

## A STEERABLE MULTITOUCH DISPLAY FOR SURFACE COMPUTING AND ITS EVALUATION

PANAGIOTIS KOUTLEMANIS, ANTONIOS NTELIDAKIS, XENOPHON ZABULIS,  
DIMITRIS GRAMMENOS and ILIA ADAMI

*Foundation for Research and Technology – Hellas (FORTH)  
Institute of Computer Science, N. Plastira 100  
Vassilika Vouton, GR-700 13 Heraklion, Crete, Greece  
{koutle, ntelidak, zabulis, grammenos, iadami}@ics.forth.gr*

Received 28 January 2013

Accepted 19 March 2013

Published 20 December 2013

In this paper, a steerable, interactive projection display that has the shape of a disk is presented. Interactivity is provided through sensitivity to the contact of multiple fingertips and is achieved through the use of a RGBD camera. The surface is mounted on two gimbals which, in turn, provide two rotational degrees of freedom. Modulation of surface posture supports the ergonomics of the device but can be, alternatively, used as a means of user-interface input. The geometry for mapping visual content and localizing fingertip contacts upon this steerable display is provided, along with pertinent calibration methods for the proposed system. An accurate technique for touch detection is proposed, while touch detection and projection accuracy issues are studied and evaluated through extensive experimentation. Most importantly, the system is thoroughly evaluated as to its usability, through a pilot application that was developed for this purpose. We show that the outcome meets real-time performance, accuracy and usability requirements for employing the approach in human computer interaction.

*Keywords:* Interactive surface; multitouch; projector-camera system; touch detection; touch localization; surface computing; steerable surface.

### 1. Introduction

The emerging trend of smart environments entails the need for direct interaction with non-instrumented physical surfaces. In this context, often referred as “surface computing”,<sup>1</sup> systems combine the projection of a user interface on a surface (e.g., tabletop, wall), with visual sensing of finger contacts with the surface to provide multitouch interaction. The use of non-instrumented surfaces simplifies the application and maintenance of such systems. Recent availability of consumer depth cameras has reinforced the interaction capabilities upon virtually any surface, as the shape of interaction surfaces and the location of user hands can be accurately found in 3D.

Surface computing has the potential for a wide range of applications and has already been applied in exhibitions and education,<sup>2,3</sup> because interactive and augmented surfaces tend to capture the interest of observers. Due to this property, they have been often employed as stand-alone exhibits, related to an exhibition or museum.<sup>4,5</sup> In this work, we explore the issues arising when enabling surface computing upon dynamically moving surfaces and, in particular, upon a disk surface mounted on two concentric gimbals, providing two degrees of freedom (DOF). The symmetry and flexibility of the particular setup can support a wide range of applications, while its manipulation is simple and intuitive. Therefore, the pertinent system exhibits the potential of being suitable for access by the general public, such as the visitors of a museum.

Besides the provision of multitouch interaction upon non-instrumented steerable surfaces, such an achievement can serve two distinct functions. First, the two rotations can be used as a means of providing user input to intuitively browse with content, such as for example when one rotates a periscope to access a panorama of views. Second, the flexibility and extensive range of projection poses supported by the system can be used in order to dynamically personalize the physical properties of an interactive projection surface. This allows the user to adjust the interactive surface to her ergonomic preferences and needs, such as for example when one adjusts a lectern to a suitable posture. In this work, we extend the early version of the system<sup>6</sup> in multiple ways. Improvements involve the accuracy of fingertip localization and thorough usability evaluation, which quantitatively validates the suitability of the system for a wide range of surface computing tasks.

In the remainder of this section the design and operation of the proposed system are overviewed and conventions, regarding the definition of coordinates and properties of the proposed display, are defined. The rest of the paper is organized as follows. In Sec. 2 related work is reviewed. Real-time disk pose estimation is described in Sec. 3. The necessary calibration steps for the proposed projector-camera system are provided in Sec. 4. The proposed methods for establishing a display upon the disk and sensing fingertip contact upon it are described in Sec. 5. In Sec. 6, experiments which evaluate the accuracy of sensing and display but, most importantly, the usability of the proposed approach are presented. In Sec. 7, conclusions are provided.

### **1.1. System overview**

The system employs a circular planar surface, or a disk, mounted on two gimbals (see Fig. 1). The outer gimbal rotates about the vertical axis at an angle  $\theta$  (yaw). The inner gimbal is horizontal, dependent on the outer gimbal, and rotates along with it; the disk is mounted at joints  $\vec{q}_{1,2}$ , and has an elevation (pitch) angle of  $\phi$ . These axes intersect at the center  $\vec{c}$  of the disk. A projector above the disk fully covers it with its projection. An RGBD camera overlooks this scene acquiring its depth map  $D$ . The two gimbals can be freely rotated by the user. Due to the finite

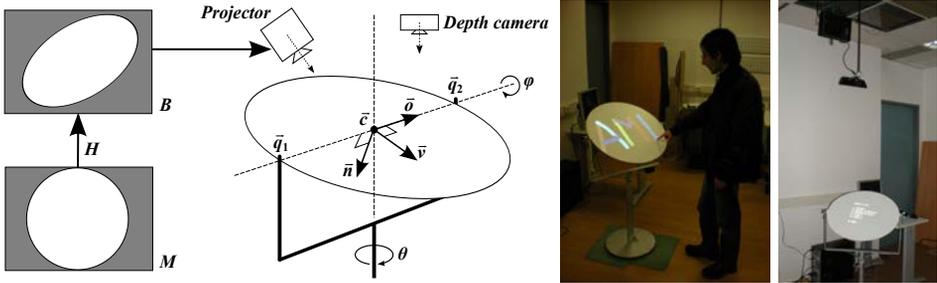


Fig. 1. (Color online) System overview. *Left, middle*: illustration of system geometry and data flow. The virtual display  $M$  is mapped upon the disk devoid of projective distortions by transforming  $M$ , according to the pose of the disk, prior copying it to the projector buffer  $B$ . Fingertip contacts are mapped back in  $M$  as multitouch events. *Right*: two photographs of a user testing a “finger-paint” pilot application, where drawings remain on the disk regardless of its posture, and an image of the system setup.

resolution of the camera and the projector, the operation of the system is limited at very oblique angles, as then, the disk corresponds to very few pixels in the camera and the projector.

The system creates the following user interface metaphor. Let  $M$  be a virtual display buffer that renders a circular display. The projector renders  $M$  upon the disk devoid of projective distortions, as if  $M$  was a multitouch display upon the disk’s surface. This is achieved by continuously estimating the disk’s pose and updating the projection appropriately. The pitch and yaw estimates can be used as additional user interfaces, by associating their values to two application variables.

Thus, a central system component is the real-time estimation of the disk’s pose from depth map  $D$ . At each camera frame, depth map  $D$  is updated yielding a new pose estimate. To support brisk and accurate interactivity upon the surface, it is essential that this operation is performed in real-time and robustly. By distorting  $M$  according to this pose, prior its copy into the projector’s pixel buffer  $B$ ,  $M$  appears undistorted on the disk. Intuitively, rotating the disk while  $M$  is displaying a static image would create the illusion of the image being “painted” upon the disk. In the projection, the row axis of  $M$  is aligned with the disk’s intrinsic horizontal rotation axis  $\vec{o}$ . Correspondingly, the column axis  $\vec{v}$  lies on the disk surface and is perpendicular to  $\vec{o}$ . When user hands are in contact with the surface, touch events are generated and attributed with the 3D contact coordinates of this contact. These coordinates are transformed into  $M$ ’s, 2D, reference frame, implementing a multitouch display upon the disk.

## 2. Related Work

To date, several touch-based interactive surfaces exist. The main emphasis has been placed upon the development of interactive screen, wall, and tabletop surfaces. Such systems typically support stylus based or hand based interaction, with some of them

providing augmented reality, as well as, single or multitouch functionalities. Input sensors may provide 2D or 3D information and be either integrated in the surface or placed above it, while visual display is typically provided by projectors.

## 2.1. Surface computing

Several research prototypes for touch-driven interactive surfaces can be found in the literature. “SmartSkin”<sup>7</sup> introduces a capacitive sensing architecture. A mesh-shaped antenna that covers a planar surface, together with a front projection unit form a tangible interactive surface. The system is sensitive to fingertip contact and supports multitouch interaction, as well as, augmented reality functionalities on objects, but which are required to have attached unique capacitance tags. “InteracTable”<sup>8</sup> introduces a design that facilitates use from multiple users around it and provides input from stylus or fingers touching a sensitive plasma display panel. “PlayAnywhere”<sup>9</sup> introduces a quick to install, front projection display. An infrared sensitive camera and an infrared light illuminant are rigidly placed on the projector at a fixed height. Localization of one finger per hand is achieved via a shadow detection algorithm.

Han<sup>10</sup> presents a rear-projection interactive system, equipped with infrared LEDs and an infrared sensitive camera on the back of an acrylic glass sheet. The latter serves as the interactive surface. It makes use of *frustrated total internal reflection* (FTIR) on the material for multiple fingertips localization, providing multitouch sensing. Gross *et al.*<sup>11</sup> propose another rear-projection “FTIR-based surface” design, that supports cooperative multitouch functionality for up to seven users. Gaver *et al.*<sup>12</sup> present the “Drift Table” as a case study of an interactive surface in the form of a coffee table for lucid engagement. Interaction is achieved via weight sensing on the surface of the table. Aside the research area, commercially available products exist.<sup>13–15</sup>

The aforementioned systems use a static planar surface as an interaction surface. One of the first systems to project on a rotating surface was the Perpsecta non-interactive display,<sup>16</sup> which employed projection of images on a rapidly rotating very oblique disk surface. Additionally, there have been some attempts to move beyond typical touch surfaces and experiment with alternative materials and setups. For example, Rakkolainen and Palovuori<sup>17</sup> employed laser scanning to support (single) touch and walk-through interaction with a FogScreen projection screen, consisting of flowing air with a little visible humidity. In another example, Virolainen *et al.*<sup>18</sup> have created an interactive ice-wall installation with a multitouch surface built from ice.

In early approaches towards augmented interactive surfaces,<sup>19,20</sup> the dynamic component concerned steering a projector to display upon the surface of choice. The display was adapted using prior modeling of the surfaces and limited hand interaction was based on the input of a color camera. More recent approaches have used a more detailed 3D model of the whole scene, availing the ability to project at

virtually any geometry of surfaces,<sup>21</sup> but used a stylus instrumented with an infrared beacon to enable single user interaction. The recent growth of depth cameras enabled the dynamic modeling, augmentation and touch interaction upon arbitrary surfaces.<sup>22</sup> Still, a training time is required in order for the system to model the interaction surfaces which, additionally, should remain static during interaction. The proposed approach constantly estimates the interaction surface and, thus, does not require an adaptation time.

