hand. Many systems focus purely on fingertip detection for on surface (see [5]) or in-air input (see literature review in [20]). Other systems have recognized a small set of discrete 3D hand poses (e.g. [14]), or leveraged 3D shape and motion analysis for approximating freehand interactions [12].

The vision community has explored higher fidelity 3D hand tracking methods (see [8] for a detailed survey). Real-time 3D pose recovery is becoming feasible, using either non-parametric methods that typically use nearest neighbor lookup into a database of RGB [31] or depth images [37], or attempt to fit a parametrized model of the hand to observed images [27]. These systems offer some of the most sophisticated mechanisms for 3D hand tracking, without the need for direct user instrumentation. However, these approaches clearly lack mobility, requiring sensors embedded in the environment and high computational cost.

Depth Cameras On the Move The rise of consumer depth cameras has led to interest in 3D interaction on the move. Researchers have explored handheld [15, 25], shoulder- [10] or even shoe-worn depth cameras [3], all requiring line of sight of the user's hands. None of these systems however support full 3D hand pose recovery, but instead focus on sensing touch interaction with planar [10] or more complex physical surfaces [15, 25]; or detect simple pinch gestures for interaction [3]. Handheld systems can restrict freeform hand interactions that require both hands. Shoulder and shoe mounted systems alleviate this issue, but can suffer from occlusions of the hands from other parts of the body. In addition, whilst compelling in terms of sensor fidelity, there are practical barriers in making depth cameras mobile, in particular power consumption and form-factor.

Ultra-mobile Wearable Gesture Systems The wearables literature has proposed many lightweight systems for mobile gestural interaction. Instead of supporting high-fidelity sensing, they use IR proximity sensors to detect coarse motion of fingers [13, 17, 18, 26], sense muscle or tendon activity to recognize a small set of discrete hand gestures [30, 32], or leverage acoustics to coarsely localize touch on the body [11]. Given their lightweight form-factor, a variety of on-body placements including forearm [32] and wrist-worn [17, 18, 30] have been demonstrated.

Our work builds upon these systems in that we aim to provide always-available body-worn input but extend the interaction scope from sensing discrete events or coarse motion gestures to much richer continuous 3D input.

Body-Worn 2D Camera Systems One final class of wearable systems uses 2D cameras to add higher fidelity sensing without greatly compromising form-factor. These systems use either monochrome cameras and diffuse IR illumination [9, 33] or RGB cameras [19, 24, 28, 34, 35]. Inferring the full 3D pose of the hand is clearly challenging from such 2D input. Currently systems have only demonstrated simple 2D pinch gestures [9] or detecting fingers using marker [24] or markerless approaches [19, 35] for simple pointing, open and closed hand gestures. [33, 34] classify a wider set of discrete hand postures e.g. for sign language. The form-factor of these systems allows for various body-worn placements (see [22]). These include placement on shoulder or head [19, 34] and around the neck [9, 33], which have the benefit of capturing both hands for bimanual input, but restrict the interaction space to a fixed region directly in front of the user's upper body. Interactions often cannot be subtle and are publicly visible (a potential barrier for adoption in public [3]). Arm fatigue can be an issue for prolonged use.

A viable option to closer couple hand and sensor are cameras directly looking across the hand of the user. [35] use a watch-like camera to count visible fingers for coarse input. [28] presents a prototype of a wrist-worn camera that looks across the hand and images fingertips with markers placed on each finger, inferring hand pose from the 3D fingertip estimate using inverse kinematics (IK). The system is fairly large in terms of form-factor and the requirement of wearing markers on each finger is a clear limitation.

Design Considerations As illustrated by the breadth of the related work, this is a rich and challenging design space. With Digits we aim to bring some of the high fidelity sensing found in data gloves and environment-based vision techniques to mobile settings. This requires the full recovery of 3D articulated hand poses in real-time without the requirement to wear a glove. We strive to build a system that is more practical than using today's depth camera technologies, which as outlined carry serious limitations for mobile use. We identify an interesting area in this design space and focus on a wrist-worn device to recover the full 3D hand pose, but do this with only a 2D camera.

We build upon learnings from the wearables literature that have shown the benefits of wrist-worn devices [17, 18, 13] for supporting eyes-free, always-available interactions [1], as well as overcoming some of the limitations of other sensor placements, including occlusions of the hands by other body parts, imposing interactions that are overly performance-centric (instead of private or subtle [3]), and physically constraining the interaction space due to line of sight requirements of the sensor.

DIGITS – SYSTEM OVERVIEW

Digits is a small camera-based sensor attached to the wrist that optically images a large part of the user's bare hand. The camera is placed so that the upper part of the palm and fingers are imaged as they bend inwards towards the device. Two separate infrared (IR) illumination schemes are used to simplify signal processing. The use of IR allows the illumination to be invisible to the user, and offers a level of robustness to ambient visible light. Both illumination schemes use low cost, readily procurable, and low-power components.

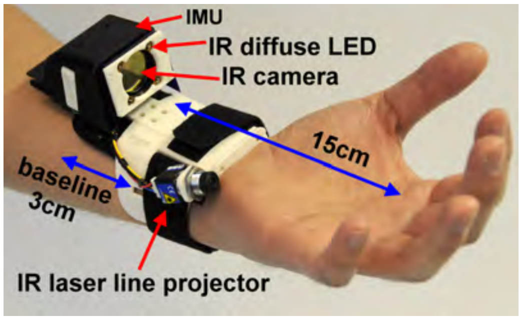

Figure 2: Digits main hardware components attached to a wrist brace.

First, an IR laser line generator projects a thin IR line across the user's hand which intersects with the fingers as they bend inwards [36]. This approach can be used to robustly sample a single 3D point on each of the fingers and thumb. From these five sparse samples, and by exploiting biomechanical constraints, we derive a forward kinematics (FK) algorithm to reconstruct a fully articulated hand skeleton. This initial approximation allows us to support a variety of 3D hand poses as shown in Fig. 9. However, each fingertip is essentially reduced to a single dimension of input, limiting its motion to curling either towards or away from the sensor (see Fig. 7B).

