Vol. 14, No. 3: 35–55

Building Interactive Multi-Touch Surfaces

Johannes Schöning[⋆], Jonathan Hook[†], Nima Motamedi[◇], Patrick Olivier[†], Florian Echtler[∩], Peter Brandl[‡], Laurence Muller[⋉], Florian Daiber[⋆], Otmar Hilliges[δ], Markus Loechtefeld[⋆], Tim Roth[θ], Dominik Schmidt[τ], Ulrich von Zadow[ξ]

[⋆]DFKI GmbH, Germany; [†]Newcastle University, UK; [◇]Simon Fraser University, Canada; [∩]Technical University Munich, Germany; [‡]Upper Austria University of Applied Sciences, Austria; [⋉]University of Amsterdam, Netherlands; [δ]University of Munich, Germany; [θ]University of Zurich, Switzerland; [τ]University of Lancaster, UK; [ξ]Archimedes Solutions GmbH, Germany.

Abstract. Multi-touch interaction with computationally enhanced surfaces has received considerable attention in recent years. Hardware implementations of multi-touch interaction such as frustrated total internal reflection (FTIR) and diffused illumination (DI) have allowed for the low-cost development of surfaces. Although many of these technologies and associated applications have been presented in academic settings, the practicality of building a high-quality multi-touch–enabled surface, both in terms of the software and hardware required, are not widely known. We draw upon our extensive experience as developers of multi-touch technology to provide practical advice in relation to building and deploying applications upon multi-touch surfaces. This includes technical details of the construction of optical multi-touch surfaces, including infrared illumination, silicone compliant surfaces, projection screens, cameras, filters, and projectors, and an overview of existing software libraries for tracking.

1. Introduction

Multi-touch technology presents a wide range of new opportunities for interaction with graphical user interfaces, allowing expressive gestural control

1086-7651/09 $0.50 per page

Figure 1. The joy of multi-touch interaction.

and fluid multi-user collaboration through relatively simple and inexpensive hardware and software configurations. The technology itself has been available in different forms since the late 1970s. Multiple patents demonstrate how camera/sensor-based touch surfaces can be constructed [Johnson 72, Mueller 74, Mallos 82, Kasday 84, White 87]. Bill Buxton's multi-touch webpage [Buxton 08] provides a thorough overview of the underlying technologies as well as the history of multi-touch surfaces and interaction. However, it was Han's 2005 presentation [Han 05] of a low-cost camera-based multi-touch sensing technique based upon frustrated total internal reflection (FTIR) that truly highlighted the potential role multi-touch could play in the development of the next generation of human-computer interfaces. Han's system was both cheap and easy to build and was used to illustrate a range of creatively applied interaction techniques—his YouTube demonstration captured the imagination of experts and laymen alike.

In 2007 interest in multi-touch grew as Apple released details of the iPhone (http://www.apple.com/iphone), a mobile phone with a multi-touch screen as a user interface. The interface and interaction techniques of the iPhone received considerable media attention and brought multi-touch to the forefront of the consumer electronics market. Later in 2007, Microsoft announced their Surface multi-touch table (http://www.microsoft.com/surface). The Surface has the appearance of a coffee table with an embedded multi-touch interactive screen. In manner similar to the HoloWall [Matsushita and Rekimoto 97], the Surface has a diffuser attached to the projection surface and is illuminated from below with infrared light. Reflections of hands and objects are captured by cameras inside the table in an approach described as diffused illumination (DI). By utilizing a grid of multiple cameras, the Surface has a sensing resolution sufficient to track objects augmented with visual markers.

Considerable research has explored the benefits of multi-touch interaction [Dietz and Leigh 01, Rekimoto 02, Moscovich 06, Valli and Linari 08, Schöning et al. 08b, Moscovich and Hughes 08, Jung et al. 08, Moscovich and Hughes 08] and multi-touch surfaces have found their way into the futuristic visions of human-computer interaction seen in TV shows and movies (e.g.,*James Bond, Quantum Of Solace* and *The Day the Earth Stood Still* [Schöning et al. 09]).

Our goal is to enable graphics and interaction-design practitioners to embrace multi-touch by providing the basic knowledge required to *build your own* multi-touch surface. Many techniques, such as resistance-based–, capacitance-based–, or surface wave-touch screens, generally require industrial fabrication facilities. In contrast, we focus exclusively on optical approaches to multi-touch sensing as these can be built quickly and easily integrated into graphical user interfaces.

2. Optical-Based Touch Surfaces

Optical approaches to multi-touch use image processing to determine the location of interactions with the surface. These systems typically use infrared illumination, and due to their simple set-up have the potential to be very robust. In addition to FTIR and DI we discuss two other related, but distinct, approaches: laser light plane (LLP) and diffused screen illumination (DSI) [Schöning et al. 08a].

2.1. Frustrated Total Internal Reflection (FTIR)

Han's work in 2005 [Han 05], which utilised the principle of FTIR in multi-touch interaction, can be seen as the critical point in the development of such optical systems. The FTIR approach is based on optical total internal reflection within an interactive surface. Electromagnetic waves transmitted within an inner material are completely reflected at its boundary if

- the inner material has a higher refractive index than the outer material;
- the angle of incidence at the boundary between the materials is small enough.

The most common FTIR set-up has a transparent acrylic pane, with a frame of LEDs around its edges, into which infrared light is injected (see Figure 2 (left)). When the user touches the acrylic, the light escapes and is reflected at the finger's point of contact due to its higher refractive index; an infrared-sensitive camera can then clearly see these reflections. A basic set of computer-vision algorithms (see Section 3.3) is applied to the camera image to determine the location of the contact point. Since the acrylic is transparent a projector can be located behind the surface (near to the camera) yielding a back-projected multi-touch sensitive display (see Figure 2 (left)).