Another aspect of dynamically moving interaction surfaces is that the location or pose of the surface itself can avail valuable information to the user interface. Song *et al.*<sup>23</sup> present a system, where a coarse estimate of the inclination of a hand-held surface (a piece of cardboard) provides input to an interactive game. Other systems<sup>24,25</sup> use a similar surface to explore maps. These systems use a conventional camera and rely on visual markers to estimate the surface pose. This makes them sensitive to marker occlusions by user hands or illumination artifacts. Other recent approaches augment planar surfaces, such as a physical desktop. Margetis *et al.*<sup>2</sup> recognize and augment pages of a marker-less book on a desktop using image recognition. Pages are treated as planar interactive surfaces, where the user can interact with them using a stylus. In another approach,<sup>3</sup> a front projection system augments both the surface on which a book is placed, as well as the book's pages.

## **2.2. *Designing interactive exhibits***

Durbin<sup>26</sup> describes the design process and observation results of interpretative devices integrated within the displays of the British Galleries of the Victoria and Albert Museum. Lehn *et al.*<sup>27</sup> examine the ways in which visitors encounter and experience exhibits and how these experiences are shaped and affected by social interaction. Hope *et al.*<sup>28</sup> focus on issues of family interaction and cooperation in a technological-augmented museum. Walter<sup>29</sup> and Heath *et al.*<sup>30</sup> provide observation study results from the use of electronic guides and interactive exhibits respectively. Moreover, they identify several problems and trade-offs between interactive media use and social interaction. Hall and Bannon<sup>31</sup> propose a number of heuristic design guidelines targeted to creating interactive museum exhibits for children. Knipfer *et al.*<sup>32</sup> present a framework for understanding informal learning in science exhibitions and explore the learning potential of related advanced applications.

## **2.3. *Interactive exhibits in museums***

The “Re-Tracing the Past” exhibition of the Hunt Museum was designed to show how novel interactive computer technologies could be introduced into a museum setting.<sup>33</sup> It comprised two room-sized spaces, one enabling visitors to explore mysterious objects, and another to record their personal opinion about them. The “Fire and the Mountain” exhibition comprised four hybrid exhibits<sup>5</sup> aiming to promote awareness about the cultural heritage of the people living around the Como Lake. The Austrian Technical Museum in Vienna opened a digitally augmented exhibition

on the history of modern media.<sup>34</sup> ARoS, an art museum in Denmark, employed four interactive exhibits targeted to an exhibition of the Japanese artist Mariko Mori.<sup>35</sup> The Ragghianti Foundation held an exhibition entitled “Puccini Set Designer”<sup>36</sup> that used new technologies to convey to the audience Puccini’s work as set designer.

#### **2.4. *The proposed work***

Most of the above approaches offer limited, or no touch-based interaction. The proposed work overcomes such limitations by employing a depth camera, which casts the approach insensitive to illumination artifacts and provides the basis for the multitouch interaction capabilities provided by the system.

Moreover, the proposed approach supports steering of the display surface which, in turn, enhances its ergonomics for a wide-range of users and, also, provides an additional means of user input. To achieve this, the pose of the disk is estimated in real-time, using a procedure that is tolerant to occlusions from user hands.

Most importantly, and in contrast to the majority of the approaches found in the literature, the proposed approach is thoroughly evaluated as to its accuracy of display and sensing, as well as, to its usability of operation. In this way, it is shown that the outcome is suitable for application in human computer interaction by generic audiences.

### **3. Disk Pose Estimation**

Disk pose estimation, is used both in the real-time operation as well as the calibration of the proposed system. It is comprised of the two processes described below. The first estimates the plane that the disk lies upon, despite outliers arising from user hands and noisy pixels in  $D$ . To meet real-time requirements, the method is parallelized in the GPU. The second disambiguates disk yaw when the disk is approximately parallel to the ground plane.

#### **3.1. *Parallel and robust plane estimation***

To estimate disk pose the plane  $\mathcal{E}$  is robustly fit to 3D points originating from depth map  $D$ . Only points originating from an elliptical region of interest (ROI) within depth map  $D$ , are considered. This ROI is large enough to enclose the entire disk and is predicted from camera calibration and disk geometry (see Sec. 4.1). Plane  $\mathcal{E}$  has equation  $\vec{n} \cdot (\vec{x} - \vec{c}) = 0$ , where  $\vec{n}$  is the normal of the plane. The spherical coordinates of  $\vec{n}$ ,  $\phi$  and  $\theta$ , indicate disk pose. A rotation of  $\pi$  about the horizontal axis is considered to bring the disk to the same posture and, thus,  $\phi \in [0, \pi/2)$  and  $\theta \in [0, 2\pi)$ , where  $\phi = 0$  corresponds to a posture parallel to the ground plane.

Significant amounts of outlier points are included in the data. Some occur due to sensor noise. Others occur as within the ROI more surfaces besides the disk are imaged, such as the user’s hands or body. A robust plane estimation is obtained using RANSAC.<sup>37</sup> A threshold of 1 cm distance to the plane is set to characterize a point as an inlier.

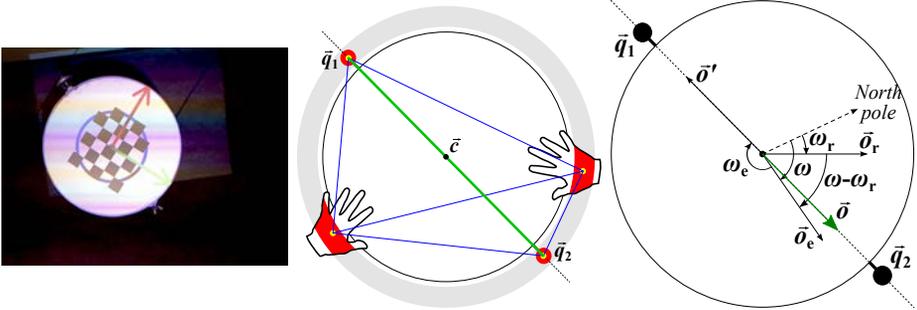


Fig. 2. (Color online) Disk yaw disambiguation *Left*: When the disk is parallel to the ground plane, yaw angle  $\theta$  becomes ambiguous and the projected image appears randomly rotated about  $\vec{n}$ . *Middle*: Horizontal axis estimation (top view). The search area around the disk periphery, is grayed. The segments connecting the discovered centroids (yellow circles) are marked blue. The disk axis (green segment) is the segment with the least distance from  $\vec{c}$ . *Right*: Horizontal axis direction disambiguation (top view). Using the current, possibly noisy, compass reading  $\omega$ , an estimation  $\vec{\sigma}_e$  is calculated by rotating  $\vec{\sigma}_r$  by  $\omega - \omega_r$ , about the gravity axis. The axis  $\vec{\sigma}'$ , estimated from 3.2.1, is inverted if the angle  $\omega_e$  is greater than  $90^\circ$ , resulting to  $\vec{\sigma}$  (green vector).

Conventionally, RANSAC iterates by selecting a random triplet of points and evaluating the number of inliers for the plane they define, until it finds a good fit or a maximum number of iterations is reached. The method is parallelized in the GPU by performing all trials in parallel threads and selecting the triplet with the most inliers. Finally, using least squares, a plane is fit to all the inlier points to this plane, which constitutes the final result. By convention, the normal of this plane is set to be in the direction of gravity.

A singularity, is encountered when  $\phi = 0$ . Then, any value of  $\theta$  produces the same  $\vec{n} = [0\ 0\ 1]^T$ . In our setup, accuracy of pose estimation begins to deteriorate due to obliqueness, when  $\phi$  becomes greater than  $80^\circ$ . For this reason, the value of  $\theta$  is determined using the method in Sec. 3.2.

### 3.2. Disk yaw disambiguation

In the case where the disk is paralleled to the ground plane, yaw disambiguation is broken down to two processes described below. The first estimates the horizontal axis of the disk. The second specifies the direction of the horizontal axis. An overview of the method is shown in Fig. 2.

#### 3.2.1. Horizontal axis estimation

When the disk is (approximately) parallel to the ground plane ( $\phi \approx 0$ ), the value of  $\theta$  becomes ambiguous. To disambiguate its pose when  $\phi$  is approximately 0, the horizontal axis of the disk  $\vec{\sigma}$  is found by estimating the line through the 3D locations of the inner gimbal joints,  $\vec{q}_1$  and  $\vec{q}_2$ . These joints are detected in  $D$  and their 3D locations extracted by the corresponding 3D depth values.

Candidate points for this detection are sought in the periphery of the ellipse (or circle) at which the disk appears in  $D$ . As above this ellipse is predicted for the particular pose from camera calibration and disk geometry (see Sec. 4.1). In addition, the 3D coordinates of candidate points are required to approximately validate the current plane equation  $\mathcal{E}$ . In  $D$ , candidate points are grouped into blobs by a Connected Component Labeling process. Each blob is represented by its centroid, which comprises a candidate point.

Spurious candidates can arise from user hands occurring at the periphery of the disk. The pair of centroids selected as  $\vec{q}_1$  and  $\vec{q}_2$  is the one at which the two candidates are diametrically opposed across the disk center  $\vec{c}$ . By considering the 3D coordinates of the candidate pair of centroids, the pair that defines the line segment with the least distance from  $\vec{c}$  is selected.

### 3.2.2. Horizontal axis direction

The estimate of the horizontal axis (see above, Sec. 3.2.1) does not specify the actual axis' direction. For this reason, this direction is disambiguated using an external hardware module.

The module is split over two distinctive units: a transmitter device and a receiver device. The transmitter is a portable, battery operated device, mounted on the disk structure in such a way that it can rotate along with the disk, about the gravity axis. Its main component is the *Honeywell HMC6352 Digital Compass Solution*, which uses a 2-axis magnetic sensor to detect the current azimuth  $\omega$ . The value of  $\omega$  refers to the North magnetic pole. The compass binary output is transmitted to the receiver device using an RF transmitter, at a rate of 2 Kbps. The receiver decodes the received binary data to an ASCII string, containing a human readable representation of  $\omega$ , in the range of  $[0, 2\pi)$ . This string is forwarded to the controlling machine, through a USB port.

A software module, parses the incoming string and uses the azimuth value to disambiguate the direction of  $\vec{\sigma}$ . During calibration, the module stores the current azimuth as  $\omega_r$  and the current horizontal axis as  $\vec{\sigma}_r$  at the reference disk pose. At runtime, a coarse estimation  $\vec{\sigma}_e$  of  $\vec{\sigma}$  is obtained by rotating  $\vec{\sigma}_r$  by  $\omega - \omega_r$ , about the gravity axis. Finally,  $\vec{\sigma}$  is inverted if the angle between vectors  $\vec{\sigma}$  and  $\vec{\sigma}_e$  is greater than  $90^\circ$ .

The magnetic sensor's accuracy degrades as metal objects or devices such as mobile phones get close to the transmitter device. Due to this reason, as well as the sensor's low detection rate, the device is not used in cases other than when  $\phi$  is approximately  $0^\circ$ .