To more faithfully replicate the pose of the hand, more features need to be sampled on the hand. A complimentary method uses a ring of modulated IR LEDs to uniformly illuminate the user's hand. We demonstrate how to robustly extract the 2D positions of fingertips from this IR image, with an associated coarse depth estimate, by modeling the light falloff from the IR LEDs. This fingertip sensing approach can be coupled with the laser line sensing method to derive a new inverse kinematics (IK) based algorithm for computing the full joint-angle configuration of the hand. This method allows for even more realistic reconstructions of the hand as shown in Fig. 12, resulting in higher DoF input.

These methods work together to help constrain the otherwise ill-posed problem of recovering the full 3D hand pose from a 2D image. Because each method may be useful by itself, we keep the description of the two approaches separate. This also aids in understanding the underlying concepts, techniques and algorithms presented in this paper. Finally, an inertial measurement unit (IMU) can be used to approximate wrist and forearm motions in 3D (see Fig. 2).

Digits is primarily a new type of sensor and an enabling technology for interaction, however before describing the system in full, we demonstrate its interactive capabilities.

Interactive Scenarios

We have explored a number of interactive scenarios enabled by our system which we briefly illustrate here. Each scenario looks at a different configuration of output coupled with Digits. The first scenario looks at spatial interaction with a situated display (see Fig. 3A+B). For example, interacting with a TV at home or a large public display. Here the user interacts at a distance using a Digits device. Application scenarios can include gaming or CAD, where the user can perform a variety of continuous or discrete hand gestures to support spatial navigation, pointing or selection in 3D (Fig. 3A+B). From this tracked 3D hand model, discrete gestures can be robustly recognized by looking at the joint angle configuration (Fig. 3C).

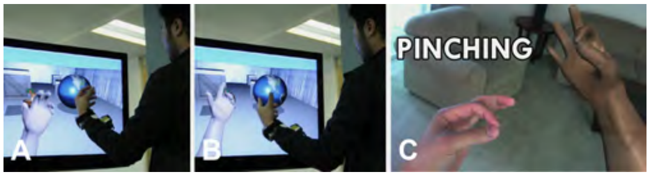

Figure 3: Illustration of potential Digits application scenarios. A+B) Continuous interaction with 3D content on a large display. C) Gesture recognition performed on reconstructed hand model.

Mobility was a strong motivation in our work. We envision Digits will expand the physical interaction area of mobile devices beyond the display. In one application scenario (see Fig. 4A+B) the user holds and interacts with a tablet (or phone) using the dominant hand and uses the non-dominant hand to provide 3D input to the application. For example, semantic zooming is initiated with an in-air pinch gesture, and the zoom factor is controlled with the remaining digits.

An interesting possibility here is to support on-screen interactions and simultaneous freehand interactions. For example, dividing between fine-grained interactions on the touch-

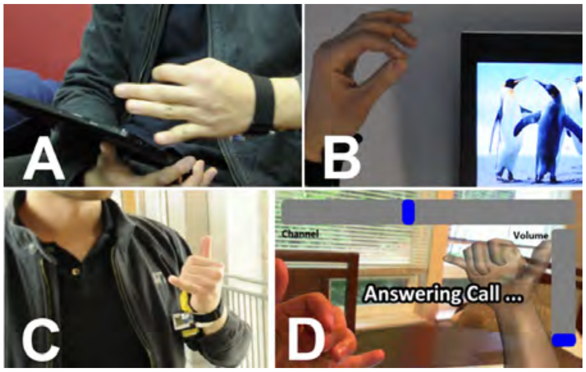

Figure 4: Digits application scenarios. A+B) Extending interaction space around a mobile device into 3D. C+D) Non-visual UIs allow users to manipulate application parameters without looking at or touching a physical device (GUI elements are for illustration only).

screen and coarser navigation tasks using the non-dominant hand and Digits. Our tracker is gloveless and enables the user to interact with the mobile device and operate Digits with the same hand. This allows standard 2D touch gestures to be coupled with above-the-surface interactions in 3D. For example, for quickly changing the Z-order of a selected object by first touching the target and then performing an in-air pinch-based zoom.

Finally, the display could be removed entirely in an eyes-free interaction scenario. The 3D input capabilities of Digits can allow spatial interactions with invisible UIs, such as dials, sliders, or buttons without visual output (see Fig. 4C+D). For example, we can leverage both proprioceptive knowledge and spatial memory to allow the user to set the volume on a mobile phone by directly reaching out and interacting with a virtual dial; turning their hand to the right of the body and performing typing gestures on a virtual number pad to place a call; or moving the hand to the left of the body and touching their thumb and individual fingers to activate other phone functions. One interesting possibility here is to detect the type of action by the initial 3D shape of the hand. For example, if the user requires to change the volume, they simply configure their hand as if they are holding a virtual dial, which can then be rotated to set the desired level.

These scenarios illustrate the utility of the Digits as a general purpose platform for a variety of hand-based interactions. In the next sections, we describe the system implementation in full, focusing on how we sense features of the hand and from these build different articulated models of the hand, with varying levels of fidelity.

SENSING HARDWARE

The Digits hardware is shown in Fig. 2. A PointGrey FireFly MV IR camera (640x480 resolution capturing frames at 90Hz) is attached to a wristband worn on the anterior (inner) side of the wrist. An IR laser line generator (Gated Cameo 1260 from Global Laser) operating at 850nm, with 105° angular spread is positioned at a fixed baseline from the camera. 4 diffuse IR LEDs (OSRAM CHIPLED SFH 4053) again operating at 850nm are attached around the lens of the camera. Finally, an IMU (x-IMU from x-IO Technologies) provides absolute tri-axis orientation data of the forearm at 120Hz.

The hardware is designed to be simple, easily reproducible with off-the-shelf components, and low-power. The setup is powered entirely over USB, both laser and the 4 LEDs powered and driven by strobes from the camera General Purpose Input/Output (GPIO) pins. This results in a total power draw of less than 400mW (16mW for the laser, 60mW for the IR LEDs, and 300mW for remaining camera hardware). This compares favorably to the 3.4 to 5W consumption of the current generation Kinect cameras. A Digits device weighs around 75g, 124g with the wireless IMU, and is lighter than standard watches with a metal wristband (160-180g). The device is attached to the forearm with a 2cm wide Velcro band around the wrist, and the contact area at the inner wrist is covered with soft padding (5x5cm).

