

SideSight: Multi-“touch” interaction around small devices

Alex Butler, Shahram Izadi, Steve Hodges
Microsoft Research
7 J J Thomson Avenue
Cambridge, UK
{ dab, shahrami, shodges }@microsoft.com



Figure 1: SideSight supports virtual multi-“touch” interactions around the body of a small mobile device. Optical sensors positioned along each edge of the device allow fingers to be sensed as they approach the device from the sides. In the example depicted, the device is resting on a flat surface, and fingers touching the surface on either side of the device are sensed to provide multi-touch input. As the left-hand finger is moved down and the right-hand one moved up (looking through the sequences of images from left to right), the photo on the display is rotated anticlockwise.

ABSTRACT

Interacting with mobile devices using touch can lead to fingers occluding valuable screen real estate. For the smallest devices, the idea of using a touch-enabled display is almost wholly impractical. In this paper we investigate sensing user touch *around* small screens like these. We describe a prototype device with infra-red (IR) proximity sensors embedded along each side and capable of detecting the presence and position of fingers in the adjacent regions. When this device is rested on a flat surface, such as a table or desk, the user can carry out single and multi-touch gestures using the space around the device. This gives a larger input space than would otherwise be possible which may be used in conjunction with or instead of on-display touch input. Following a detailed description of our prototype, we discuss some of the interactions it affords.

ACM Classification: H5.2 [Information interfaces and presentation]: User Interfaces. - Graphical user interfaces.

General terms: Design, Human Factors, Algorithms

Keywords: Proximity sensing, Multi-touch, Mobile device interaction, Novel hardware

INTRODUCTION

Touch can be an intuitive and effective mode of input for mobile devices when compared to alternatives such as

physical buttons, keypads, jog dials and joysticks. Consumer products by Apple, LG, HTC and others often predominantly use a touch-sensitive screen for input, minimizing the need for other physical input mechanisms, hence maximizing the display size. Some of these products also extend the touchscreen sensing capabilities to support multi-touch gestures, giving rise to even richer user interactions.

Despite the flexibility of touchscreens, using such an input mode carries a number of tradeoffs. For many mobile devices, e.g. wristwatches and music players, a touchscreen can be impractical because there simply isn't enough screen real estate. With a continued trend for ever-smaller devices, this problem is being exacerbated. Even when a touchscreen is practical, interacting fingers will occlude parts of the display, covering up valuable screen pixels and making it harder to see the results of an interface action.

SideSight is a new technique which supports multi-touch interaction on small mobile devices such as cellphones, media players, and PDAs. However, the ‘touch sensitive’ regions used for these interactions are not physically on the device itself, but are instead formed around the periphery of the device. Proximity sensors embedded along the edges of the device can detect the presence and position of fingers adjacent to them. When using SideSight to interact with a device, it is typically first placed onto a flat surface such as a table or desk. The space immediately around the device acts like a large, virtual, multi-point track pad. Thus the display itself is not occluded as the user interacts. Figure 1 shows an example two fingered rotate gesture on SideSight.

This paper describes the concept, implementation and potential application of SideSight. We start with a brief overview of related work, then we describe the implementation

of our SideSight prototype in some detail. Following this, we present three different interaction scenarios which we have implemented using our first prototype hardware, namely multi-touch object manipulation, menu selection, and the ability to use a stylus directly on the display and touch around the device simultaneously. We end with a brief discussion of future improvements we have in mind for SideSight.

RELATED WORK

As mentioned, one of the main issues with using touch for a mobile device is that of occlusion of the display. With multi-touch displays, these problems are naturally increased. A stylus greatly reduces occlusion but can be inconvenient. Not only does it require the use of an additional object (the stylus), but there is a loss of direct contact and multi-touch input is impractical.

Techniques such as LucidTouch [10] and Shift [9] offer novel solutions for a variety of scenarios and minimize occlusion. LucidTouch supports a rich repertoire of interaction possibilities for in-hand use, but is less suited to scenarios in which the rear of the display is not accessible – for example if the device is resting on a table. Shift permits occluded pixels to be seen by redrawing them next to the fingertip. However it is unclear how Shift would scale to multi-touch input or to smaller displays, where the size of the occluding fingertips is relatively large compared to the display area left visible.

SideSight mitigates the need for user input on the precious screen real estate, or in fact on any part of the device itself, instead using proximity sensing to divert the user input region to the areas on either side of the device. Proximity sensing for user-interfaces has been explored on mobile devices in past work. Hinckley demonstrates the use of a single proximity sensor mounted on the front of the device to support coarse interaction [2]. The virtual laser-projection keyboard [8] uses an infrared camera to detect fingertip interactions more accurately, but requires a suitable optical projection angle which does not necessarily suit very thin form factor devices. Smith et al. [7] describe using low frequency electric fields for both touch and proximity-based user interactions, although not for mobile scenarios. This approach can be susceptible to noise and unsuitable for accurate multiple finger detection. Given these issues we have opted for an approach based around optical sensing for SideSight, although other sensing techniques based on capacitance or ultrasonics may also be feasible.