## 4. Calibration

System calibration includes the estimation of center  $\vec{c}$  and the spatial modeling of the projection, both in the depth camera's coordinate frame. The additional, color camera is used only during the second part of this calibration (see Sec. 4.2) providing

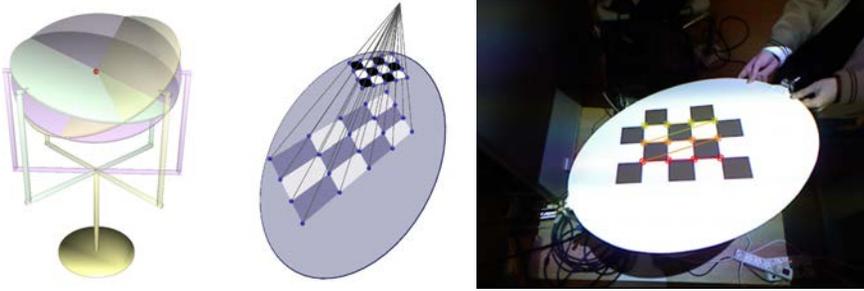


Fig. 3. (Color online) Calibration geometry. *Left*: The intersection of all plane estimates during calibration yields an estimate of center  $\vec{c}$ . *Middle*: By projecting a calibration pattern and finding the 3D coordinates of its reference points, the projector is calibrated. *Right*: The projected pattern detected in image  $I$ .

image  $I$ . We conveniently use the *Kinect* depth camera that already incorporates this additional sensor. A calibration of the depth and color cameras is assumed, based on Ref. 38. We have developed a calibration process that is as quick, simple and intuitive as possible. During this process, the user is only required to freely rotate the disk along the vertical and horizontal axis. The system extracts the necessary data and auto-calibrates.

#### 4.1. Disk center

Center  $\vec{c}$  is estimated from the depth maps acquired, while the disk is freely rotated about both its axes.

At each frame, the method in Sec. 3.1 estimates the plane approximating the disk. Angles  $\phi$  and  $\theta$  are discretized, at a step of  $1^\circ$ , yielding  $k = 360 \times 90$  potential planes. A  $k \times 4$  matrix, is employed as a lookup table (LUT). Each time a plane is estimated, its 4 parameters are copied into the corresponding LUT entry. In this way, very similar planes are considered once.

The result is obtained as the intersection of all, let  $m$ , estimated planes and computed as the point that minimizes the sum of its squared distances from all planes (see Fig. 3, left). The equation of the plane mentioned in Sec. 3.1 is expanded to  $\alpha x + \beta y + \gamma z - \delta = 0$ . As  $\vec{c}$  should ideally validate the equations of all planes, we seek  $\vec{x}$  so that  $\|A\vec{x} - B\|$  is minimized.  $A$  is a  $m \times 3$  matrix containing in each row the first 3 parameters  $\alpha_i, \beta_i, \gamma_i$  of the estimated planes,  $B$  is a  $m \times 1$  matrix containing values  $\delta_i$  in each row, and  $i \in [1, m]$ . A least-squares solution is then found through the SVD decomposition of  $A$ .

Typically, less than a minute is required to fill a sufficient proportion ( $\approx 20\%$ ) of the LUT for an accurate estimate of  $\vec{c}$ .

#### 4.2. Projector calibration

The projector operation is modeled by a  $3 \times 4$  perspective projection matrix  $P$  that predicts the pixel coordinates in  $B$  that will illuminate a given homogeneous point

in 3D space. Given the high quality optics of projectors and the usage of only its central portion of the display, the lens distortion of the projector is assumed to be negligible. During the calibration process both color  $I$  and depth  $D$  images of the sensor are used. Using their calibration, a registration of these two images is obtained and, thus, 3D coordinates of the surfaces imaged in the color camera can be obtained.

Matrix  $P$  is estimated as follows. The image of a calibration target, a checkerboard, is constantly projected upon the disk. This image appears distorted upon the disk, in accordance with its posture (see Fig. 3, middle, right). As in Sec. 4.1, the disk is freely rotated in both angles, while the system is acquiring frames. For each frame, the projected checkerboard upon the disk surface is detected in  $I$  and the image coordinates of its corners identified using a checkerboard detector.<sup>39</sup> The 3D coordinates of these points are then found at the same pixel coordinates of, registered, image  $D$ . Finally, these 2D-3D correspondences are used in a standard method<sup>40</sup> to estimate matrix  $P$ . Inclusion of lens distortion in this optimization is left for future work.

## 5. The Interactive Display

The real-time system implementing the interactive display, is comprised of two modules that operate at the camera's frame rate. Using the module in Sec. 3.1 a pose estimate of the disk is available for each camera frame. Using this estimate, the transformation that undistorts display  $M$  upon the disk is computed. At the same time, a second module estimates finger contacts with the disk and produces multitouch user interface events in  $M$ 's, 2D, coordinate frame.

### 5.1. Display

This module computes the distortion that  $M$  must undergo before projection so that (i) it appears undistorted upon the disk and (ii) each of its pixels illuminates the same physical region upon the disk, regardless of its posture. The distortion that an image undergoes when projected upon the disk is modeled through a  $3 \times 3$  homography matrix, let  $H$ .

The system's displayed content is limited to a circular region of  $M$ . This region is centered at  $M$ 's center and its diameter is equal to the minimum of  $M$ 's dimensions, so that the maximum area of  $M$  is utilized. After distorting  $M$ , the region appears undistorted upon the disk, exactly covering its surface. The limits of the displayed content are predicted as the corners of a hypothetical square. This square encloses the region and is aligned with  $M$ 's row and column axes. Let  $Q_1$ ,  $Q_2$ ,  $Q_3$  and  $Q_4$  be the corners of this square, in a clockwise order (see Fig. 4).

The homography  $H$  can be defined if we determine its required physical limits on the disk surface and predict their coordinates in  $B$ . These limits are predicted as the corners of a hypothetical square that lies upon  $\mathcal{E}$ , encloses the disk, and is

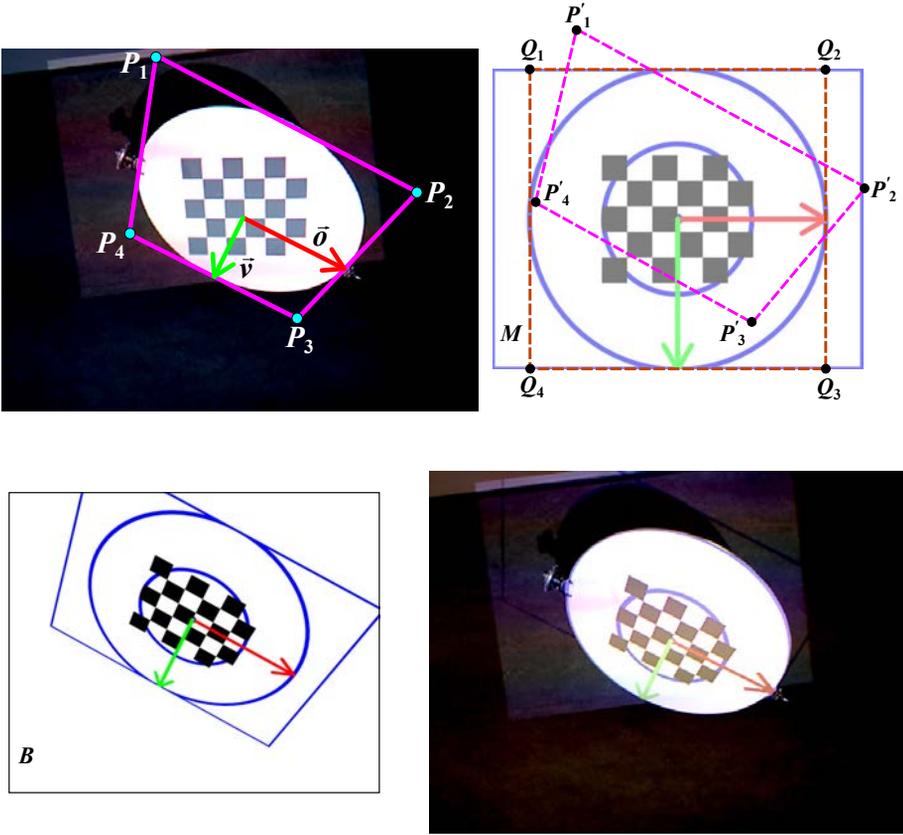


Fig. 4. (Color online) Display geometry (left to right, top first). (a) The disk enclosed in a hypothetical square. Without the warping effect, the projected checkerboard image does not appear aligned with the disk intrinsic axes. (b) Establishing point correspondences for homography computation:  $M$  is shown annotated with (i) the four points that correspond to the corners of the hypothetical square ( $Q_1, Q_2, Q_3, Q_4$ ) and (ii) the corresponding corners from the image on the left, projected on the projector buffer ( $P'_1, P'_2, P'_3, P'_4$ ). (c) Warping of  $M$  according to  $H$  makes  $M$  appear undistorted upon the disk and aligned with its intrinsic axes. (d)  $M$  is set to contain a checkerboard image and visualize axes  $\vec{o}$  (red) and  $\vec{v}$  (green); when warped and projected on the disk, checkers appear undistorted and axes accurately aligned with the disk intrinsic axes for the observer.

aligned with its intrinsic axes,  $\vec{v}$ ,  $\vec{o}$  which are computed from the estimates of  $\phi$  and  $\theta$ .

Using  $\vec{c}$ ,  $\vec{o}$ ,  $\vec{v}$  and assuming a known disk radius  $\rho$ , these corners are:

$$P_1 = \vec{c} - \vec{o} * \rho - \vec{v} * \rho \quad (1)$$

$$P_2 = \vec{c} + \vec{o} * \rho - \vec{v} * \rho \quad (2)$$

$$P_3 = \vec{c} + \vec{o} * \rho + \vec{v} * \rho \quad (3)$$

$$P_4 = \vec{c} - \vec{o} * \rho + \vec{v} * \rho \quad (4)$$

in a clockwise order. Through the use of  $P$ , the coordinates of these points in  $B$  are predicted, let  $P'_1, P'_2, P'_3$  and  $P'_4$ . At the current disk posture, the pixels of  $B$  at these coordinates illuminate the corresponding points upon  $\mathcal{E}$ .

By associating  $P'_{\{1,2,3,4\}}$  and  $Q_{\{1,2,3,4\}}$ , the homography  $H$  is calculated. Image  $M$  is warped using  $H$  and the result is copied into  $B$ .

## 5.2. *Fingertip contact detection and localization*

This process detects which physical points of the disk are in contact with the fingertips of the user, and converts the coordinates of these points in  $M$ 's coordinate frame. Besides the fingertip area which is in contact with the disk, a single point of contact is also estimated. This point of contact is utilized as the location of the touch event, when required, for tasks such as pressing a button on the display or drawing shapes in painting applications. As the accuracy of this estimation is important to fine manipulation or drawing tasks, two different approaches are considered and comparatively evaluated (in Sec. 6).