SIGNAL PROCESSING

The main processing pipeline is broken down into the following steps:

Background Subtraction We reduce the influence of ambient IR light by capturing three consecutive frames from the IR camera under different illumination conditions (giving an effective frame rate of 30Hz). The first frame turns all active illumination off, capturing only ambient IR. This is subtracted from the other two frames; the first with only the laser on, and the second with just LEDs on. As shown, ambient IR light is greatly reduced in our input images e.g. from room lights or sunlight (Fig. 5).

Image Rectification Rectifies both actively lit images based on a previous intrinsic camera calibration step.

Finger Separation Splits the LED lit IR image into regions that correspond to different unique fingers or thumb.

Laser Line Sensing Triangulates the 3D points where each finger or thumb intersects with the laser line generator.

Forward Kinematics These 3D points are passed to a new forward kinematics algorithm, which reconstructs the full 3D hand pose, based on assumptions regarding the biomechanical constraints of the hand.

Diffuse IR Fingertip Detection Additionally, depending on the hand pose, we can also use the LED illuminated image to robustly extract high-quality surface normals and coarse depth estimation for robust detection of fingertips.

Inverse Kinematics The 3D points sensed from the laser and 2D fingertip locations are passed onto a new IK model for higher DoF recovery of hand poses.

Finger separation One of the important initial pipeline steps associates regions of the image with each of the digits of the hand. A technique utilized in *seam carving* [2] is used to disambiguate the main vertical boundary between pairs of fingers in the IR illuminated image. A one dimensional sobel filter finds vertical edges in the IR image. We detect *valleys* [20] between two fingers as concavities in the traced hand contour. At each valley location, we trace multiple vertical paths along the edges, and use dynamic programming to detect the path with the lowest overall energy by penalizing paths not following the edge (see Fig. 5F). This method divides the image into five areas each mapped to a unique digit.

Laser Line Sensing The laser line generator projects a horizontal line above the palm that intersects with parts of

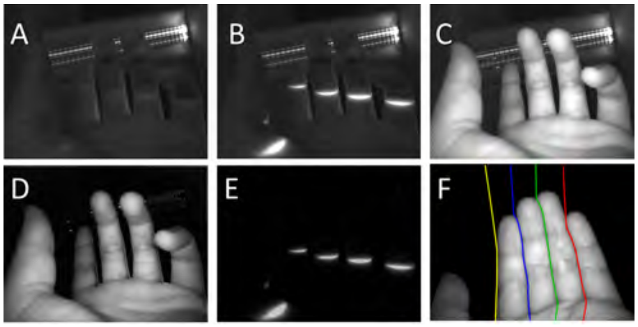

Figure 5: Background subtraction. A) Active illumination off. B) IR laser + background IR. C) IR LEDs + background IR. D) Background subtracted IR LEDs. E) Background subtracted IR laser. F) Finger separation via seam carving.

each finger. These intersections appear as bright regions in the 2D IR camera image, and move towards the palm as the finger is bent, and in the opposite direction when the finger is straightened (see Fig. 7). With a fixed known baseline between laser and the camera, it is possible to triangulate the exact 3D position of each laser line segment (our current implementation uses a 3cm baseline as a reasonable trade-off between depth accuracy and ergonomics, see Fig. 2).

In a one-off process, the camera and laser are calibrated: The camera's intrinsic parameters are retrieved using a checkerboard calibration method [39], and are used for image rectification. Next, the user moves the same target and intersects it with the laser line (see Fig. 6A). The 6DoF extrinsic pose of the target is computed relative to the camera center [39]. The user clicks on an intersection point, and the associated 3D point is recorded. The process is repeated until three non-coplanar points are selected to define the laser plane relative to the camera.

To triangulate the 3D intersections of the finger and laser, the background subtracted and rectified image is first binarized for connected component analysis. Intersections are clearly visible as elongated ellipsoids which are filtered based on size and shape. Merged connected components (when fingers are close to each other) are separated using the previous seam carving output. The centroid of each connected component is reprojected using the camera intrinsics. From the camera center a ray through the centroid is intersected with the derived laser plane (see Fig. 6B). This defines a 3D point for each finger, relative to the camera.

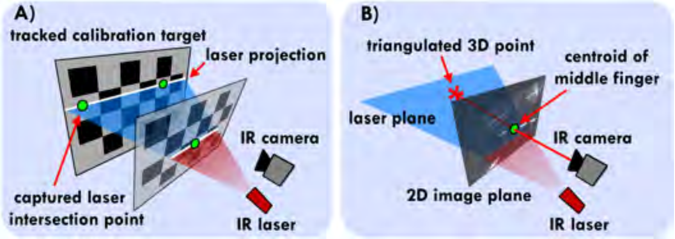

Figure 6: A) Laser plane calibration procedure. B) Reprojected ray intersecting with the laser plane.

A SIMPLE KINEMATIC HAND MODEL

After retrieving the 3D intersections of laser with fingers, we use a new kinematic method to recover the hand pose. Fig. 7 shows the main finger bones and joints in a hand. Each finger is comprised of three bones, namely proximal, middle, and distal phalanges. From fingertip to palm, these bones are interconnected by a 1DoF revolute joint called the distal interphalangeal (DIP) joint, a 1DoF revolute proximal interphalangeal (PIP) joint, and a 2DoF spherical joint called the metacarpophalangeal (MCP) joint [4].

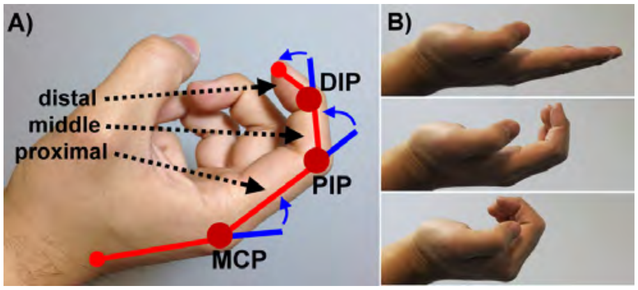

Figure 7: A) Illustration of main finger bones and joints. B) Natural flexing of fingers. Proposed forward kinematics model reconstructs this behavior.