Diffuse illumination (DI) systems are similarly configured, with both a projector and an infrared-sensitive camera placed behind the projection surface. However, for DI, infrared lighting is also placed behind the projection surface causing the area in front of the surface to be brightly lit in the infrared spectrum. Consequently, the camera is capable of detecting the reflection of fingers

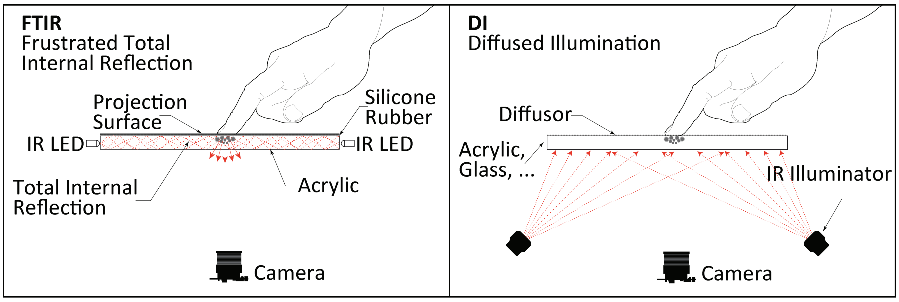

Figure 2. General set-up of a FTIR system (left). General set-up of a diffuse illumination system (right). The projector is mounted inline with the projector below/behind the surface.

and objects on, or in close proximity to, the surface (see Figure 2 (right)). Touch detection exploits the fact that the projection surface diffuses light, blurring objects at a distance. The main advantage of FTIR is that it allows very robust tracking of fingers; however, diffuse illumination additionally allows tracking of physical objects, which can be identified either by their shape or the use of fiducial markers [Costanza and Robinson 03] (easily recognizable markers) on the base of the objects. Furthermore, hovering gestures can also be recognized, and any transparent surface (such as safety glass) can be placed between the projection screen and the user since sensing does not rely on surface contact.

3. BYO Multi-Touch Surface

When designing and constructing an optical multi-touch surface a number of challenges need to be addressed. In this section we divide up these issues as they relate to both hardware and software and provide practical advice based on our own experiences of developing robust tabletop systems.

3.1. Hardware

The hardware of an optical multi-touch system is comprised of infrared illumination sources, silicone compliant surfaces, projection screens (or the use of LCDs), cameras, filters, and projectors.

3.1.1. Infrared Illumination

Both FTIR and DI require an infrared light source. Achieving the correct infrared illumination can be challenging and requires a knowledge of both the different methods of illuminating a surface and the different types of IR

LEDs (5mm, 3mm, SMD (synchronous mirror delay)) that are available commercially. Almost all existing IR-based set-ups employ light-emitting diodes (LEDs) as light sources. Two commonly used types of IR LEDs are Osram SFH4250 (SMD) and Osram SFH485 (5 mm). Whether SMD devices or standard LEDs are more appropriate depends on a number of factors; for example, if the LEDs have to be mounted to the rim of an acrylic glass plate, this is easier with SMDs, as it is possible to simply attach them to the rim with instant glue. After hardening, instant glue is chemically identical to acrylic glass and is therefore able to create a very strong, transparent bond. Mounting standard LEDs requires holes to be drilled into the material, which can be a time-consuming and error-prone process and should be undertaken with care.

One major problem for both FTIR and IR systems is their sensitivity to ambient IR light from the external environment. This can be mitigated by adding a small electronic circuit to the set-up which supplies short high-current pulses instead of a continuous low current. The pulse current is usually set high enough such that under sustained operation, the LEDs would be likely to suffer permanent damage after a few seconds. Typically, these pulses are given a duration of between a hundred microseconds and a few milliseconds. The high current level, which is possible during the short pulses, results in a much higher light output.

The pulse duration and the following cool down period should be kept as close to the manufacturer's specification as possible to prevent overheating of the LEDs. As modern computers are usually not equipped with the hardware or software to undertake such real-time control tasks, we suggest using a simple microcontroller (e.g., PIC or AVR) or the venerable 555 timer for pulse generation. A second-level switching element is also necessary to handle the high currents which flow through the LEDs. Field-effect transistors (FETs), such as the IRF512 logic-level FET, are particularly easy to integrate with logic circuits, and we suggest using these as second-level switches. A final precaution against LED damage is an ordinary fuse. A fuse with a *lower* rating than the expected pulse current should be inserted in series with the LEDs. Although more current will flow through the fuse than it is rated for, it is unlikely to blow during pulsed operation.

Pulsing the LEDs significantly increases total light output, but this in itself does not produce enough contrast with ambient light levels. Instead, the pulses need to be synchronized with the camera in such a way that

- one pulse is emitted for each camera frame;
- each pulse's duration is equivalent to the camera's exposure time.

As the LEDs are usually brighter by approximately one order of magnitude during the pulse, the contrast ratio with respect to environmental light is also significantly higher. If the camera exposure time is longer than a single pulse, stray light from the environment is accumulated during the cool-down period between pulses, decreasing the contrast ratio. However, in the continuous mode, the brightness of the background is approximately 160 (when the LED is displayed with a maximum brightness (255 in 8-bit mode)), whereas in the pulsed mode, the background values are approximately 20, an eightfold difference. To realize the pulsed-operation mode, the camera needs to have configurable trigger output and exposure duration. These are standard features incorporated in almost all industrial-grade cameras. Some camera models even allow the generation of the entire control pulse with the trigger output, thereby reducing the external circuitry to two components (FET and fuse). For illustrative purposes, we can consider how to calculate the correct pulse/exposure duration for a specific camera and LED combination (Pointgrey Firefly MV and Osram SFH4250 LEDs)—for more details of the component characteristics, see http://www.osram-os.com.

If we assume a frame rate of $f = 60$ Hz, then one full pulse/cool-down cycle must have a duration of $D_{\max} = \frac{1}{f} = 16.67$ ms. If we are operating the LEDs at a voltage of 2.4 V (12 V divided by 5 LEDs), then the current is 1 A. We now have to calculate the total cycle duration, based on the duty cycle for each curve and the allowed pulse duration at a current of 1 A. For example, at a duty cycle of 3.3%, the pulse duration is approximately $t_P = 120$ μs for a total cycle duration of $D = 3.6$ ms. At a ratio of 1% with a pulse duration of $t_P = 250$ μs, the total duration rises to $D = 25$ ms $> D_{\max}$, which is too long. We must therefore select a duty cycle of 2%, resulting in a pulse duration of $t_P = 200$ μs with a total duration of $D = 10$ ms, which still offers a comfortable safety margin. Of course, the camera must be able to provide such short exposure times (as is the case for the Pointgrey Firefly MV).