IR based interaction has previously tended to focus on on-screen interactions through versions of the Carroll Touch [1] multiple IR beam break technique or more modern incarnations such as used in Entertaible [4] and mobile-device friendly RPO touch screen technology [6]. LightGlove [5] is a wearable input device using IR reflections to sense finger orientation from a wrist mounted module - to provide simplified data-glove type user input.

SideSight should be viewed as a hardware add-on which allows small, potentially thin form-factor devices to use the space beyond the periphery of the device for touch input.

The proximity sensors can cover a relatively large area compared to the size of the device and its display, and are capable detecting multiple fingers simultaneously. SideSight can be employed either independently, or as an additional interactive technique to enhance existing multi-touch displays. In this way, input clatching and mode switching can be reduced. For interactions beyond the display SideSight can provide for finer angular resolutions for rotation type gestures due to the greater separation of fingers.

THE SIDESIGHT PROTOTYPE

Our SideSight prototype is based around an HTC Touch mobile phone which we have augmented with two linear arrays of discrete infrared (IR) proximity sensors. In our first prototype we use one array of sensors ‘looking out’ from each long side of the device – although the entire perimeter could be made active.

Figure 2 shows the basic principle behind SideSight. It shows the left hand side of a device augmented with a sensor board with an array of IR LEDs each emitting infrared light outwards from the device. When an IR reflective object such as a fingertip is placed in range of this signal, some of the IR light is reflected back from the object towards the device. This light is detected by IR photodiodes that are paired with each LED.

Sensor Boards

Each sensor board consists of ten Avago HSDL 9100-021 940nm IR proximity sensors spaced 10mm apart. These are low cost integrated units which contain an IR LED and matching IR photodiode in a compact package. The IR sensor is capable of ranging from a few mm to somewhere between 5 and 10cm in typical use, by detecting the amount of reflected IR. This is measured using the same charge-discharge IR sensing circuitry as used in [3]. By aggregating data from the array of proximity sensors, the output from each SideSight sensor panel forms a simple 10x1 pixel “depth” image. Figure 3 shows the SideSight proximity sensing PCB.

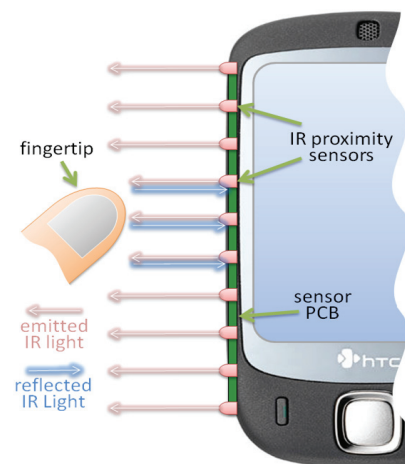


Figure 2: The basic principle behind SideSight. IR is shone outwards from the device via a series of IR LEDs; reflections from nearby objects are sensed using a corresponding array of IR photodiodes.

Through experimentation, we found a slight inward bias in the IR sensor emitter and detector directions making it important to orientate the device to avoid emitted IR light being directed into the tabletop surface (which can saturate the photodiode on matte tabletop surfaces). Mounting the sensor LEDs at the bottom - to emit IR light slightly up and away from the tabletop surface helped avoid these issues.



Figure 3: The SideSight sensor PCB. The ten IR proximity sensors are clearly visible. Note: photodiodes are on the top, emitters directly underneath.

For expedience we built our prototype sensor PCBs with a USB interface. Since the HTC touch device cannot act as a USB host, we actually interface the sensor PCBs with a PC which then relays the raw sensor data stream directly to the HTC Touch using Bluetooth. Our prototype runs at around 11 frames per second which is a limitation of the test rig rather than the sensors themselves which are capable of significantly faster sensing with alternative circuits. Sensor processing is performed in the phone itself. Our aim in the future is to move to a faster self-contained solution without the need for the PC proxy.

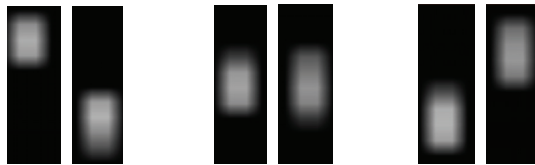


Figure 4: Examples of left- and right-hand greyscale depth maps from the 10x1 pixel SideSight sensor PCB. These correspond to the 3 photos in Figure 1.

Sensor Processing

Each sensor board currently provides a 10x1 depth image which shows any IR reflective objects in range of each side. This allows us to determine where an object is in relation to the device along both X and Y axes.