These bones do not move in an entirely independent fashion. Specifically, it has been shown that the DIP joint angle depends on the PIP angle owing to interaction of tendons attached to middle and distal phalanges [4]. Furthermore, as shown experimentally by [16] during natural flex of the finger (i.e. when not explicitly controlling MCP independently of other joints) MCP joint angles depend on the PIP angle when bending the middle and distal phalanges. We leverage the interdependency of these bones and joints to map from sparsely sensed 3D locations on each finger to a 3D articulated model of the hand.

Specifically, during natural flex of each finger (see Fig. 8 Right) a linear relationship exists between all three joints of the finger, such that both MCP and DIP can be derived if the PIP angle is known. Kamper [16] experimentally found the ratio between PIP and DIP is $\frac{1}{0.84}$ and $\frac{1}{0.54}$ for PIP to MCP respectively. Using these ratios, we can approximate a common finger motion, when an outstretched finger curls inwards until it touches the palm, only with a single parameter.

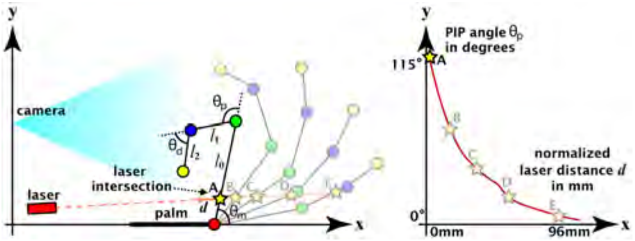

Figure 8: Left: Forward kinematics model for a single finger intersecting with laser line. Right: Graph mapping between laser distance and PIP joint angle.

Calculating joint angles

To determine the articulation of each finger during natural flex, we experimentally derived the mapping between 3D intersection points and the PIP joint angle using a simple forward kinematic model Fig. 8. We simulate each of the bones with predefined lengths (l_0, l_1, l_2 respectively). Average bone lengths can be taken from the literature [4, 16] or measured per user. We compute the PIP angle at simulation time as θ_p $[0°, 120°]$. MCP and DIP are calculated as $\theta_m = a\theta_p$ and $\theta_d = b\theta_p$ respectively where a and b are the joint ratios derived from [16].

During calibration, we also derive the 6DoF pose of the palm with respect to the camera (using a checkerboard placed on the palm), which allows us to determine the offset of the laser and its direction with respect to the palm. We simulate the forward kinematics model by changing the joint angle θ_p. At 0° the finger is outstretched and fully bent at 120°. For each of the angles of θ_p we measure the intersection between the laser ray and each of the bones, and take the minimum distance. The sampled data is plotted in Fig. 8 and used to map from laser distance to θ_p. The graph can be fitted using the following cubic function (where d is the distance to the intersection in mm):

$$\theta_p = -0.0003 * d^3 + 0.059 * d^2 - 4.07 * d + 119.75 \quad (1)$$

As fingers have similar anatomy, it is reasonable to assume that this function is valid for all fingers. We therefore provide a simple one-off online calibration process for each finger, where we plot the principal axis of motion for each finger. New intersections are normalized along this axis. Because we normalize along a 3D line, this approach also works for the thumb which moves more diagonally in the sensor image. While articulated thumb motion is reasonably tracked in practice, results could be further refined by explicitly building a similar model as in Fig. 8 purely for the thumb. Our model can be extended to lateral motions of fingers (i.e. allowing fingers to move left and right), by mapping deviation from the calibrated principal axis to a 3D rotation, which is applied to each finger after articulating finger bend.

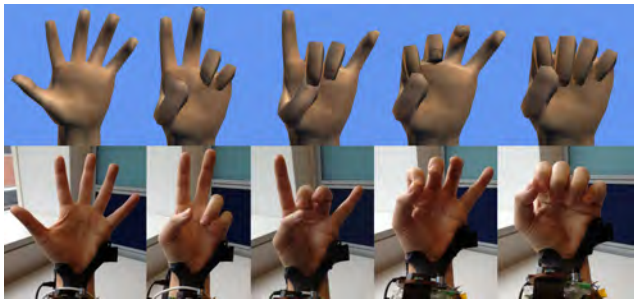

Figure 9: Top: Various hand poses supported by our forward kinematics model. Bottom: User's real hand poses.

Strengths and limitations

This approach provides a simple, but yet natural approximation of hand poses. Fig. 1 and Fig. 9 show a number of real-world hand poses and how these can be replicated using Digits. Whilst the combination of laser line sensing and forward kinematics model is powerful in its own right, there are limitations. In particular, it assumes a strong relationship between all of the joints of each finger. However, there are also common cases where for example the MCP joint moves independently of the PIP. To deal with these wider range of hand poses, we need to sense other parts of the hand and provide an extended kinematics model. Through experimentation we have found that illumination from IR LEDs can be used to robustly detect fingertips, which allows us to define a more complete kinematic model of the hand.

FINGERTIP DETECTION

Despite it being unfeasible to place a full depth camera on the wrist (due to form-factor and power), we demonstrate how to generate high-resolution normal maps and coarse depth estimates, by modeling the light falloff from the LEDs. This provides a robust method for identifying fingertips in the 2D image, even when fingers are directly facing the camera.

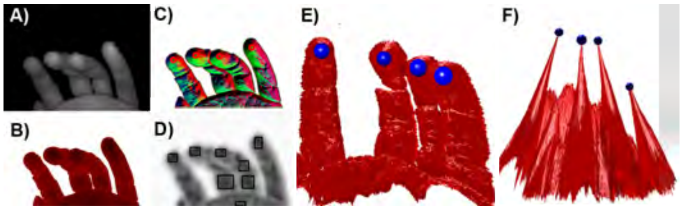

Figure 10: Our fingertip detection pipeline. A) Background subtracted imaged fingers. B) Estimated depth encoded in red color channel. C) Normal map computed from depth map. D) Response map from template matching (darker is closer match), black rectangles mark fingertip candidates. E) Recovered mesh viewed from the physical camera's view. F) Depth distortion visible when viewed from off-center viewpoint. Note the distinct peaks caused by fingertips.