### 5.2.1. *Contact area estimation*

Fingertip contact localization upon the disk is found as follows. Given the current plane  $\mathcal{E}$ ,  $\vec{c}$ , and  $\rho$ , the ellipse upon which the disk projects in  $D$  is predicted. The pixels of  $D$  within this ellipse are transformed into 3D points.

For each of these points, its distance to  $\mathcal{E}$  is computed. Similarly with Wilson,<sup>41</sup> we use two thresholds to detect contact. The first ( $d_{\max}$ ) indicates if a pixel is closer to the camera than the disk surface. However, only this constraint will include points belonging to the user's arm as well. The second threshold ( $d_{\min}$ ) eliminates points that are overly far from the surface to be considered part of object in contact. This double thresholding allows to gather 3D points  $V$  within a specific distance range from the disk surface. Due to sensor noise, when the value of  $d_{\max}$  is small (i.e.  $\approx 1$  cm) points of the disk are detected as being above the disk and, spuriously, detected as touch points.

We tackle this noise in a two stage process. At a first stage, we set  $d_{\max}$  in a desired precision ( $\approx 0.5$  cm) and use an appropriate value on  $d_{\min} = \tau_h$  (i.e. 10 cm) to ensure that points belonging to  $V$  correspond to a user's hand. Then, we perform a 3D Connected Component Labeling on  $V$  and we keep only the biggest components  $C$  assuming to be the user's hands. Points grouped to small components are considered noise and removed from  $V$ . At a second stage, we double threshold again on  $V$ , using a stricter value on  $d_{\min} = \tau_f$  (i.e. 3.4 cm). The remaining points in  $V$  will ideally belong to parts of the user's fingers near fingertips. We perform the 3D Connected Component Labeling once more, to group them into  $F_j$  components (see Fig. 5),  $j \in [1, w]$ , where  $w$  is the number of components. We compute a point of contact  $\vec{q}_j$  for every component  $F_j$  assuming to be a fingertip. We associate  $\vec{q}_j$  to its corresponding  $F_j$  and we track fingers with ids that are consistent across frames.

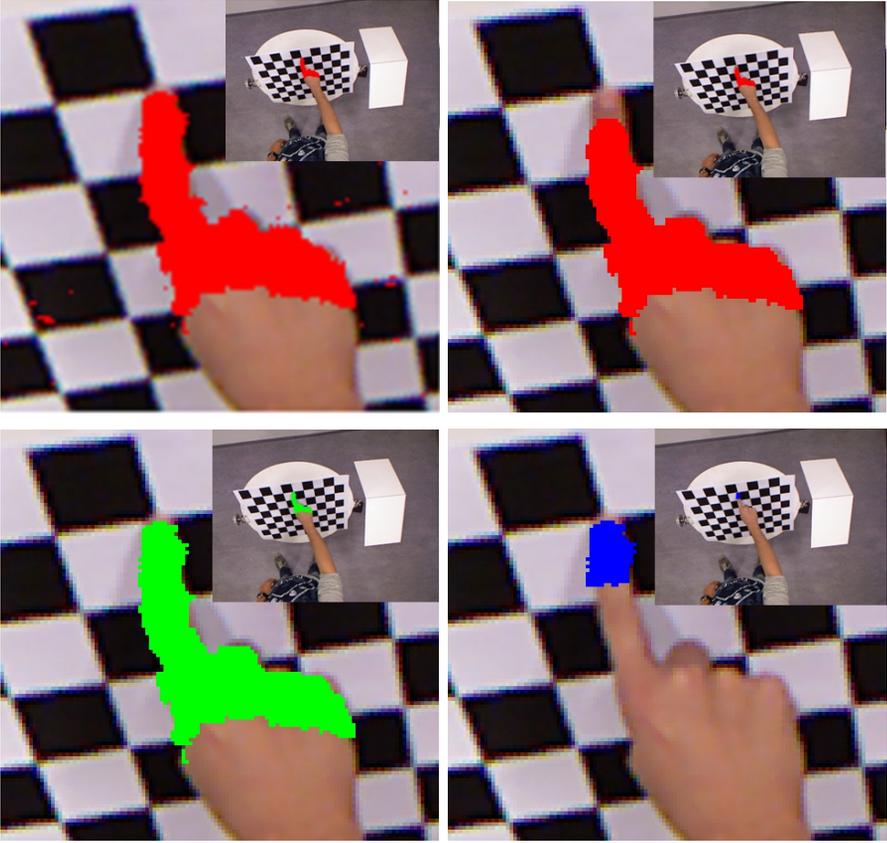


Fig. 5. (Color online) Removing spurious 3D points (left to right, top first). (a)  $V$  points within the range of  $d_{\max} = 0.5$  cm and  $d_{\min} = 10$  cm, projected on the corresponding sensor’s image with red color. Spurious points are gathered due to the bold  $d_{\min}$  threshold. (b)  $V$  within the range of  $d_{\max} = 1.5$  cm and  $d_{\min} = 10$  cm. The conservative  $d_{\max}$  threshold removes spurious points. This undesirably removes points near finger tip as well. (c) We reduce noise in  $V$  by keeping the biggest  $C$  components (green color). We use the same  $d_{\max}$  and  $d_{\min}$  with  $a$ , preserving points near fingertip. (d) Points belonging to  $C$ , thresholded to a finer range ( $d_{\max} = 0.5$  cm,  $d_{\min} = 3.4$  cm), then grouped to  $F$  component corresponding to a part of the finger near finger tip (blue color).

Finally, we use  $P$  (see Sec 4.2) to project  $\vec{p}_j$  3D points to image coordinates on the projector buffer  $M$ . Then we use  $H^{-1}$  (see Sec 5.1) to distort these coordinates in accordance to the disk’s pose (see Fig. 6). We evaluate the following approaches to approximate  $\vec{q}_j$ .

### 5.2.2. Point of contact

We experimented with two approaches to approximate points of contact  $\vec{q}_j$ , the *Planar Touch* and the *Mean Touch* approach. We apply these approaches on every  $F_j$  component to associate a point of contact. Each approach uses points belonging to a component  $F_j$  at a time.

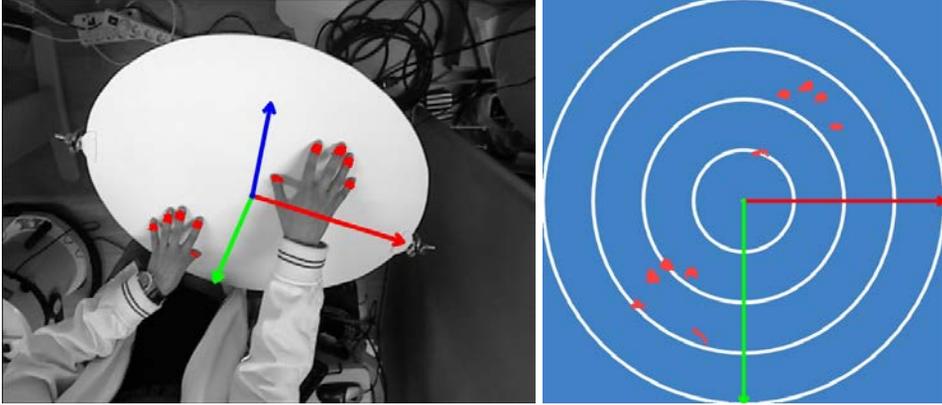


Fig. 6. (Color online) Contact detection and localization. *Left*: a user touches the surface with all his fingers and the projections of the detected contact points on  $I$  are superimposed with red. For reference, estimates of  $\vec{o}$ ,  $\vec{c}$ , and  $\vec{n}$  are also superimposed in red, green, and blue respectively. *Right*: contact blobs in the  $M$ 's reference frame; column (red) and row (green) axes correspond to  $\vec{o}$  and  $\vec{c}$ .

*Mean Touch approach.* The disk plane  $\mathcal{E}$  is computed (see Sec. 3.1) and defined in the normal constant form Eq. (5).

$$\mathcal{E} : \vec{n}_1 \cdot \vec{x} = c_1 \quad (5)$$

A component  $F_j$  consists of 3D points  $F_j = \{\vec{x}_1, \vec{x}_2, \dots, \vec{x}_n\}$ . We find the centroid  $\vec{m}_f$  for  $F_j$  Eq. (6). We approximate a point of contact  $\vec{q}_j$  for  $F_j$  as the orthogonal projection of the centroid  $\vec{m}_f$  of  $F_j$  on the disk plane  $\mathcal{E}$  Eq. (7).

$$\vec{m}_f = \frac{1}{n} \sum_{i=1}^n x_i \quad (6)$$

$$\vec{q}_j = \vec{m}_f - (\vec{n}_1 \cdot \vec{m}_f - c_1) \vec{n}_1 \quad (7)$$

The *Mean Touch* approach is illustrated in Fig. 7 (top).

*Planar Touch approach.* In this approach, we try to exploit the direction of the finger as it touches the disk plane. We fit a plane  $\mathcal{E}_f$  on the  $F_j$  3D points using least squares. The plane  $\mathcal{E}_f$  is normalized and defined in the normal constant form Eq. (8).

$$\mathcal{E}_f : \vec{n}_2 \cdot \vec{x} = c_2 \quad (8)$$

We find the 3D line  $L$  at which the disk plane  $\mathcal{E}$  intersects with  $\mathcal{E}_f$ :

$$L : \vec{L}(k) = \vec{l}_0 + k\vec{u}, \quad (9)$$

where  $\vec{u} = \vec{n}_1 \times \vec{n}_2$  is the line direction (not normalized),  $k$  is a parameter and  $\vec{l}_0$  a point on the line.  $\vec{l}_0$  belongs to both  $\mathcal{E}$  and  $\mathcal{E}_f$  planes and is found as follows:

$$\vec{l}_0 = a\vec{n}_1 + b\vec{n}_2, \quad (10)$$

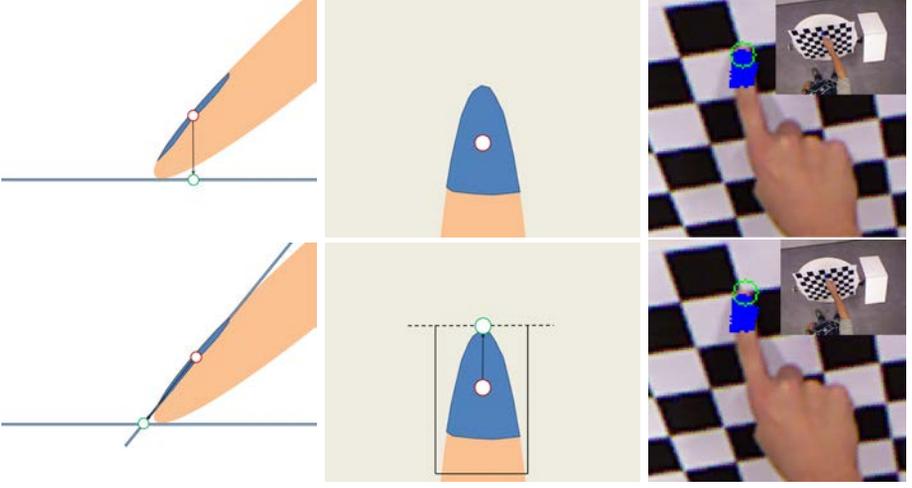


Fig. 7. (Color online) Touch point estimation of a finger touching the disk. *Top: Mean Touch* approach. Side and top view sketch of the finger touching the table. The blue area corresponds to 3D points of a component  $F_j$ . We approximate the touch point  $\vec{q}_j$  (green circle) as the orthogonal projection of the centroid  $\vec{m}_f$  (red circle) of the component  $F_j$  on the disk plane  $\mathcal{E}$ . The right image highlights with a green circle a touch estimate of this approach on a sensor's image. *Bottom: Planar Touch* approach. We fit a plane  $\mathcal{E}_f$  (rectangle) on the 3D points of the  $F_j$  component. We find the projection  $\vec{m}'_f$  (red circle) of  $\vec{m}_f$  on  $\mathcal{E}_f$ . We approximate the touch point  $\vec{q}_j$  (green circle) as the projection of  $\vec{m}'_f$  on the line where the disk plane  $\mathcal{E}$  intersects with  $\mathcal{E}_f$ .