3.1.2. Cameras, Lenses, Filters and Projectors

Cameras. FTIR and DI rely on cameras to detect fingers touching the surface. To create a functional surface, a camera set-up must be found that is capable of sensing light in the near-IR spectrum; this must be coupled with a configuration of special filters that are designed to cut off interference from visible light. Although this can be challenging, the correct choice of camera and filter is essential to gaining the high camera-signal quality required of a responsive multi-touch surface. Camera sensors that are capable of detecting IR light are required; however, the sensitivity of CMOS/CCD image sensors to infrared light varies considerably. When choosing a camera, it is important to find out which sensor is used and determine (from the datasheet) its sensitivity to specific wavelengths of IR light. In many cases illuminators are used that have a wavelength of 880 nm. For low-cost initial prototypes, a USB web camera such as the Philips SPC900N, which uses a Sony CCD image sensor (type: ICX098BQ) is ideal. Web cameras often contain an infrared filter to block ambient infrared light. This filter layer must be removed. In some cases it is detachable, although often it is either glued on to the lens or applied as a coating on the camera sensor itself. The Philips camera, for example, has an infrared blocking filter glued onto the lens; therefore, it is necessary to replace the original lens.

While high-end consumer USB cameras are capable of transmitting images of VGA resolution (640×480 pixels) at high frame rates, they often introduce significant latency. Any latency will reduce the responsiveness of the multi-touch interface; therefore FireWire-based cameras are generally preferred, e.g., the Unibrain Fire-i board color camera. This camera uses the same sensor (Sony ICX098BQ) as the Philips web camera but has a much lower latency. Depending on the size of the display and the projected image, cameras should normally be run at VGA resolution or higher (so as to achieve a reasonable precision), and smooth interaction requires a frame rate that is at least 30 fps. Because the camera only needs to be sensitive to infrared illuminated objects, it is advisable to mount an IR band-pass filter to prevent interference from light in the visible spectrum (for example, from an image projected on your multi-touch surface). For optimal performance this should be a (relatively expensive) band-pass filter that blocks all light other than the IR wavelength of the LEDs you are using; an alternative (cheaper) solution is to use an overexposed developed negative which acts as a (less specific) IR band-pass filter.

Lenses, exposure, and gain. Once a camera has been chosen it must also be correctly configured so as to provide a highly sensitive camera image at a low latency. The *exposure time* controls how long a camera's shutter is held open and thus how much light reaches its sensor. Setting the exposure appropriately is important for high-quality tracking; although a longer exposure time increases the camera's sensitivity, it can negatively impact the camera's frame rate. The camera's *gain* brightens images and increases contrast, but too much gain can lead to unwanted noise in an image. Another important choice is the type of camera lens. Integrating a wide-angle lens in a system allows smaller distances between the camera and the surface. However, lens correction and image rectification waste pixels and so reduce tracking accuracy (especially toward the edge of the camera image).

Projectors. Rear projection is commonly used to display the actual image on the surface; but a number of factors must be considered when deciding upon an appropriate projector for a multi-touch surface. One important factor is the required display resolution. The necessary projection resolution is strongly application dependant, however, a resolution of at least 1024×768 pixels (XGA) is usually sufficient. Additionally, when choosing a type of projector—usually digital light processing (DLP) or liquid crystal display

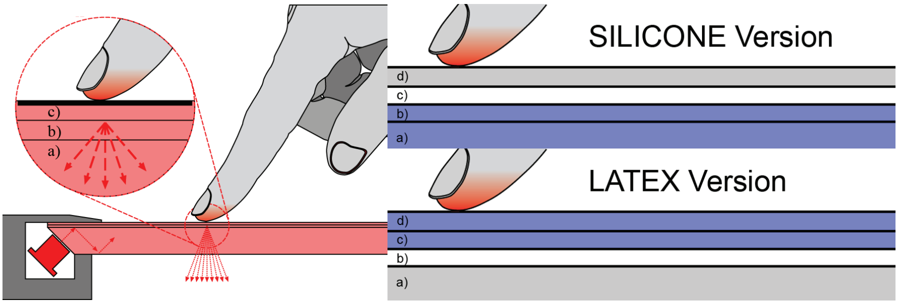

Figure 3. The three layers needed to track finger touches. Left: the polycarbonate plate (a) is covered with a compliant surface layer (b) and a diffuse layer (c). Top right: Silicone compliant layer. The gap (c) is between the projection surface (d) and the combined silicone layer (b) and polycarbonate layer (a); Bottom right: Latex compliant layer. The projection (d) and the latex layer (c) must be combined; the gap (b) is between these two layers and the polycarbonate plate.

(LCD)—it is important to consider both the contrast ratio and the brightness (in lumens). Rear projection generally means that a lower brightness can be tolerated. In most cases, standard office projectors are not appropriate because of their long *throw* (the distance between the projector and projection surface required to produce a clear focussed image). It is possible to use mirrors to reduce this distance, but this usually reduces the quality of the image and significantly complicates the physical design. Where necessary, a front-surface mirror should be used to remove the double projection (ghosting) that can occur due to reflections from the glass front of a conventional mirror. More details can be found in [Muller 08].

3.1.3. Compliant Surfaces and Projection Screens.

The FTIR set-up is comprised of a layer of polycarbonate augmented with a frame of infrared LEDs. When a finger comes in contact with this layer, light from the LEDs, which reflects internally in the polycarbonate, is frustrated and produces a bright intensity region that can be tracked by a camera.

Compliant layer. A plain polycarbonate surface requires the user to apply significant pressure to achieve the frustrated light levels necessary for responsive tracking. The use of a compliant surface can overcome this problem. Applying an additional layer on top of the polycarbonate material can greatly improve the sensitivity of the surface. These compliant surfaces are typically composed of a soft and transparent material. Figure 3 (left) highlights the relevant layers of a commonly used composition. When pressure is applied on the surface, the coupling of the diffuse layer and the polycarbonate surface triggers the FTIR effect. Use of the correct material for a compliant surface is critical as different materials can give rise to two common problems:

1. a strong enough contact is not made with the FTIR layer (see Figure 4 (b));
2. the material sticks to the surface, constantly triggering the FTIR effect even after a finger has been removed (see Figure 4 (d)).