Depth approximation Our approach adapts work on shape-from-shading (SFS) (see overview in [29]). Our scenario makes SFS more tractable, given partially known parameters for (LED) light position, light power and radial intensity falloff, as well as the ability to approximate the skin reflectance model as purely Lambertian (given the typically small angle of incidence between surface and light source [21]).

We first estimate distance measurements for each pixel under the *inverse-square law*. The intensity at distance d is $I = 1/d^2$ solving for d this gives us an initial distance estimate for each pixel u as: $d(u) = \sqrt{I(u)}$. This distance estimate is then attenuated according to the radial falloff in light intensity for pixels further away from the light's central ray. The final distance value is computed as:

$$D(u) = \sqrt{I(u)} * \frac{1}{\cos(\arctan(\frac{(u-pp)}{fl}))}$$

with known principal point pp and focal length fl from camera calibration. This computes a depth map, where each pixel can be reprojected as a 3D point in camera coordinate space. Finally, we compute surface normals for each pixel from adjacent pixels in the depth map (see Fig. 10).

Strengths and Limitations This technique provides only approximated depth values, as pixel intensity depends on many factors beyond distance to the light source and light falloff. In particular, we do not model the shape and material of the imaged 3D surface nor do we take the surface orientation into account. While the resulting depth map looks plausible when viewed from the cameras perspective, distortions and non-linearity of the signal become clearly visible when viewed off-center (Fig. 10F).

However, this approach can be powerful in detecting fingertips and computing a relative depth estimate. Because fingertips are fairly spherical in shape (in particular when pointing towards the camera) they produce a distinct signal as it can be seen clearly in the normal map (Fig. 10C) and the mesh rendering (Fig. 10F). Each fingertip produces a very distinct peak in depth very similar in shape to a Gaussian sphere centered around the finger's most protruding part.

To detect and track these peaks many methods are viable. We have obtained robust results with just simple template matching (based on matching scores as squared distances between a sliding synthetic fingertip template and the live normal map). Fig. 10D-F shows the final result of our technique, reliably detecting fingertip locations. In particular, our technique works for fingertips pointing towards the camera and multiple fingers touching each other, situations in which simple 2D techniques such as peak-and-valley algorithms, or connected component analysis alone would have difficulties.

A NEW KINEMATICS MODEL

Fingertip estimates can be combined with the laser line intersections to recover a more accurate hand model using inverse kinematics (IK). IK typically derives joint angles from the 3D position of the end effector – the fingertip. We do not have an accurate 3D measurement for fingertips, and the 3D point sampled with the laser is not directly associated with the end effector. However, the two sensing modalities can be combined to derive a new IK model enabling separate articulation of the MCP joint and the PIP/DIP joints.

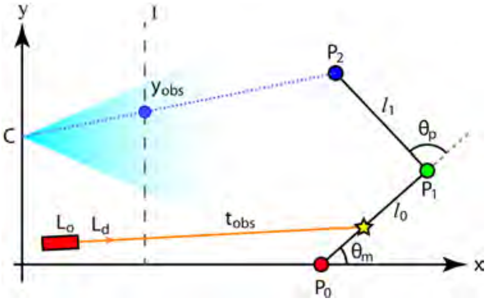

Figure 11: IK model that articulates both the MCP joint and the PIP/DIP joints based on an sensor fusion approach

Fig. 11 illustrates a simplified parametrization of our IK model. Note for illustrative purposes we have reduced the problem to 2D and combined PIP and DIP joints, which cannot be moved independently unless the finger is pressed against a surface. In this parametrization, the palm is again assumed to be resting directly on the X axis. The position of the MCP joint is given by P_0, the position of the PIP joint is P_1 and the end effector is at P_2. Whilst the 3D location of the end effector is not known, we can observe the projection of the point (y_{obs}) on the image plane (I) (as this equates to the centroid of the detected fingertip in the IR image). Given the calibration matrix of the camera, we can project a ray from the camera center (C) through the image plane. We know that P_2 exists somewhere along this ray.

The length of each of the bones (l_0, l_1), of the finger are again assumed to be known, either by measurement or from assuming predefined values. We are solving for the MCP and PIP joint angles given as θ_m and θ_p respectively. We can also parameterize the laser as an offset from the origin (L_o) and direction (L_d). We also have an observed 3D point with distance t_{obs}, sensed from the laser (L_{obs}) which we know intersects one of the bones (the specific bone is unknown).

P_0 is known ahead of time, using the laser line to calculate the minimum 3D extent of each finger (as described previously). This allows us to calculate P_1 by applying a local transform (translation by bone length l_0 and rotation around the joint angle θ_m) to P_0. So that $P_1 = \mathbf{R}(\theta_m) \cdot [l_0, 0]^T + P_0$ and $P_2 = \mathbf{R}(\theta_p) \cdot [l_1, 0]^T + P_1$.

Our aim now is to find the optimal combination of θ_m and θ_p to best describe the observed data (the location of the 2D fingertip sensed using the LEDs and 3D point measured with the laser). For our fingertip location, we define the following energy function:

$$E_{led} = |proj(P_2) - y_{obs}|^2 \tag{2}$$

This function generates estimated positions for P_2 given variations of θ_m and θ_p, and projects these onto the image plane I (using the intrinsic camera calibration parameters). It has a low energy for points that are close to the observed point on the image plane y_{obs}.

Our second energy function first calculates intersections between the laser line and each bone in the finger, based on variations of θ_m and θ_p and takes the minimum:

$$\begin{aligned} t =& \min\{\text{isect}(\overrightarrow{L_0 L_d}, \overline{P_0 P_1}), \\ & \text{isect}(\overrightarrow{L_0 L_d}, \overline{P_1 P_2})\} \end{aligned} \tag{3}$$

It then minimizes the distance between the observed 3D laser point L_{obs} and this estimated intersection:

$$E_{las} = |tL_d + L_o - L_{obs}|^2 \tag{4}$$

Our full energy function is specified as:

$$\underset{\theta_m, \theta_p}{\arg\min}\, E = E_{led}\lambda_{led} + E_{las}\lambda_{las} \tag{5}$$

This allows us to weight the contribution of either the LED or laser based sensing accordingly, using a scalar (λ). In our current implementation, we evaluate this energy function across θ_m $[0^\circ, 100^\circ]$ and θ_p $[0^\circ, 90^\circ]$ in a brute force manner and select the value with the minimum energy.