where

$$a = \frac{c_2 \vec{n}_1 \cdot \vec{n}_2 - c_1 \|\vec{n}_2\|^2}{(\vec{n}_1 \cdot \vec{n}_2) - \|\vec{n}_1\|^2 \|\vec{n}_2\|^2} \quad (11)$$

and

$$b = \frac{c_1 \vec{n}_1 \cdot \vec{n}_2 - c_2 \|\vec{n}_1\|^2}{(\vec{n}_1 \cdot \vec{n}_2) - \|\vec{n}_1\|^2 \|\vec{n}_2\|^2}. \quad (12)$$

We find the centroid  $\vec{m}_f$  for  $F_j$  using Eq. (6) and its orthogonal projection  $\vec{m}'_f$  on  $\mathcal{E}_f$  Eq. (13).

$$\vec{m}'_f = \vec{m}_f - (\vec{n}_2 \cdot \vec{m}_f - c_2) \vec{n}_2. \quad (13)$$

The point of contact  $\vec{q}_j$  for the component  $F_j$  is approximated as the orthogonal projection of  $\vec{m}'_f$  on  $L$ , as follows:

$$\vec{q}_j = \vec{l}_0 + t\vec{u}, \quad (14)$$

where

$$t = \frac{\vec{u} \cdot (\vec{m}'_f - \vec{l}_0)}{\vec{u} \cdot \vec{u}}. \quad (15)$$

The *Planar Touch* approach is illustrated in Fig. 7 (bottom).

## 6. Experiments

The experiments focused in validating the suitability of the proposed system as a multitouch interactive surface. In the first set of experiments, the accuracy of contact detection for single and multiple fingertips was measured. In the second, the accuracy of disk pose estimation and projection upon the disk were measured. The third set of experiments, concerned simple interaction and measured the adequacy of the system in simple tasks, such as successfully sensing a fingertip contact or fingertip-tracing multiple lines simultaneously. In the fourth set of experiments, the effectiveness of the system in more complex tasks that are required in interactive applications was assessed by an extensive usability evaluation.

The experiments were performed on a conventional PC equipped with *nVidia GeForce GTX 260* 1.2 GHz GPU. The system is adequately fast to keep up with the image acquisition and image projection rates, which occur at 60 Hz. In fact, in off-line experiments the system’s operation rate is  $\approx 120$ . In our setup, the disk has a radius of  $\rho = 30$  cm and is placed 150 cm from the ground. A Microsoft Kinect sensor ( $640 \times 480$  pixel resolution) and a projector ( $1280 \times 800$  pixels) overlook the disk from a height of 149 cm and 102 cm, respectively.

### 6.1. *Fingertip contact localization accuracy*

In this experiment, the localization accuracy of touch events was measured in 3D, for multiple postures of the disk. A checkerboard that was printed upon a planar sheet of paper was mounted upon the disk. For each posture, the disk is initially imaged without any user or object occluding it. Using a checkerboard detector<sup>39</sup> and a calibration method,<sup>40</sup> the corners of the checkers are detected in  $I$  and the relative pose of the camera to the checkerboard plane is estimated. In this way, the 3D locations of the checker corners were found and were considered as ground truth.

Users were instructed to touch with their finger the inner corners at a particular order, while the system was acquiring images. The error was quantified as the Euclidean distance between the 3D touch estimate and a 3D corner. To compare the two methods proposed in Sec. 5.2.2, both of them were employed to estimate the 3D touch location and their respective errors under the same captured datasets of users performing the task. The experiment was repeated for 3 users for 5 elevation angles in the range of  $[-50^\circ, 50^\circ]$ , in steps of  $25^\circ$ . A  $10 \times 8$  checker grid with 63 inner corners was used.

Figure 8, (right) shows the mean error among all users for each elevation angle, for each of the two 3D touch estimation approaches. Table 1 shows the mean error and the standard deviation cumulatively for all elevation postures. We deduce that the planar approach yields slightly more accurate 3D contact localization estimates.

### 6.2. *Accuracy of disk pose estimation and projection*

In this experiment, the accuracy of disk pose estimation was evaluated. The disk was placed at known poses and the error was measured as the difference between

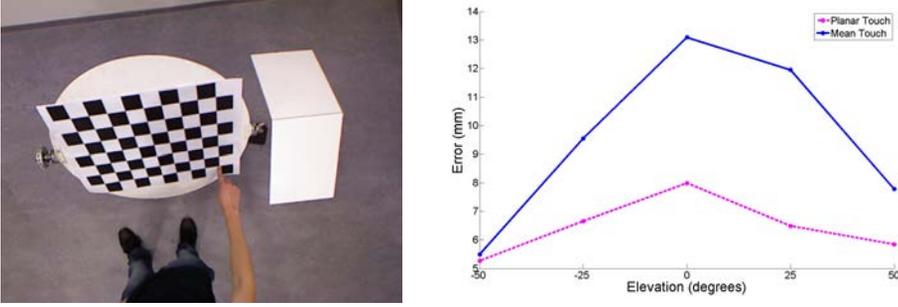


Fig. 8. (Color online) *Left*: original image  $I$  from the experiment, where a user touches a checker corner, at  $50^\circ$  elevation posture of the disk. *Right*: The errors for each method plotted as a function of the elevation angle, for each method.

Table 1. Fingertip contact localization error (3D).

Method	Mean Error (mm)	Standard Deviation (mm)
Planar	6.444	4.076
Mean	9.566	9.566

the ground truth values and the estimation of the elevation and azimuth angles  $\phi$  and  $\theta$ , respectively. The disk's elevation range from  $0^\circ$  to  $60^\circ$  was discretized in 7 steps of  $10^\circ$  each. The full azimuth range ( $360^\circ$ ) was discretized in 18 steps of  $20^\circ$ . Ground truth was measured using a digital inclinometer, temporarily mounted on the disk. No occlusions of the disk occurred in the experiment. The results are presented in Fig. 9. The disk was securely firmed at each of these angles and the system's elevation and azimuth angles estimation was recorded.

The accuracy of projection, based on the disk's pose estimation was assessed as follows. A checkerboard pattern was displayed in  $M$  and correspondingly projected

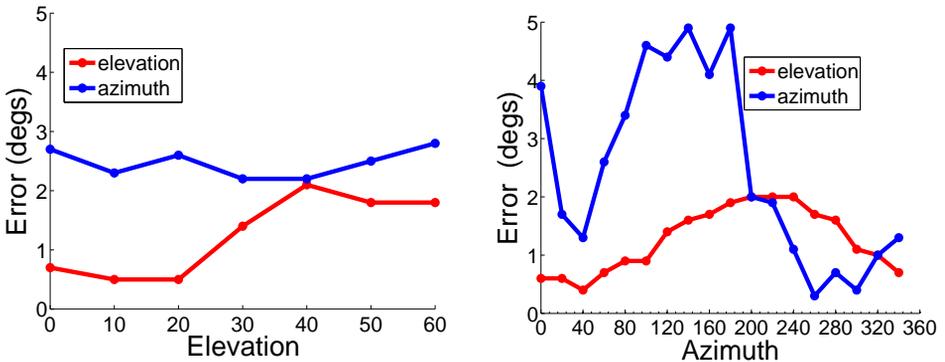


Fig. 9. (Color online) Accuracy experiments. *Left, middle*: Elevation and azimuth error in degrees per elevation (left) and per azimuth (right).

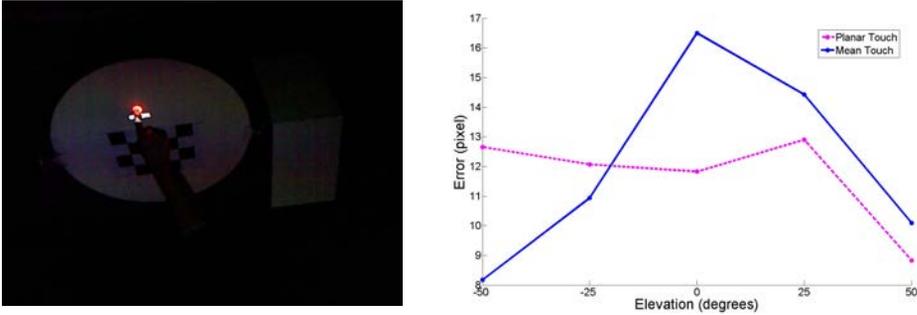


Fig. 10. (Color online) Interaction experiments. *Left*: Image from the experiment where the user touches with its finger a projected dot on the disk. *Right*: The errors for each method plotted as a function of the elevation angle, for each method.

upon the disk. The aspect ratio of checkers was confirmed to be 1 upon the disk, for the above range of elevations, meaning that  $M$ 's appearance on the disk is devoid of projective distortions for that range of elevations. We observed that the aspect ratio was preserved, up to angles as steep as  $70^\circ$ . However, images projected in angles more oblique than  $\approx 50^\circ$  were poor in quality and, thus, limited operation of the system up to that obliqueness.

### 6.3. Interaction experiments

To assess the system's suitability as a touch display two interaction experiments were performed. In the first, the accuracy of *mean* and *planar touch* detection was measured. In the second, tracking of multiple fingers in simultaneous contact was evaluated. In essence, the evaluation concerned the accuracy of registration between the projected display and contact localization estimates. The two experiments were performed by 5 users each that were naive to the experimental hypotheses.