In our experiments the best results for the compliant surface were achieved with SORTA-Clear$^{\text{TM}}$40[1] and ELASTOSIL®RT 60 silicone[2] both materials being relatively hard (hardness shore $A \geq 40$), non-tacky, and very clear. Once hardened, both silicone layers can easily be removed from, and reattached to, the polycarbonate surface. However, using silicone as a compliant surface poses a construction problem since the material comes as a gel, which must be poured evenly over the surface (a relatively difficult and messy task). ELASTOSIL®RT 601 is less viscous and hence easier to pour, resulting in fewer air bubbles in the vulcanized layer. As an alternative to silicone, we found that a thin layer of latex also works well. The latex also has the significant advantage of not having to be poured, reducing the construction time for the combined layer significantly. Furthermore, latex is easier to handle, cheaper to produce, and more readily accessible as an off-the-shelf component.

The order in which the compliant surface is combined with the other projection and polycarbonate layers is important in creating a functional surface; the order varies depending on the material used. Latex must be combined with the projection layer, with an air gap between the latex and the polycarbonate base plate. In the silicone version, we have exactly the opposite requirements. Figure 3 (right) shows the difference between the latex- and silicone-layer construction.

The air gap in both set-ups is a basic requirement for a reliable tracking result. If latex is used, the gap has to be between the latex (that is combined with the projection screen on top) and the underlying acrylic plate. To achieve this gap, it is sufficient to prevent the latex from sticking to the acrylic. Latex becomes adhesive when it is wet, which can be used to glue it to the projection layer. To create an air gap, however, the latex has to be perfectly dry. Depending on the type of latex, a partial sticky effect might remain. In this case, applying a thin layer of powder can help, for example. For the silicone version, the air gap needs to be created between the projection and the silicone layer. During the production process, the silicone is poured onto the acrylic surface, which creates a combined layer construction without any gap in between. To create the gap between the silicone and the projection

[1] http://tb.smodev.com/tb/uploads/SORTA_CLEAR_40_32707.pdf

[2] http://www.wacker.com/internet/webcache/en_US/PTM/TM/Elastosil/Elastosil_RT_Addition/ELASTOSIL_RT_601.pdf

layer, adhesiveness between these two layers has to be avoided. Projection screens like Rosco or matte tracing paper are suitable, whereas glossy backside materials such as backlit films would lead to unwanted sticking effects.

Projection layer. As mentioned in the previous section, the configuration of surface layers varies with the choice of compliant surface material. Depending on whether silicone or latex is used, a different projection screen must be chosen. The main requirements to base this choice upon are that an air gap should be achievable between the two layers, and that when the screen chosen is pressed against the compliant surface, the FTIR effect is triggered. Not all materials meet these requirements. Figure 4 shows different results for projection materials on top of silicone.

Figure 4 (a) shows an optimal result for FTIR with a high contrast touch point. Materials that resulted in touch points that are too dark (b) or those that showed permanent traces on the silicone as well as materials that completely stuck to the silicone (c) are not suitable for FTIR. Rigid PVC and tracing paper appear to be a good solution in combination with silicone. They do not stick to the silicone but trigger the FTIR effect quite well.

When using latex, we found HP Colorlucent Backlit UV (a material originally designed for use in backlit signs) to be a good choice. Similar to

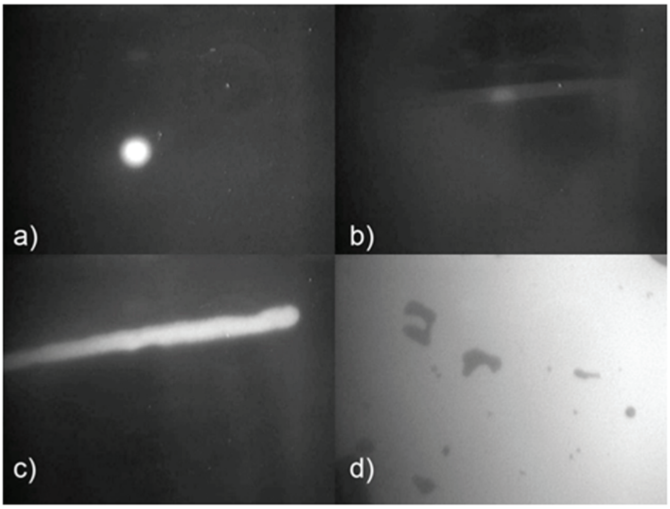

Figure 4. (a) Rigid PVC (backlit); (b) Rosco translucent [4]; (c) Sihl polyester film 100 at; (d) HP backlit UV. Using the wrong combination of materials can results in two main problems: either the FTIR effect is not strong enough (b) or the layers stick together (d).

rear-projection screens, it yields a diffuse image without any hotspots from the projector, making it a good rear-projection surface. Because of its glossy backside, it cannot be used with the silicone, as it adheres to the silicone as shown in Figure 4 (d). Rosco screens can also be combined with latex.

3.2. LCD Enabled MT Surfaces with Optical-Based Sensors

The use of an LCD monitor to display an image on a surface affords several key advantages over projector-based systems for the development and deployment of multi-touch interfaces. Generally, LCD monitors provide a higher display resolution than projectors (often for a lower price). For instance, a screen with full 1080-pixel HD resolution will cost several thousands of dollars less than a projector with a similar pixel output. Additionally, the slim profile of LCD monitors makes them easy to house; this is especially important for those wishing to embed a multi-touch surface into the structure of a tabletop. Lastly, unlike projectors, LCD screens do not have issues with key-stoning and throw-distance. These unique properties of LCDs make them a compelling option for multi-touch interfaces, especially for applications that demand a high degree of visual fidelity and resolution.

There are however challenges that need to be overcome for the multi-touch developer wishing to utilize an LCD screen as a display technology. It is imperative to have knowledge of how LCD technology works and be familiar with their manufacturing and assembly. The first issue that must be understood is that each pixel of an LCD monitor is comprised of three electronically controlled filters (red, green, and blue) which modulate over a backlight to emit a desired color. Essentially, the LCD glass panel is transparent when no current is running through the screen. Crisscrossed polarizing filters are on the front and back side of the glass panel. The polarizing filters give an LCD its black appearance since their opposing orientation blocks visible light. However, polarizing filters do not polarize light within the IR spectrum. So while a LCD panel looks opaque to our eyes, IR light can be transmitted through the screen unperturbed. This concept is crucial for the use of LCD screens in optical multi-touch systems.