This new kinematic model give us even higher DoF input sensing. As shown in Fig. 12 a wider range of hand poses can be more accurately predicted from the raw sensor data. This includes a wide range of poses that are difficult to predict using our simpler kinematics model. The combination of the two sensing modalities – both laser line and light falloff – allow us to solve the otherwise ill-posed IK problem.

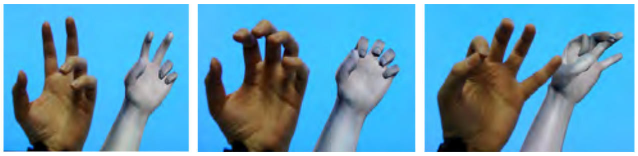

Figure 12: Truer hand pose recovery with inverse kinematics. Note PIP and MCP joint angles recovered correctly.

INITIAL EVALUATION

We performed a preliminary evaluation of Digits, to get a sense of the *accuracy* and *repeatability* of the system at reconstructing different hand poses. Whilst not a detailed user or system evaluation, this study was meant to provide an initial feasibility test for this approach to 3D hand tracking.

Gesture Feasibility Experiment

To provide a first step towards quantifying accuracy and repeatability, we asked participants to mimic hand postures rendered on screen in an interactive 3D application. We used ground truth data gathered using a Vicon motion tracking system as proposed in [23]. Six static hand poses, as shown in Fig. 13, were generated. In the experiment we tested the full IK model described in the previous section.

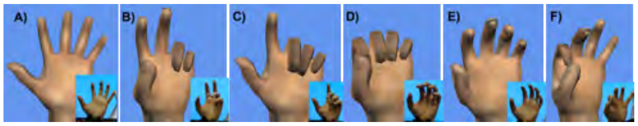

Figure 13: Hand postures in experiment. A) Open palm B) Counting two C) Pointing D) Grasping small object E) Grasping large object F) Pinching.

Participants We recruited 12 participants aged 24 to 39, with a mean age of 32. 10 participants were right handed. None of the participants had any physical disabilities or limited range of motion. Related literature has shown that anthropometric differences in the upper limbs can make comparisons across genders incorrect [6]. Therefore for this first experiment we only tested Digits with male participants.

Procedure and Task The system was mounted on each user's wrist, as in Fig. 13. The system was calibrated for each user's hand (including bone lengths and finger motion range and trajectory). For left handed users, the camera and laser placement was adjusted and a full camera/laser calibration performed. During the study phase participants were presented with the six reference 3D hand configurations on screen (Fig. 13). A second 'live' hand model, driven by the participant's input, was rendered on-top of the reference hand. We ignored wrist and forearm reconstruction in this study (and hence did not use the onboard IMU). Left and right-handed reference postures were selected based on the participant's handedness.

Participants started each trial in a neutral pose (fully closed fist). The experimenter triggered each task. Subsequently a reference pose was rendered on screen and users were asked to align their instrumented hands as closely as possible (labeled as the 'acquisition' phase). Users self-reported once they were satisfied with their pose (by pressing a button with their uninstrumented hand), at which point they were asked to hold the pose for 3 seconds (labeled as the 'static' phase). Each user completed 3 blocks each consisting of 6 trials, which included all the 6 hand postures in random order. Users had 10 minutes training time with the application.

Results and Discussion The mean *acquisition time* was 4074 ms. The first block averaged 4417 ms, second 3981 ms and third 3823 ms. A repeated measures ANOVA reveals that there is a significant main effect for *acquisition time* across blocks ($F_{2,28} = 48, p < 0.01$). Post-hoc analysis (Bonferroni corrected $\frac{\alpha}{n} = \frac{0.05}{3} = 0.0167$) reveals linear improvement between blocks 1 and 2 ($p < 0.01$) but no significant difference between blocks 2 and 3 ($p > 0.35$).

Repeatability As a measure of repeatability we compute intraclass correlation coefficients (ICC), the standard test for repeatability [6, 23, 38] for this kind of time series data. For each of the six poses we computed ICC scores per finger and per joint between the three blocks of repetition (static phase only). We repeatedly selected two blocks randomly. Then for each block a trial was selected randomly and a ICC score was computed between them. This procedure was repeated 40 times to ensure consistency and ICC scores were averaged together. For brevity we only report individual scores for the PIP angles but overall the ICC scores were in the range of 0.6 to 0.97 (moderate to strong correlation). Correlation between repetitions was found to be moderate for little finger ($ICC = 0.66$) and good for thumb ($ICC = 0.74$) to strong for index finger ($ICC = 0.88$), ring finger ($ICC = 0.81$) and middle finger ($ICC = 0.84$). Overall these results suggest that Digits produces moderate to strongly correlated joint-angle measurements when the same pose is repeated multiple times and by multiple participants.

Accuracy To determine accuracy, data was averaged across blocks and participants, keeping joints and gestures separate. Averaging across fingers would not be very meaningful as finger motion is not entirely independent (see discussion on biomechanical constraints earlier). Again for brevity we only report results for PIP bend angles but results for PIP bend and MCP bend and tilt angles are comparable.

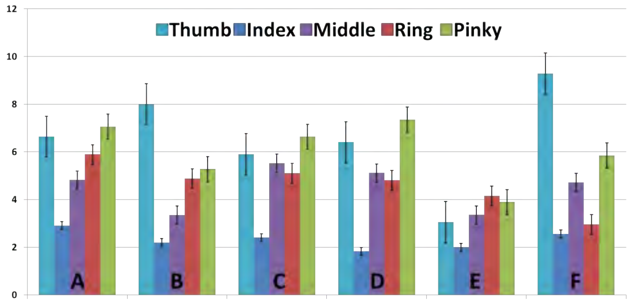

Figure 14: Mean error in PIP joint-angle (in degrees) per finger and for all six gestures. Error bars show std deviation.