In the first experiment, users had to touch a sequence of dots, of 0.75 cm radius, that were appearing upon the disk (see Fig. 10, left). The users were instructed to touch the dots at their centers. The error was measured by comparing the distance of the touch event in  $M$  with the projected location of the dot (see Fig. 11). The sequence consisted of 70 dots, laid out on 7 concentric circles, centered at the disk's center. The entire surface of the disk was used. The dots were presented to the users one at a time and in random order. The experiment was repeated for 5 elevation angles, in the range of  $[-50^\circ, 50^\circ]$ , in steps of  $25^\circ$ . Figure 10 (right) presents the average error in pixels for each elevation angle. Table 2 shows the mean error and the standard deviation cumulatively for both approaches for all elevation postures in pixel coordinates of the projection buffer. Both approaches were evaluated under the same captured datasets of users performing the task.

In the second experiment, the users were presented with 5 vertical, parallel line segments, which were instructed to trace with their fingertips (see Fig. 12).

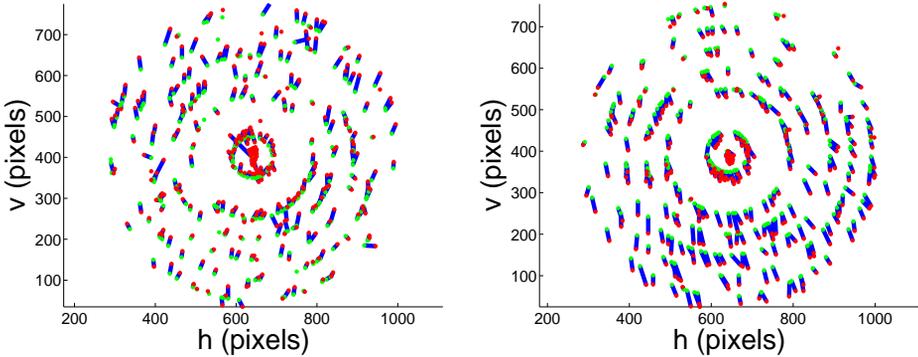


Fig. 11. (Color online) Interaction experiments. The two plots illustrate the projected dots (green) and the estimated contact locations (red) connected with a blue line for  $\theta = 25^\circ$  and  $\theta = -50^\circ$ . Pixel coordinates of the projector buffer are transformed to *mms* on the coordinate system of the disk plane.

Table 2. Fingertip contact localization error (2D).

Method	Mean Error (pixels)	Standard Deviation (pixels)
Planar	11.657	7.874
Mean	12.027	5.529

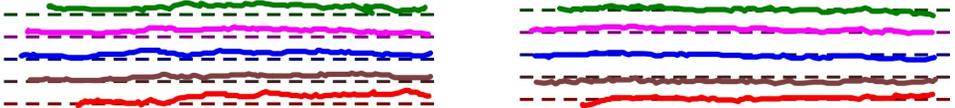


Fig. 12. (Color online) Interaction experiments. Two plot of the projected lines (straight dashed lines) and traced contact trajectories for the same posture. Both examples are shown in *M*'s reference frame.

The stripes appeared at a distance of 3 cm from each other. The experiment was repeated for each of the same 5 elevation angles of the previous experiment (see above). The trajectories were recorded and using the stripes as ground truth, the Multiple Object Tracking Accuracy (MOTA) metric<sup>42</sup> was calculated to evaluate the accuracy of multiple simultaneous contacts to be 1 in the whole range of system operation, that is when  $\phi \in [0^\circ, 50^\circ]$ .

#### 6.4. Usability evaluation

To test the system in a realistic setting, through a set of representative user tasks, an interactive application was developed, which was installed in a working prototype.

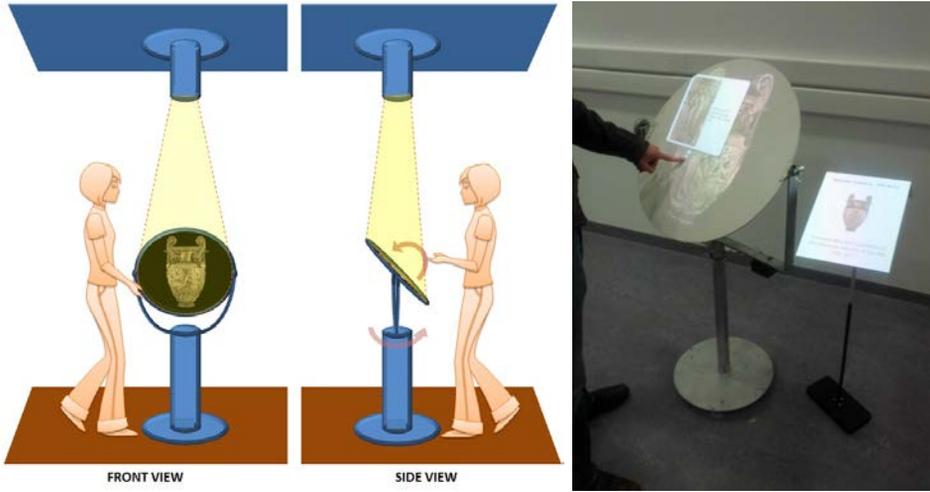


Fig. 13. (Color online) Pilot application. *Left*: Schematic view of the pilot application installation. *Right*: Working prototype of the pilot application. During system’s operation, a projected image of the artifact is warped appropriately on the rotating disk. Additional context information is projected on a side panel.

#### 6.4.1. Pilot application

The installation comprises the following physical components (see schematic view in Fig. 13):

- High, on the ceiling there is a projector and a *MS Kinect* device, which are connected to a PC, located at a certain distance from the installation, so that it is not visible to the users.
- On the floor, there is a stand upon which resides a metal disk. The disk can be rotated in full  $360^\circ$  about its vertical and horizontal axis.

The application supports the exploration of ancient artifacts in  $360^\circ$ . The application loads datasets of ancient artifacts. Each artifact has been placed on a turn and photographed from the side, in 360 steps of  $1^\circ$ . For each step, the direction of illumination was mechanically modulated to follow an arc trajectory above the artifact in 20 steps. Using images corresponding to the same set of illuminations, a sparse reconstruction of the artifact was obtained using Ref. 43 and regions of interest were defined upon this point, corresponding to “hotspots” on the surface of the artifact.

In the application, a photograph of the artifact is initially projected on the surface of the metal disk, as can be seen from the angle that the metal disk is rotated. By rotating the disk around the vertical axis ( $\theta$ ), the user can see 360 different views of the artifact, as if the actual object was placed behind the disk’s surface, thus creating a 3D visualization effect. By rotating the disk surface around the horizontal axis ( $\phi$ ), the projected artifact is warped appropriately for a natural visual result.

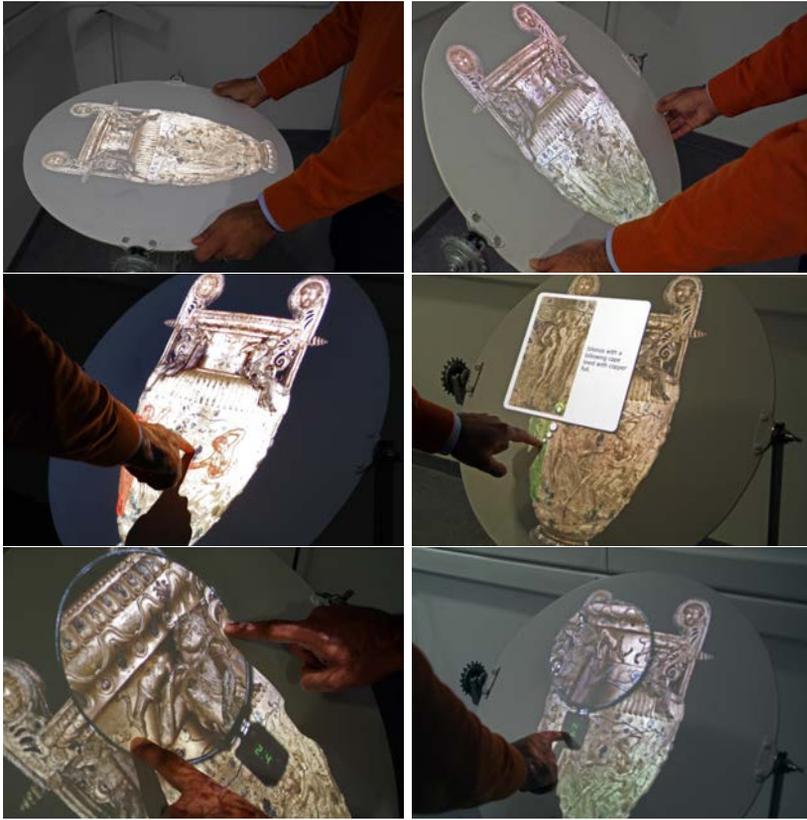


Fig. 14. (Color online) Instances of the pilot application working. *Top Left, Right:* When the user rotates the disk around the horizontal axis ( $\phi$ ), the projected artifact is warped appropriately for a natural visual result. *Middle Left, Right:* When the user touches the metal surface, hotspot areas of the current view are revealed. When touching such an area related information is presented. *Bottom Left, Right:* A two-finger zoom in gesture, triggers a magnifying glass that allows zoom in/out functionality. The user can move the magnifying glass by dragging its lens and change the magnification level by dragging its grip left or right. A two-finger zoom out gesture closes the magnifying glass.

When the user touches the metal surface, hotspot areas of the current view are revealed. Upon touching a hotspot, related information is presented. Additionally a zoom in/out functionality is supported. Using the universal two-finger zoom in gesture, a magnifying glass appears. The user can move the magnifying glass by dragging its lens. The magnification level changes by dragging its grip left or right. The magnifying glass closes by performing a two-finger zoom out gesture. Instances of the working prototype are presented in Fig. 14.

A preliminary user-based evaluation was conducted on the prototype of the 360° interaction system with 14 users. The objective of the evaluation was twofold. On the one hand it aimed at assessing the system's overall usability as perceived by the users, and on the other hand, at capturing the users general attitude towards this

interaction method as a way to present a museum's exhibit content. The evaluation was conducted by a usability expert and took place in a laboratory space of FORTH. A camera was set up in the room in order to capture users' reactions and comments during the evaluation. The evaluator remained in the room during the sessions to provide any assistance to the users and to guide them through the series of tasks that they had to complete during the evaluation.

#### *6.4.2. Participants*

As mentioned earlier, a total of 14 users participated in the evaluation, 6 of which were females and 8 were males. The average age of the participants was 32 years old. The targeted audience for this evaluation was users with intermediate to high familiarity with technology in general and specifically with touch screen systems. At this stage of development, it was decided to have the system evaluated by users that already have a notion of how touch screen applications work and have set expectations on how such a system should react to typical touch and multitouch gestures. The reasoning behind this decision was that such users would evaluate the usability of the system based on previous experiences with other touch screen applications, and thus would be able to identify more easily any areas where the interaction deviates from set standards than novice users would. However, in the future a second evaluation will be planned with novice users in order to have a comparison of the perceived usability and general attitude towards the system between expert and novice users.