The next part of the LCD assembly is the backlight and filter-chain. A backlight is necessary in order to illuminate the LCD pixels. The backlight for monitors that are less than 23 inches in size consist of a long thin fluorescent light bulb, which lines the length of the monitor. Attached to the bulb is an acrylic sheet (called the light guide), which has a honeycomb pattern of white dots. Based on the principle of total-internal reflection, the light from the fluorescent tube travels inside the acrylic sheet until it reflects off one of these white dots. This method for backlighting allows for thin displays. For LCD monitors that are 27 inches and larger, the acrylic-guide method is replaced by a rail of lights that are placed behind the screen to provide the backlighting. However, because of the polarizing filters and the method in which the crystals distort and filter light, having only a backlight is somewhat ineffective for illuminating the display. This can be understood if one imagines adjusting a laptop screen in order to achieve the best viewing angle, which is orthogonal to line of sight; if you tilt the laptop screen too much, the display image loses much of its color and appearance. To improve the lighting conditions of the display, LCD manufacturers include a layering of several different filters, which modulate and affect the backlight in various ways. The most common filters include the following.

1. *Diffuser.* This filter diffuses the backlight to disperse in every direction.
2. *Brightness enhancement film (BEF).* This filter can magnify light with a shorter focal length in different directions. It is used to disperse light 180 degrees.
3. *White reflector.* This is an opaque white filter, which reflects any light that may have escaped the filters.

These are the three basic filters for LCD monitors; however some manufacturers may use additional filters to improve their product quality; for instance, using different types of diffusers, or more than one BEF to improve the viewing angle. Of these filters, the only one that impedes IR light, and is therefore of concern when developing optical multi-touch surfaces, is the last white opaque filter. This filter is totally white and hence needs to be removed; the rest can and should remain to keep optimal viewing performance.

There are two broad methods, which so far have been successful for creating interactive LCD surfaces with optical sensing. The first, and easiest, is the side-illuminated method where IR LEDs are installed around the bezel of the LCD. The LEDs shine IR light across the top surface of the screen; when a finger touches the LCD screen, light reflects off the finger and traverses through the monitor, which is then captured by an IR-sensitive camera. The illumination hardware required for this approach is very similar to the FTIR method. Therefore, it is often possible to install an FTIR panel on top of an LCD screen and then remove the acrylic, keeping the LEDs intact. With this method, it is recommended to identify IR LEDs with a small package (3 mm or SMD) and with a small viewing angle (typical angles for LEDs are 30 degrees, but shorter angles are available at around 15–18 degrees). Choosing a smaller angle will focus more of the light to cross the screen. Finally, emerging IR laser LEDs promise to be the ideal choice for this method, because these light sources ensure that the IR light beam is small and hence focused as a blanket over the surface [Motamedi 08].

The second method for enabling multi-touch with LCD screens is to create a matrix of IR transceivers behind the LCD panel as described in [Hodges et al. 07]. Each transceiver consists of an IR emitter and an IR detector. The emitter pulses IR light at a certain frequency, which the sensor can detect (similar in theory to IR remote controls except here the light is not encoded to pulse information). When a finger or an object touches the screen, the finger reflects back the light, which is detected by the sensor. By creating a matrix that consists of many of these transceivers, it is possible to cover the entire surface area of the LCD screen. The number of transceivers and their size and pitch (distance between sensors) determines the accuracy and resolution of the touch surface. This approach allows for a thin form-factor display as the sensors can be placed directly against the surface, in contrast to the cameras used in other approaches. However this approach is not simple to construct and requires expert knowledge of electronics, circuit design, and digital-signal processing (DSP). Additionally, this approach is not scalable, as larger surfaces require more sensors, which increases cost and latency.

3.3. Software

Once the hardware is in place the next major challenge, which must be faced, is the software processing of the camera image to interpret the users interactions. To achieve this, a pipeline of image processing operators that transform a camera image into user interface events must be set up.

3.3.1. FTIR Tracking Pipeline

Figure 5 shows the typical imaging pipeline of an FTIR set-up. In an initial step, images captured by a camera are preprocessed. Preprocessing consists of first rectifying the camera image so that the image pixels and display pixels match up. Possible camera models for this include those described in [Tsai 87] and [Ojanen 99], with Tsai's model generally accepted as a standard. This transformation can also be done during postprocessing, where only the actual blob coordinates have to be calculated. Done at the beginning, it has the advantage that intermediate images can be displayed on the surface without distortion.

Following the rectification, history subtraction is used to remove any unchanging parts. If the camera image is noisy, an appropriate noise-reduction filter (opening/closing, lowpass, or similar) can be added to the pipeline.

Simple threshold-based segmentation using a connected-components algorithm (described for example in [Han and Wagner 90]) finds bright regions—so-called "blobs"—in the preprocessed image. These are the areas where something is touching the surface. Principal components analysis (PCA) can be used to calculate statistical data (size, eccentricity, etc.) for the blobs. Using this data, it is possible to distinguish touches with fingers from other objects and from noise.

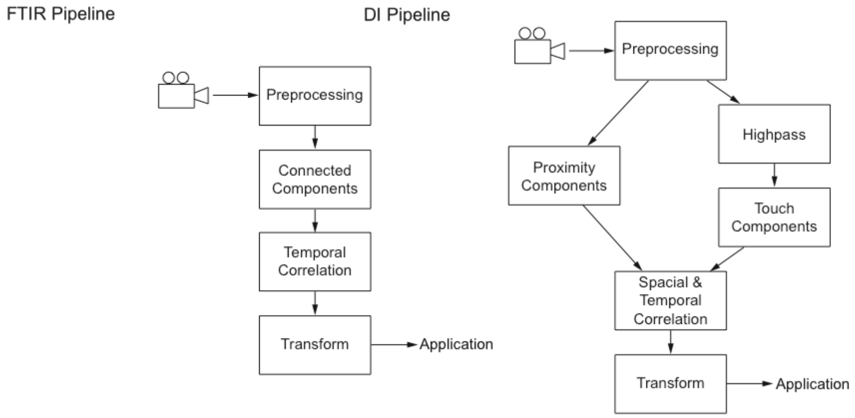

Figure 5. FTIR tracking pipeline (left). DI tracking pipeline (right).