Fig. 14 summarizes results for the five fingers and all six gestures. Mean errors throughout the data remain relatively small for all fingers and gestures (all < 9°). The best accuracy is usually found with the index finger (min 1.8° pose "D", max 2.9° pose "A") this could be attributed to the importance of the index finger for many everyday activities (e.g., pointing, pinching, scratching). For most poses the error is largest for the thumb, this maybe explained by the fact that our IK model treats the thumb as a regular finger while in reality the thumb shows a more complex motion (cf. [23]). The mean error for the pinky is also comparatively high (min 3.8° pose "E", max 7.3° pose "D") this may be explained by the camera placement in our setup – to achieve a compact form factor the camera was moved very close to the palm so that the pinky may not be visible in certain circumstances.

Although preliminary, these overall results are promising.

The achieved accuracy of the tracking and specifically the joint-angle error rates are comparable to those reported in studies on data gloves [6, 38]. Furthermore, the average accuracy (between 2° and 9°) is better than that defined in the literature for manual goniometry (between 7° and 10°) which is considered *"in clinical practice to be the gold standard of joint angle measurement"* [23].

DISCUSSION

Digits is a general purpose wearable 3D hand tracker. It avoids direct instrumentation of the user's hand, but instead is wrist-worn. We have shown how the full 3D pose of the user's hand can be inferred, without requiring high-fidelity and currently impractical hardware such as a depth camera to be worn on the body. The preliminary evaluation of our prototype system demonstrates tracking close to existing data gloves, but without direct instrumentation of the hand.

A tale of two kinematic models In this paper we have introduced two complementary ways to recover 3D pose from a small number of samples on the user's hand. While the different models (based on forward and inverse kinematics respectively) provide different levels of interactive expressiveness they should not be seen as one fully replacing the other. Both have their own strengths and weaknesses and utility in different contexts. Our first model has the advantage of simplicity. It allows reconstruction of rich 3D hand poses from a single measurement on each finger which already enables a number of compelling spatial continuous gestural interactions (Fig. 1 and Fig. 9) and can facilitate gesture recognition in mobile applications. The second model adds higher DOF sensing and more independent movement of each finger joint, and therefore provides more fidelity in recovering the user's hand pose. This model also allows a truer reconstruction of the hand as shown in Fig. 15. This can be useful for example in a physics-enabled application where you wish to model grasping of an arbitrary virtual object or a medical application where accurate joint-angle measurements are needed. We enable this by adding only simple additional hardware, although there is more algorithmic complexity.

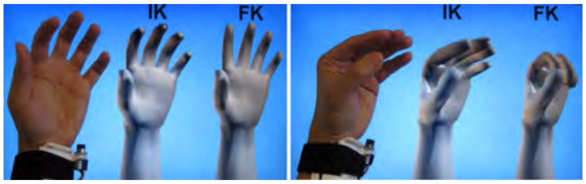

Figure 15: Failure cases for our forward kinematics (FK) model. Notice mismatch in PIP angle (left) and MCP angle (right). Using our inverse kinematics (IK) model we can mitigate such issues, and offer a truer reconstruction.

Limitations With our current implementation being a vision based technique, occlusions resulting from crossed fingers, overly bent thumb and handheld objects are problematic for hand pose reconstruction. However, these are special cases and we anticipate they can be avoided by careful gesture design. Furthermore, more advanced techniques for finger separation and identification could be used to mitigate these issues. In the current Digits prototype we do not model wrist bend and rotations about the forearm explicitly. In particular a fully flat or over-arching hand is problematic, while our image processing techniques can cope better with lateral motion of the hand relative to the camera.

Our proof-of-concept implementation already is wearable and not overly bulky but requires user instrumentation. Furthermore, the device is still tethered to a PC or laptop where computations are being performed. It is however conceivable to further miniaturize the device until it is either a standalone watch-like device or fully integrated into a regular watch. Future depth camera technologies such as time-of flight might enable such compact form factor devices. For the time being we argue that our approach is the best trade-off between practicality and availability concerns but also between computational complexity, power consumption and form-factor.

Always available input It is worthwhile noticing that with Digits input is *not* restricted to a fixed space around the user, but instead moves with the user's hand. For example, gestures can be conducted in front of the body (much like other body-worn systems [10, 24, 33]). However, gestural input may also happen in a more subtle, effortless way. For example, performing hand gestures whilst the hand is lowered by the side of the body or resting on a physical surface, avoiding arm fatigue during long periods of use. Examples of this are shown in the accompanying video figure.

Emergent interactions There are other emergent interactive features of Digits which can be fruitful to explore in future work. For example, Digits may be used to track the spatial positions of fingers from the other (non-instrumented hand) when both hands interact. Using the palm of the reference hand as a track pad, or using the segments of fingers on the reference hand to control sliders in a GUI.

Whilst Digits has been designed to be a general purpose interaction platform, we have demonstrated both in this paper and accompanying video interactive scenarios using this technology. We believe Digits is particularly useful for mobile scenarios, where the sensor can coexist with existing personal devices such as mobile phones and tablets. The combination of touch and 3D input in free space around the device is particularly interesting. Also of interest are eyes-free interfaces which allow for interactions without needing to remove mobile devices from our pockets. One final application area for Digits is in gaming, where technologies such as the Xbox Kinect, or Nintendo Wii do not currently support the level of fidelity of hand sensing. Again such a device could be complimentary to these existing sensing modalities. For example, combining the Kinect full body tracker with high fidelity freehand interactions of Digits.

CONCLUSION

We presented Digits a wrist-worn system for sensing the full 3D pose of the user's hand. The system targets mobile settings, and is specifically designed to be low-power and easily reproducible using only off-the-shelf hardware that is both smaller and more power efficient than current consumer depth cameras. We have shown two complimentary methods for robustly tracking features of the hand, and we demonstrated the evolution of a kinematics model for reconstructing the full articulated hand from these sparse samples. We demonstrated the utility of Digits for a variety of application scenarios, including 3D spatial interaction on mobile phones, eyes-free interaction on the move, and gaming.