#### *6.4.3. Evaluation process*

At the beginning of the evaluation session, the participants were welcomed. They were introduced to the system and the evaluation goal. They were then asked to carry out a series of 6 tasks with the system. The tasks at hand included locating the areas of content interest (hotspots) on the artifact representation, reading the information presented, activating the magnifier tool and panning it across the surface to see details of the artifact, changing the level of magnification, and rotating the surface to see the artifact from all 360 angles. Jacob Nielsen's User Success Rate<sup>44</sup> method was used to measure the percentage of tasks that users completed successfully. Task accomplishment was rated as a success if it was completed on the first or second trial, partial success if the task was completed on the third trial and failure if the task was completed on the fourth trial or more, or if the user exhibited frustration and annoyance. The Think-Aloud process was used during the evaluation and the participants were requested to express verbally their thoughts, comments, suggestions, and opinions throughout the completion of each task. At the end of the evaluation, the participants were asked to comment on their overall experience with the system. They were also asked if there was any additional functionality they would have liked to see in the system that was not currently supported and

whether they would like to use this system in a museum exhibition setting. After the evaluation, the participants were requested to fill-out John Brooke's System User Satisfaction (SUS) Questionnaire,<sup>45</sup> which consists of 10 Likert type questions and calculates the system's overall usability.

#### 6.4.4. *Evaluation results*

*User comments analysis from Think-Aloud<sup>46</sup> process.* The 360° system received very positive comments by the majority of participants who found the concept “refreshing”, offering a very interesting approach of presenting information about an artifact in a museum exhibit. They thought that the system not only engages the user, but also presents the information in a very effective way. They particularly liked the fact that there was no separation between the system's functionality and the content of the presentation. They particularly liked that they could touch on certain areas of interest (hotspots) on the artifact and read more detailed information about them. When asked if they would prefer a more traditional way of presenting information (e.g. written information about the artifact next to its digital, but non-interactive representation), they all said that they would prefer to be able to interact with the digital representation through touch gestures. They were also all impressed with the disk rotation functionality to view the artifact from all aspects, however, they also pointed out that they would prefer it if they did not have to move around the artifact while rotating the disk because it was not natural for them to do so. In addition, they thought that having to move around the artifact while holding the disk could cause problems in the context of a crowded exhibit room. When asked what other additional functionality they would have liked to see, 10 out of 14 said they would like to be able to rotate in the horizontal and vertical axis the 3D artifact through touch gestures without having to move around it. 7 users also expressed that they would want to be able to get a top or a bottom view of the artifact by tilting the disk to a certain degree angle. 1 user suggested adding social networking links right on the information text containers to be able to email the picture to a friend or post it on social networks.

*SUS Score.* At the end of each evaluation session, the participant was asked to fill out the John Brooke's System User Satisfaction Questionnaire (SUS). The questionnaire consists of 10 questions 5 of which are positive statements and 5 negative statements. The participants marked each question with a number from 1 to 5, where 1 = strongly disagree and 5 = strongly agree. The scoring of the positive questions is produced by taking the number that the user marked and subtract 1 point from it, while the scoring of the negative questions is produced by subtracting the marked number from 5. The SUS score the 360° system received was 75%, above the average which is considered to be 68%. From the users' responses, it is evident that the users thought that the system was easy to use and learn in general, it did not require much effort from the participants to learn how to use it and its functionality was consistent with other touch applications. As it can be

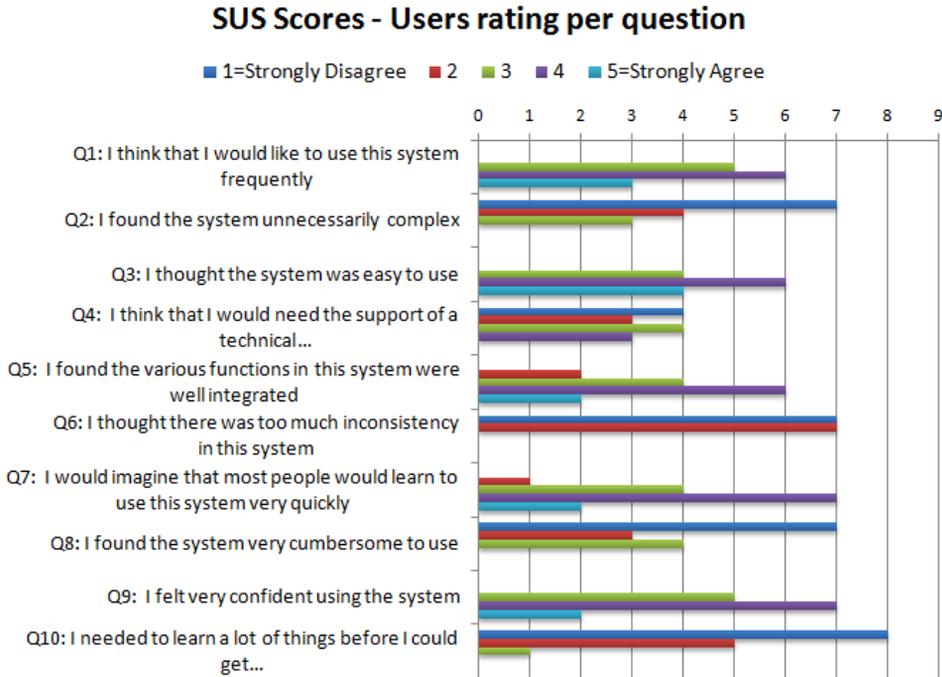


Fig. 15. (Color online) SUS scores. Ratings per question.

seen in Fig. 15, most of the positive questions received ratings of 4 and 5 meaning that the participants leaned towards agreement with those statements, while the majority of the negative questions received ratings of 1 and 2, meaning that the majority of the users did not agree with them.

*User Success Rate.* The participants were asked to perform sequentially the following tasks:

- Task 1. Find the hotspots on the artifact.
- Task 2. Choose one of the hotspots and read the information provided. After reading the information, close the hotspot window.
- Task 3. Using your two fingers activate the magnifier tool (*the gesture was demonstrated to the users*).
- Task 4. Drag the magnifier tool over the image to view details of the artifact.
- Task 5. Change the degree of magnification. What is the highest value of magnification allowed by the tool?
- Task 6. Using two fingers deactivate (close) the magnifier tool (*the gesture was demonstrated to the users*).

To calculate the User Success Rate for this application, the evaluator marked each task with an “S” (Success) if the user completed the task successfully on the first or second trial, a “PS” (Partial Success) if the user completed the task

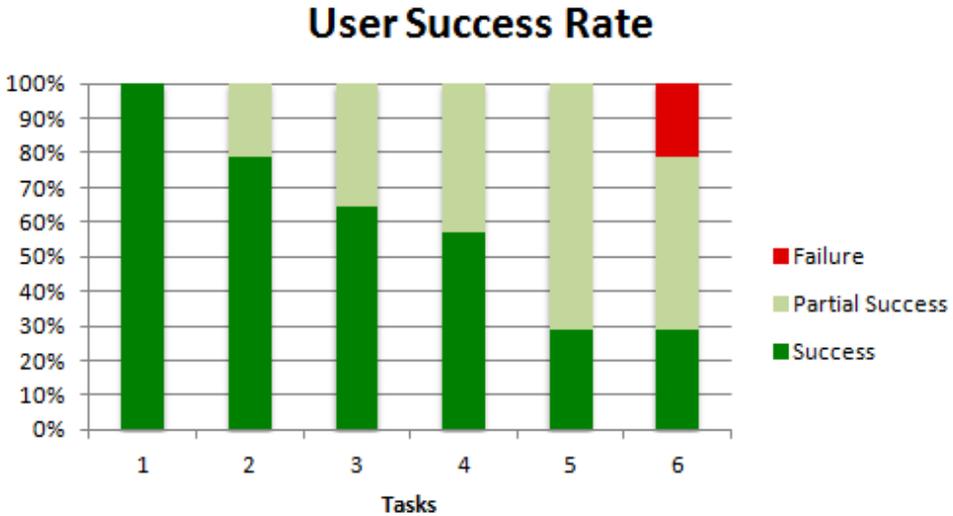


Fig. 16. (Color online) Percentage of tasks completed with success, partial success, and failure.

successfully, but on the third trial, and with an “F” (Failed) if it took the user more than three trials to complete the task or if the user expressed annoyance and frustration with the system. The successful attempts were awarded 1 point, the partially successful attempts were awarded 0.5 point, and the failed attempts received 0 points. Using Nielsen’s formula, the User Success Rate for the 360° system was 79%, which is considered a high rate well above the average of 50%. Figure 16, shows the percentage of the successful, partial successful, and failed attempts.

The tasks that caused the most trouble to the users were Tasks 5 and 6, which had to do with changing the level of magnification and dis-activating the magnifier tool. It seemed that if the magnifier tool handle was in the lower end of the surface, it was not as responsive to the touch and the users had to try more than a couple of times in order to change the level to the highest point of 4.0. This behavior was not noticed as much, if the magnifier was located in the center of the disk. In addition, the closing of the magnifying tool also caused some problems to the users. A few of them discovered on their own that they had to do the opposite gesture than the one they had to do in order to activate the tool, but still they were not sure from what distance they had to drag the fingers on the surface to dis-activate the tool. It seemed that the system was not very responsive to the gesture and the majority of the users had to try a few times before they managed to close it. They commented that they would have liked a simpler gesture to close it, such as simply clicking anywhere outside the tool or on an “X” button that would appear next to the tool. Another problem that was noticed, was during Task 2, which entailed the users having to select an area of interest (hotspot) to view its hidden content. Once the hotspot window appeared on the surface, the users were not sure if they could further click on it to get an augmentation of the depicted picture. The majority

thought that the little hand icon situated at the bottom of the picture was an action button and they pressed there hoping to augment the picture, whereas it was only a short animation that indicated that the picture was interact-able. When they were asked where they would click to close down the hotspot window, they all, without exception, said that they would click anywhere outside the container instead of inside the container which is how it currently behaves.

## **7. Conclusion**

A passive approach for the implementation of an interactive, steerable and multitouch display has been proposed and evaluated. The approach employs a depth camera and a non-instrumented 2-DOF rotating surface, which is used as the interactive display.

Applications of the proposed approach are found in the general domain of surface computing. Nevertheless, the ability of the system to apply the principles of surface computing to a steerable surface provides two novel capabilities. First, the values of the two rotations of the surface can be employed to acquire user input, which is quite intuitive due to the physical manipulation of the surface. Second, the posture of the display can be dynamically adjusted to accommodate the ergonomic needs of individual users, thus being ideal for installations used by a wide variety of users, such as in museums and exhibitions.

Thereby, the rotation of the display is central to the appreciation of the system and is, at the same time, a means of user input. For this reason, the accuracy by which disk pose is estimated and the accuracy by which the projected display is mapped upon the disk, have been thoroughly evaluated. The results from experiments show that disk pose is estimated very accurately despite the presence of sensor noise and occlusions due to user hands. Moreover, user interface input provided by angles  $\phi$  and  $\theta$  can be reliably used for fine operations. For example, in the pilot application modulation of viewpoint occurred smoothly and accurately when users moved the disk.