Post-processing involves finding corresponding touches in different camera frames (temporal correlation). Finding an algorithm that consistently detects the movements of touches from one frame to another turns out to be surprisingly hard. A simple greedy algorithm that goes through all new blobs and picks the closest old blob for each one is less than optimal. Blobs that split and merge confuse it. Also, the greedy algorithm often incorrectly exchanges blobs when many fingers are quickly moving over the surface.

There are several improvements that can be made over the naive algorithm:

1. A robust solution first calculates all distances between blob pairs and sorts the results by distance. In a second step, the blob pairs with the closest distances are correlated. The second step is repeated until all new blobs have been accounted for.

2. Dead reckoning can be used to extrapolate the position of the old blob using its previous speed before calculating the distance.

3. The distance function need not be the Euclidian distance between the blob centers. Statistical data from the PCA performed earlier (e.g., size and eccentricity of the blobs) can be factored into the distance function.

Together, these three changes result in temporal correlation that is robust. The first change mainly helps in the face of splitting and merging blobs. Using dead reckoning, almost all quickly moving fingers are resolved correctly. A more complicated distance function can be used to improve correlation further if needed.

3.3.2. DI Tracking Pipeline

DI tracking is a more complex process but allows for proximity as well as touch to be sensed. DI touch detection exploits the fact that objects at a distance from the surface are appear blurred. The algorithm reacTable [Kaltenbrunner and Bencina 07] does this by adaptive thresholding based on the curvature of the luminance surface (see [Costanza et al. 03] for a detailed description of the algorithm). The multimedia platform *libavg* [3] used in the *c-base MTC* pioneered the use of a high-pass filter to achieve the same effect. Note that a full high-pass filter is computationally expensive, so *libavg* uses fragment shaders to implement the filter.

As can be seen in Figure 5, the image pipeline is split and the end the connected components algorithm is run twice, once each for touch and once for proximity sensing. Touch-sensing involves an additional high-pass filter to isolate areas very close to the surface. After the regions have been found, touch and proximity information can be correlated. With appropriate thresholds, hand (proximity) blobs reliably enclose the finger (touch) blobs, so a correlation is easy to establish. Additionally, the vector from the hand center to the finger center is a very good approximation of the direction in which the finger is pointing.

3.3.3. Interface Considerations

The tracking pipeline provides higher-level software layers with information about finger and hand positions. TUIO [Kaltenbrunner et al. 05] uses open sound control over UDP to transmit this information in a format that can be interpreted easily by a wide variety of tools and languages. By default Touchlib and many other libraries come with a wrapper that sends TUIO events over the commonly used OpenSound Control[4] protocol. For many modern programming languages such as C#, Adobe Flash (Actionscript 3), Java, Max/DSP, Processing, Pure Data, Python, and Visual Basic, OSC libraries are available. When using Flash, it is required to convert UDP packages to TCP. This can be done by using the tool Flosc which acts as a proxy. Work is in progress to provide higher-level interfaces (*libavg*, *libtisch* [Echtler 08]); *libavg* includes event processing that correlates touches to a hierarchy of on-screen widgets[5]. This corresponds to the mouse event handling that window systems provide and hence affords the basis for robust implementation of classical GUI widgets like buttons and scroll bars. Both libraries support emerging gesture standards that allow for dragging, rotating, and scaling of GUI elements. When an application uses the OSC protocol, it is only able to

[3] http://www.libavg.de/

[4] OSC http://www.cnmat.berkeley.edu/OpenSoundControl/

[5] https://www.libavg.de/wiki/index.php/Event_Handling

receive events containing properties of the detected blobs. It is not possible to adjust the settings of Touchlib from the application. However, since OSC uses the UDP network protocol to transfer data, it makes it possible to create a set-up in which a dedicated system provides blob tracking and transfers the data to another system that provides the visualization.

At higher levels, window-system-like event processing, classical GUI widgets (buttons, etc.), and emerging gesture standards (dragging, rotating, and scaling elements, for instance) are supported by some libraries. The most commonly used tracking libraries that are currently available are Touchlib, tbeta, libavg, multi-touch lib T-Labs, OpenFTIR, VVVV, and OpenTouch. For a more detailed overview please refer to [Schöning et al. 08a].

4. Examples

To conclude we present an actual example of how to construct an interactive multi-touch table (see Figure 6). The cost of this table including all materials (projector, camera, LEDS, and other materials) is around $3000. Step-by-step instructions are provided by Schmidt in [Schmidt 09]. Various other useful websites and communities with additional information that will help you build your own multi-touch surface can be found online. Some specific examples are the following: FTIR Multitouch and Display Device (http://www.lowres.ch/ftir), David Smith and David Holman's guide, "Building a Multi-Touch Sensitive Table"[6], the NUI group forum (http://nuigroup.com/forums/), and on Harry van der Veens (now commercial) website[7] with working Flash applications for multi-touch applications (http://www.multitouch.nl), There are of course a great number of commercial applications on the market. The work [Han 05] of Jeff Han has had a great impact on the community. More information about his multi-touch project (http://cs.nyu.edu/~jhan/ftirtouch) and his company can be found online (http://www.perceptivepixel.com).

The Microsoft Surface is a multi-touch interactive table which is one of the most prominent examples. It is developed as an integrated hardware and software system that allows a user, or multiple users, to manipulate digital content with natural motions, hand gestures, or physical objects. The surface employs a slightly modified DI set-up that is explained on the (non-technical) website[8]. With Microsoft Touch Wall[9] and the Microsoft Touch Sphere,[10]

[6] http://dundee.cs.queensu.ca/wiki/index.php/Building_a_Multi-Touch_Sensitive_Table

[7] http://www.multitouch.nl/documents/multitouchdisplay_howto_070523_v02.pdf

[8] http://arstechnica.com/news.ars/post/20070530-what-lurks-below-microsofts-surface-a-qa-with-microsoft.html

[9] http://www.microsoft.com/presspass/events/ceosummit/default.mspx

[10] http://research.microsoft.com/~benko/projects/sphere

Microsoft recently presented two new interactive surfaces. Other realted previous work are Microsoft's TouchLight [Wilson 04] and LucidTouch [Wigdor et al. 07].