REFERENCES

1. Ashbrook, D. L. *Enabling mobile microinteractions*. Phd thesis, Georgia Institute of Technology, 2010.
2. Avidan, S., and A. Shamir. Seam carving for content-aware image resizing. *SIGGRAPH Comput. Graph. 26*, 3 (July 2007), 10.
3. Bailly, G. et al. ShoeSense: A New Perspective on Hand Gestures and Wearable Applications. In *ACM SIGCHI* (2012), 1239–1248.
4. Becker, J. C., and N. V. Thakor. A study of the range of motion of human fingers with application to anthropomorphic designs. *IEEE Trans. Biomed. Eng. 35*, 2 (1988), 110–117.
5. Buxton, B. Multi-touch Systems that I have Known and Loved. Tech. rep., Microsoft Research, 2007.
6. Dipietro, L., A. M. Sabatini, and P. Dario. Evaluation of an instrumented glove for hand-movement acquisition. *Int. J. Rehabil. Res. 40*, 2 (2003), 179–189.
7. Dipietro, L., A. M. Sabatini, and P. Dario. A Survey of Glove-Based Systems and Their Applications. *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. 38*, 4 (2008), 461–482.
8. Erol, A. et al. Vision-based hand pose estimation: A review. *Comput. Vision Image Understanding 108*, 1-2 (Oct. 2007), 52–73.
9. Gustafson, S., D. Bierwirth, and P. Baudisch. Imaginary interfaces. In *ACM UIST* (2010), 3–12.
10. Harrison, C., H. Benko, and A. D. Wilson. OmniTouch: Wearable Multitouch Interaction Everywhere. In *ACM UIST*, ACM (2011), 441–450.
11. Harrison, C., D. Tan, and D. Morris. Skinput: Appropriating the Body as an Input Surface. *ACM SIGCHI 3*, 8 (2010), 453–462.
12. Hilliges, O. et al. HoloDesk: Direct 3D Interactions with a Situated See-Through Display. In *ACM SIGCHI* (2012), 2421–2430.
13. Howard, B., and S. Howard. Lightglove: Wrist-worn virtual typing and pointing. In *IEEE ISWC* (2001), 172–173.
14. Izadi, S. et al. C-Slate: A Multi-Touch and Object Recognition System for Remote Collaboration using Horizontal Surfaces. In *IEEE TABLETOP* (2007), 3–10.
15. Izadi, S. et al. KinectFusion: Real-time 3D Reconstruction and Interaction Using a Moving Depth Camera. In *ACM UIST* (2011), 559–568.
16. Kamper, D. G., T. George Hornby, and W. Z. Rymer. Extrinsic flexor muscles generate concurrent flexion of all three finger joints. *J. Biomech. 35*, 12 (2002), 1581–1589.
17. Kim, J. et al. The Gesture Watch: A Wireless Contact-free Gesture based Wrist Interface. In *IEEE ISWC* (2007), 1–8.
18. Lee, S. C., B. Li, and T. Starner. AirTouch: Synchronizing In-air Hand Gesture and On-body Tactile Feedback. In *IEEE ISWC* (2011), 3–10.
19. Lee, T., and T. Hollerer. Handy AR: Markerless Inspection of Augmented Reality Objects Using Fingertip Tracking. In *IEEE ISWC* (2007), 1–8.
20. Malik, S. Real-time hand tracking and finger tracking for interaction. Tech. rep., University of Toronto, 2003.
21. Marschner, S. R. et al. Image-Based BRDF Measurement Including Human Skin. In *Eurographics Rendering Workshop*, vol. 5 (1999), 1–15.
22. Mayol-Cuevas, W., B. Tordoff, and D. Murray. On the Choice and Placement of Wearable Vision Sensors. *IEEE Trans. Syst. Man Cybern. Part A Syst. Humans 39*, 2 (Mar. 2009), 414–425.
23. Metcalf, C. et al. Validation and Application of a Computational Model for Wrist and Hand Movements using Surface Markers. *IEEE Trans. Biomed. Eng. 55*, 3 (2008), 1199–1210.
24. Mistry, P., and P. Maes. SixthSense: a wearable gestural interface. In *ACM SIGGRAPH ASIA Sketches*, K. Anjyo, Ed., ACM (2009), 60558.
25. Molyneaux, D. et al. Interactive Environment-Aware Handheld Projectors for Pervasive Computing Spaces. In *Pervasive Computing* (2012), 197–215.
26. Nakatsuma, K., and H. Shinoda. Touch interface on back of the hand. In *ACM SIGGRAPH Emerging Technologies* (2011), 4503.
27. Oikonomidis, I., N. Kyriazis, and A. A. Argyros. Full DOF tracking of a hand interacting with an object by modeling occlusions and physical constraints. In *IEEE ICCV* (Nov. 2011), 2088 – 2095.
28. Pamplona, V. The image-based data glove. In *Symposium on Virtual and Augmented Reality* (2008), 204–211.
29. Prados, E., and O. Faugeras. Shape From Shading. In *Handbook of Mathematical Models in Computer Vision*, N. Paragios, Y. Chen, and O. Faugeras, Eds. Springer, 2006, 375–388.
30. Rekimoto, J. GestureWrist and GesturePad: unobtrusive wearable interaction devices. In *IEEE ISWC*, vol. 5 (2001), 21–27.
31. Romero, J., H. Kjellstrom, and D. Kragic. Hands in action: real-time 3D reconstruction of hands in interaction with objects. In *IEEE ICRA* (2010), 458–463.
32. Saponas, T. S. et al. Enabling always-available input with muscle-computer interfaces. In *ACM UIST*, no. 38 (2009), 167–176.
33. Starner, T. et al. The gesture pendant: a self-illuminating, wearable, infrared computer vision system for home automation control and medical monitoring. In *IEEE ISWC* (2000), 87–94.
34. Starner, T., J. Weaver, and A. Pentland. A wearable computer based American sign language recognizer. In *IEEE ISWC*, vol. 1 of *ISWC '97* (1997), 130–137.
35. Vardy, A. The wristcam as input device. In *IEEE ISWC* (1999), 199 – 202.
36. Villar, N. et al. Mouse 2.0. In *ACM UIST* (New York, New York, USA, Oct. 2009), 33–42.
37. Wang, R. 6D hands: markerless hand-tracking for computer aided design. In *ACM UIST* (2011), 549–558.
38. Wise, S. et al. Evaluation of a fiber optic glove for semi-automated goniometric measurements. *Int. J. Rehabil. Res. 27*, 4 (1990), 411–424.
39. Zhang, Z. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell. 22*, 11 (2000), 1330–1334.