Two other significant properties of system usability are the robustness and the accuracy by which fingertip contact is detected and localized, respectively. Through the same policy of extensive evaluation they were found sufficiently accurate for conventional multitouch screen interaction. Indeed, this accuracy decreases at oblique angles of the disk, and originates mainly from the reduction in area that the disk undergoes in the sensor's (depth) image. At such angles, though, the projector's limitations in resolution are also encountered, as fewer pixels can be used to form an image upon the disk surface. Improvements could use steerable projectors<sup>20</sup> and sensors to compensate for this obliqueness, by rotating in accordance to the elevation ( $\phi$ ) of the disk.

A limitation of the approach is met in multitouch and multi-hand interaction and in particular when using more than one hand upon the surface. In such cases, user digits that may be in contact with the disk are occluded from the sensor and

missed. Nevertheless, this is a limitation common to any approach that employs a single view to detect fingertip contact. Though this topic could be addressed with additional sensors a topic of future work is to investigate whether a more retentive tracking of user hand position would suffice user requirements.

The above findings were validated through an in-depth usability evaluation which involved a wide diversity of users, engaged in a variety of characteristic surface computing tasks through the pilot application. The usability evaluation results confirm that the approach is highly valid, applicable and, due to the generic nature of the pilot application, suitable for a broad spectrum of applications.

## Acknowledgments

This work has been supported by the FORTH-ICS internal RTD Programme “Ambient Intelligence and Smart Environments”. The authors thank Athanassios Toutountzis for the integration of the digital compass and Spyros Paparoulis for the construction of the disk apparatus.

## References

1. Rowell, L., Scratching the surface, *netWorker* **10** (2006) 26–32.
2. Margetis, G., Zabulis, X., Koutlemanis, P., Antona, M. and Stephanidis, P., Augmented interaction with physical books in an ambient intelligence learning environment, *Multimedia Tools and Applications* (2012) 1–23.
3. Margetis, G., Ntelidakis, A., Zabulis, X., Ntoa, S., Koutlemanis, P. and Stephanidis, P., Augmenting physical books towards education enhancement, in *1st IEEE Workshop on User-Centred Computer Vision* (2013).
4. Robertson, T., Mansfield, T. and Loke, L., Designing an immersive environment for public use, in *Proc. of the Ninth Conf. on Participatory Design: Expanding Boundaries in Design*, Vol. 1 (New York, NY, USA, ACM, 2006), pp. 31–40.
5. Fleuret, F., Berclaz, J., Lengagne, R. and Fua, P., Multicamera people tracking with a probabilistic occupancy map, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **30** (2008) 267–282.
6. Zabulis, X., Koutlemanis, P. and Grammenos, D., Augmented multitouch interaction upon a 2-dof rotating disk, in Bebis, G., Boyle, R., Parvin, B., Koracin, D., Fowlkes, C., Wang, S., Choi, M. H., Mantler, S., Schulze, J. P., Acevedo, D., Mueller, K. and Papka, M. E. (Eds.), *ISVC* (1), Vol. 7431 of Lecture Notes in Computer Science (Springer, 2012), pp. 642–653.
7. Rekimoto, J., Smartskin: An infrastructure for freehand manipulation on interactive surfaces, in *CHI* (2002), pp. 113–120.
8. Streitz, N., Tandler, P., Muller-Tomfelde, C. and Konomi, S., Roomware: Towards the next generation of human-computer interaction based on an integrated design of real and virtual worlds, in J. Carroll (Ed.), *Human-Computer Interaction in the New Millennium* (Addison Wesley, 2001), pp. 553–578.
9. Wilson, A., Playanywhere: A compact interactive tabletop projection-vision system, in *UIST* (2005), pp. 83–92.
10. Han, J., Low-cost multi-touch sensing through frustrated total internal reflection, in *UIST* (2005), pp. 115–118.
11. Gross, T., Fetter, M. and Liebsch, S., The cuetable: Cooperative and competitive multi-touch interaction on a tabletop, in *CHI* (2008), pp. 3465–3470.

12. Gaver, W., Bowers, J., Boucher, A., Gellerson, H., Pennington, S., Schmidt, A., Steed, A., Villars, N. and Walker, B., The drift table: Designing for ludic engagement, in *CHI* (2004), pp. 885–900.
13. Microsoft: (Microsoft surface) <http://www.surface.com>.
14. Dietz, P. and Leigh, D., Diamondtouch: A multi-user touch technology, in *UIST* (2001), pp. 219–226.
15. SMART: Smart table (2008) <http://www.smarttech.com/>.
16. Chun, W., Napoli, J., Cossairt, O., Dorval, R., Hall, D., Purtell, T., Schooler, J., Banker, Y. and Favalora, G., Spatial 3D infrastructure: Display-independent software framework, high-speed rendering electronics, and several new displays, in *SPIE*, Vol. 5664 (2005), pp. 302–312.
17. Rakkolainen, I. and Palovuori, K., Laser scanning for the interactive walk-through fogscreen, in *VRST* (2005), pp. 224–226.
18. Virolainen, A., Puikkonen, A., Karkkainen, T. and Hakkila, J., Cool interaction with calm technologies: Experimenting with ice as a multitouch surface, in *ITS* (2010), pp. 15–18.
19. Pinhanez, C., Using a steerable projector and a camera to transform surfaces into interactive displays, in *CHI* (2001), pp. 369–370.
20. Kjeldsen, R., Pinhanez, C., Pingali, G., Hartman, J., Levas, T. and Podlaseck, M., Interacting with steerable projected displays, in *FG* (2002).
21. Jones, B., Sodhi, R., Campbell, R., Garnett, G. and Bailey, B., Build your world and play in it: Interacting with surface particles on complex objects, in *ISMAR* (2010), pp. 165–174.
22. Harrison, C., Benko, H. and Wilson, A., Omnitouch: Wearable multitouch interaction everywhere, in *UIST* (2011), pp. 441–450.
23. Song, P., Winkler, S. and Tedjokusumo, J., A tangible game interface using projector-camera systems, in *HCI* (2007), pp. 956–965.
24. Grammenos, D., Michel, D., Zabulis, X. and Argyros, A., Paperview: Augmenting physical surfaces with location-aware digital information, in *TEI* (2011), pp. 57–60.
25. Reitmayr, G., Eade, E. and Drummond, T., Localisation and interaction for augmented maps, in *ISMAR* (2005), pp. 120–129.
26. Durbin, G., Interactive learning in the british galleries, in *Interactive Learning in Museums of Art and Design* (2002).
27. vom Lehn, D., Heath, C. and Hindmarsh, J., Exhibiting interaction: Conduct and collaboration in museums and galleries, *Symbolic Interaction* **24** (2001) 189–216.
28. Hope, T., Nakamura, Y., Takahashi, T., Nobayashi, A., Fukuoka, S., Hamasaki, M. and Nishimura, T., Familial collaborations in a museum, in *Proc. of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09)* (New York, NY, USA, ACM, 2009), pp. 1963–1972.
29. Walter, T., From museum to morgue? electronic guides in roman bath, *Tourism Management* **17** (1996) 241–245.
30. Heath, C., Lehn, D. V. and Osborne, J., Interaction and interactives: Collaboration and participation with computer-based exhibits, *Public Understanding of Science* **14** (2005) 91–101.
31. Hall, T. and Bannon, L., Designing ubiquitous computing to enhance children’s interaction in museums, in *Proc. of the 2005 Conf. on Interaction Design and Children (IDC '05)* (New York, NY, USA, ACM, 2005), pp. 62–69.
32. Knipfer, K., Mayr, E., Zahn, C., Schwan, S. and Hesse, F. W., Computer support for knowledge communication in science exhibitions: Novel perspectives from research on collaborative learning, *Educational Research Review* **4** (2009) 196–209.

33. Ferris, K., Bannon, L., Ciolfi, L., Gallagher, P., Hall, T. and Lennon, M., Shaping experiences in the hunt museum: a design case study, in *Proc. of the 5th Conf. on Designing Interactive Systems: Processes, Practices, Methods, and Techniques (DIS '04)* (New York, NY, USA, ACM, 2004), pp. 205–214.
34. Hornecker, E. and Stifter, M., Learning from interactive museum installations about interaction design for public settings, in *Proc. of the 18th Australia Conference on Computer-Human Interaction: Design: Activities, Artefacts and Environments (OZCHI '06)* (New York, NY, USA, ACM, 2006), pp. 135–142.
35. Kortbek, K. J. and Grønbaek, K., Interactive spatial multimedia for communication of art in the physical museum space, in *ACM Multimedia Conf.* (2008), pp. 609–618.
36. Sparacino, F., Scenographies of the past and museums of the future: From the wunderkammer to body-driven interactive narrative spaces, in Schulzrinne, H., Dimitrova, N., Sasse, M. A., Moon, S. B. and Lienhart, R. (Eds.), *Proc. of the 12th ACM Int. Conf. on Multimedia* (New York, NY, USA, ACM, October 10–16, 2004), pp. 72–79.
37. Fischler, M. and Bolles, R., Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography, *Communications of the ACM* **24** (1981) 381–395.
38. Zhang, C. and Zhang, Z., Calibration between depth and color sensors for commodity depth cameras, in *ICME* (2011), pp. 1–6.
39. Vezhnevets, V., Velizhev, A., Chetverikov, N. and Yakubenko, A., GML C++ camera calibration toolbox (2011), <http://graphics.cs.msu.ru/en/science/research/calibration/cpp>.
40. Hartley, R. and Zisserman, A., *Multiple View Geometry in Computer Vision*, 2nd edn. (Cambridge University Press, 2004).
41. Wilson, A., Using a depth camera as a touch sensor, in *ACM Int. Conf. on Interactive Tabletops and Surfaces* (2010), pp. 69–72.
42. Bernardin, K. and Stiefelhagen, R., Evaluating multiple object tracking performance: The CLEAR MOT metrics, *Journal of Image and Video Processing* **2008** (2008) 1–10.
43. Snavely, N., Seitz, S. and Szeliski, R., Photo tourism: Exploring photo collections in 3D, in *SIGGRAPH* (2006), pp. 835–846.
44. Tullis, T. and Albert, B., Levels of success, in *Measuring the User Experience. Collecting, Analyzing, and Presenting Usability Metrics* (Morgan Kaufmann, 2008), pp. 70–72. <http://www.useit.com/alertbox/20010218.html>.
45. P. W. Jordan, B. Thomas, B. A. W. and McClelland, A. L., SUS: A quick and dirty usability scale, in *Usability Evaluation in Industry* (Taylor and Francis, 1996), pp. 189–194.
46. Nielsen, J., Thinking aloud, in *Usability Engineering* (Academic Press, 1993), pp. 195–199.