The NUI group (Natural User Interface group) is one of the most active groups (with about 4000 members) dealing with multi-touch interaction and interactive media. Their goal is to create open-source machine-sensing techniques, which will benefit artistic and educational applications. Their focus is on the creation of an "Open Source Interface." On their webpage http://nuigroup.com, they provide a wiki and a active forum. TouchLib (see Section 3.3) and its successor tbeta (http://tbeta.nuigroup.com/) are their main projects.

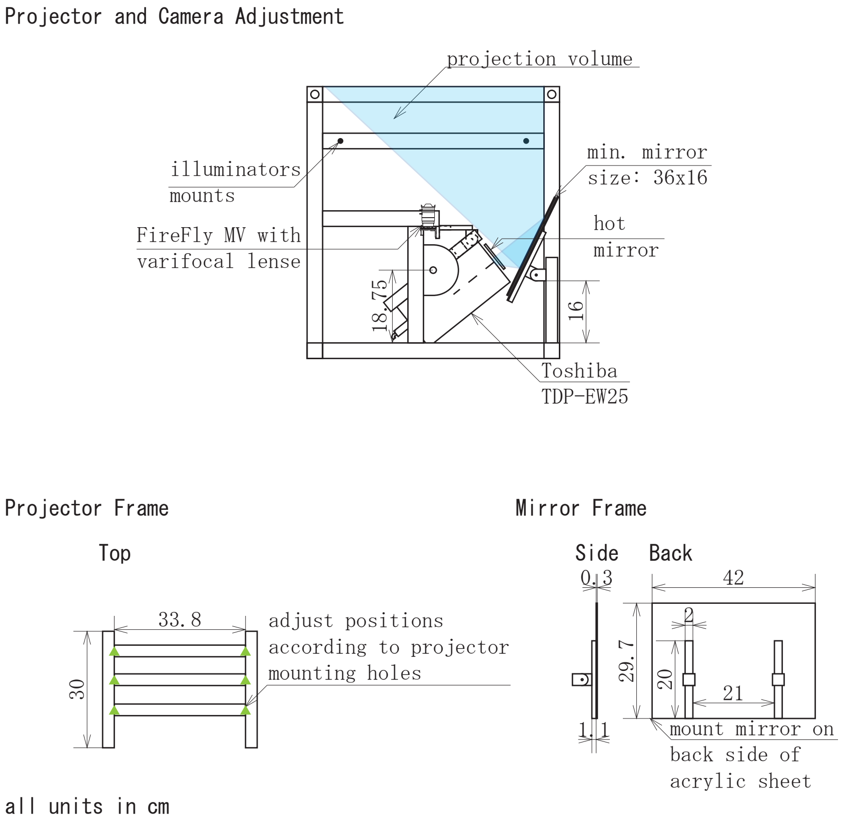

Figure 6. Detailed engineering drawing for an interactive multi-touch table.

5. Looking Forward

Despite the innovations described in this paper many open questions remain for researchers: What are the benefits of multi-touch systems over single-touch systems? What can graphics and interaction design practitioners "do" with multi-touch surfaces? For which applications is multi-touch input appropriate, viable, and useful? Are there more interaction possibilities than "just" rotating and scaling photos or zooming into maps? We hope that our description of the realities of building optical multi-touch surfaces will enable you to join us in answering these questions and in building more and more examples of interactive multi-touch surfaces of various kinds.

Finally we would like to encourage developers to consider the following guidelines, which we think are important in designing good applications for such interactive surfaces.

- Let non-experts explore your systems (like in the City wall project, which provided a lot of interesting feedback).
- Design interfaces that help users forget WIMP (window, icon, mouse and pointer).
- Design systems that can only be used by performing multi-touch gestures to investigate the advantages over single-touch systems.
- Do fewer lab studies and give the technology to users and test it in the wild.

As Bill Buxton said: "Remember that it took 30 years between when the mouse was invented by Engelbart and English in 1965 to when it became ubiquitous." We want to emphasize this and let multi-touch become a genuine useful technology that successfully passes through the inevitable hype.

References

[Buxton 08] Bill Buxton. "Multi-Touch Systems that I Have Known and Loved." Available online (http://www.billbuxton.com/multitouchOverview.html) 2008.

[Costanza and Robinson 03] E. Costanza and J. Robinson. "A Region Adjacency Tree Approach to the Detection and Design of Fiducials." In *Proceedings of Vision, Video and Graphics (VVG*, pp. 63–70. Aire-la-Ville, Switzerland: Eurographics Assocation, 2003.

[Costanza et al. 03] E. Costanza, SB Shelley, and J. Robinson. "Introducing Audio d-touch: A Tangible User Interface for Music Composition and Performance." *Proc. of the 6th Int'l Conf. on Digital Audio Effects (DAFX,* pp. 63–70. Available online (http://www.elec.qmul.ac.uk/dafx03/proceedings/pdfs/dafx27.pdf), 2003.

[Dietz and Leigh 01] P. Dietz and D. Leigh. “DiamondTouch: A Multi-User Touch Technology.” *Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology*, pp. 219–226. New York: ACM Press, 2001.

[Echtler 08] Florian Echtler. “TISCH: Tangible Interactive Surfaces for Collaboration between Humans.” Available online (http://tisch.sourceforge.net/), 2008.

[Han and Wagner 90] Y. Han and R.A. Wagner. “An Efficient and Fast Parallel-Connected Component Algorithm.” *Journal of the ACM (JACM)* 37:3 (1990), 626–642.

[Han 05] Jefferson Y. Han. “Low-Cost Multi-Touch Sensing through Frustrated Total Internal Reflection.” In *UIST ’05: Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology*, pp. 115–118. New York: ACM Press, 2005.

[Hodges et al. 07] Steve Hodges, Shahram Izadi, Alex Butler, Alban Rrustemi, and Bill Buxton. “ThinSight: Versatile Multi-Touch Sensing for Thin Form-Factor Displays.” In *UIST ’07: Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology*, pp. 259–268. New York: ACM Press, 2007.

[Johnson 72] Ralph G. Johnson. “US Patent #3,673,327: Touch Actuable Data Input Panel Assembly.” Available online (http://www.google.com/patents?vid=USPAT3673327), 1972.

[Jung et al. 08] Y. Jung, J. Keil, J. Behr, S. Webel, M. Zöllner, T. Engelke, H. Wuest, and M. Becker. “Adapting X3D for Multi-Touch Environments.” In *Web3D ’08: Proceedings of the 13th International Symposium on 3D Web Technology*, pp. 27–30. New York: ACM Press, 2008.

[Kaltenbrunner and Bencina 07] M. Kaltenbrunner and R. Bencina. “reacTIVision: A Computer-Vision Framework for Table-Based Tangible Interaction.” In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction*, pp. 69–74. New York: ACM Press, 2007.

[Kaltenbrunner et al. 05] M. Kaltenbrunner, T. Bovermann, R. Bencina, and E. Costanza. “TUIO: A Protocol for Table-Top Tangible User Interfaces.” *Proc. of the The 6th Int’l Workshop on Gesture in Human-Computer Interaction and Simulation*. New York: ACM Press, 2005.

[Kasday 84] Leonard R. Kasday. “US Patent #4,484,179: Touch Position Sensitive Surface.” Available online (http://www.google.com/patents?vid=USPAT4484179), 1984.

[Mallos 82] James B. Mallos. “US Patent #4,346,376: Touch Position Sensitive Surface.” Available online (http://www.google.com/patents?vid=USPAT4346376), 1982.

[Matsushita and Rekimoto 97] N. Matsushita and J. Rekimoto. “HoloWall: Designing a Finger, Hand, Body, and Object Sensitive Wall.” In *Proceedings of the 10th Annual ACM Symposium on User Interface Software and Technology*, pp. 209–210. New York: ACM Press, 1997.

[Moscovich and Hughes 08] Tomer Moscovich and John F. Hughes. "Indirect Mappings of Multi-Touch Input Using One and Two Hands." In *CHI '08: Proceeding of the Twenty-Sixth Annual SIGCHI Conference on Human Factors in Computing systems*, pp. 1275–1284. New York: ACM Press, 2008.

[Moscovich 06] Tomer Moscovich. "Multi-Touch Interaction." In *CHI '06: CHI '06 Extended Abstracts on Human Factors in Computing Systems*, pp. 1775–1778. New York: ACM Press, 2006.

[Motamedi 08] Nima Motamedi. "HD Touch: Multi-Touch and Object Sensing on a High Definition LCD TV." In *CHI '08: CHI '08 Extended Abstracts on Human Factors in Computing Systems*, pp. 3069–3074. New York: ACM Press, 2008.

[Mueller 74] Robert E. Mueller. "US Patent #3,846,826: Direct Television Drawing and Image Manipulation System." Available online (http://www.google.com/patents?vid=USPAT3846826), 1974.

[Muller 08] Laurence Y. L. Muller. "Multi-Touch Displays: Design, Applications and Performance evaluation." Master's thesis, University of Amsterdam, Amsterdam, The Netherlands, 2008.

[Ojanen 99] H. Ojanen. "Automatic Correction of Lens Distortion by Using Digital Image Processing." Technical Report, Rutgers University, Dept. of Mathematics, 1999.

[Rekimoto 02] Jun Rekimoto. "SmartSkin: An Infrastructure for Freehand Manipulation on Interactive Surfaces." In *CHI '02: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 113–120. New York: ACM Press, 2002.

[Schmidt 09] D. Schmidt. "Design and Realization of an Interactive Multi-Touch Table." Technical report, Lancaster University, 2009.

[Schöning et al. 08a] Johannes Schöning, Peter Brandl, Florian Daiber, Florian Echtler, Otmar Hilliges, Jonathan Hook, Markus Löchtefeld, Nima Motamedi, Laurence Muller, Patrick Olivier, Tim Roth, and Ulrich von Zadow. "Multi-Touch Surfaces: A Technical Guide." Technical report, Technical University of Munich, 2008.

[Schöning et al. 08b] Johannes Schöning, Brent Hecht, Martin Raubal, Antonio Krüger, Meredith Marsh, and Michael Rohs. "Improving Interaction with Virtual Globes through Spatial Thinking: Helping Users Ask "Why?"." In *IUI '08: Proceedings of the 13th International Conference on Intelligent User Interfaces*, pp. 129–138. New York: ACM Press, 2008.

[Schöning et al. 09] Johannes Schöning, Antonio Krüger, and Patrick Olivier. "Multi-Touch is Dead, Long Live Multi-Touch." In *CHI 2009: Workshop on Multi-Touch and Surface Computing.*pp. 1–4. New York: ACM Press, 2009.

[Tsai 87] R. Tsai. "A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-The-Shelf TV Cameras and Lenses." *IEEE Journal of Robotics and Automation* 3:4 (1987), 323–344.

[Valli and Linari 08] Alessandro Valli and Lorenzo Linari. "Natural Interaction Sensitivetable." In *CHI '08: CHI '08 Extended Abstracts on Human Factors in Computing Systems*, pp. 2315–2318. New York: ACM Press, 2008.

[White 87] Richard M. White. “US Patent #4,484,179: Tactile Sensor Employing a Light Conducting Element and a Resiliently Ddeformable Sheet.” Available online (http://www.google.com/patents?vid=USPAT4484179), 1987.

[Wigdor et al. 07] Daniel Wigdor, Clifton Forlines, Patrick Baudisch, John Barnwell, and Chia Shen. “Lucid Touch: A See-Through Mobile Device.” In *UIST ’07: Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology*, pp. 269–278. New York: ACM Press, 2007.

[Wilson 04] Andrew D. Wilson. “TouchLight: An Imaging Touch Screen and Display for Gesture-Based Interaction.” In *ICMI ’04: Proceedings of the 6th International Conference on Multimodal Interfaces*, pp. 69–76. New York: ACM Press, 2004.

Web Information:

http://jgt.akpeters.com/papers/SchoningEtAl09

More information on the actual development on interactive multi-touch surfaces can be found at http://ifgi.uni-muenster.de/multi-touch-bootcamp.

Johannes Schöning, DFKI (German Research Center for Artificial Intelligence) GmbH,
Campus D3 2, Stuhlsatzenhausweg 3, D-66123 Saarbrücken, Germany
(johannes.schoening@dfki.de)

Received February 15, 2009; accepted in revised form December 17, 2009.