Figure 4 shows some typical left- and right-hand sensor data visualized as paired greyscale 10x1 pixel depth maps. The three pairs of images in the figure correspond to the images in Figure 1. Fingers appear as grey ‘blobs’ in the image, and get brighter the closer they are sensed to the device. We detect the location of a finger on the X-axis using the brightest pixel intensity associated with each ‘blob’ in the sensor image. Pixel intensity is only an approximation of distance and dependent on the object’s reflective properties – although we have found fairly consistent behaviour in practice when tracking fingers. In practice, we found the limit of our depth of sensing to be approximately 8cm from the phone.

To detect the location of the finger along the Y-axis, we find the centre of mass of ‘blobs’ in the sensor image – by first thresholding the image, and then carrying out connected component analysis. In practice, we found that often when people are interacting with a single finger, other fingers can accidentally become visible in the field of the view of the sensors, shifting the centre of mass and thus trigger-

ing undesirable input. To avoid this, we take the topmost extent of the ‘blob’ furthest from the user in the sensor image as the active contact point. Although this limits us to tracking one contact on each side of the device, it still allowed users to carry out common multi-touch interactions using both hands. We found this approach to work well in practice, although we are still experimenting with other approaches that enable multiple fingers to be used on each side of the device. A ‘blob’ that rapidly appears and then disappears in the sensor image can also be used to generate a “mouse click” type selection event.

We were pleasantly surprised by the performance of the SideSight sensors in the typical office environments we tried given that we took no special precautions to reject ambient light. We attribute this in part to the fact that the sensors are looking horizontally rather than vertically upwards towards overhead lighting. We believe that improved periodic background subtraction and use of modulated IR techniques will substantially improve robustness in brighter environments and are actively experimenting with this.

INTERACTING WITH SIDESIGHT

In our first experiments with SideSight, only simple single-finger interactions were supported. In this mode the user can translate on-screen objects in both X and Y using only one finger on one side of device. Input clutching is feasible supporting panning of larger on-screen objects. We quickly found that a scale factor was useful for mapping the motion of the finger to the motion of objects.

Following our initial single-touch experiments, we quickly moved to a dual-touch approach, using simultaneous input from both sides of the display to support typical zoom, pan and object rotate gestures. We found that these gestures – around the body of the device – feel quite natural, even though they break the direct manipulation metaphor. They work particularly well when the virtual object is larger than the screen size, anecdotally giving the user the sense that they are actually touching the object.

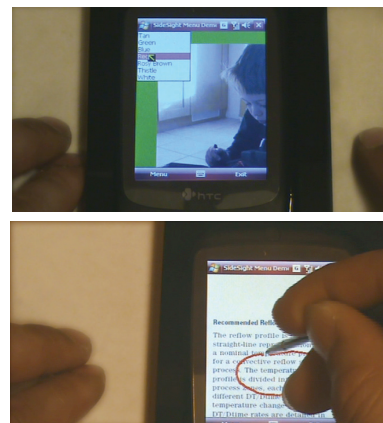


Figure 5: Top: “conventional” menu control using SideSight to control a mouse cursor and make selections by ‘tapping’. Bottom: multi-modal pen and gesture interactions are also supported by SideSight. Here the user marks up a document directly on the screen with a stylus and simultaneously uses the non-dominant hand to pan the document.

We have also demonstrated simple virtual mouse input. As the fingers on the right of the device are moved around, the on-screen mouse cursor follows. A selection may be made by either left or right ‘clicking’ (momentarily touching the surface on one side of the device). Other alternatives we have experimented with include using cursor dwell timeout or localised gestures such as using two fingers (on one side of the device) to indicate selection. We have combined these features to support actions such as selecting items from a pull-down menu (see Figure 5).

SideSight can also be combined with a traditional resistive touchscreen and more specifically stylus-based input to support some interesting multi-modal interactions. For example, the user can write on the device surface using the dominant hand, whilst movement of the other hand to the side of the device is used to generate independent movement information. Figure 5 gives an example of this where the stylus is used to mark up an electronic document whilst SideSight is used to pan the document horizontally and vertically. This is a fairly intuitive and powerful mode of interaction akin to how we write on real paper – typically both hands are used and the dominant one writes whilst the non-dominant one controls the movement of the paper.

CONCLUSIONS AND FUTURE WORK

We have demonstrated the use of sideways looking proximity sensing to extend user interaction beyond the physical display. SideSight is capable of retaining the kinds of gesture interactions that are supported by larger multi-touch surfaces, and which are beginning to become familiar to users. By choosing to extend the interaction area to the region around a small display it is possible to greatly increase both the richness and utility of interaction. We have also demonstrated the more traditional single cursor control as well as novel multi-modal interactions which combine stylus and touch input. It may also be possible to combine on- and off-screen gestures, for example supporting a finger sweep moving from the display surface through to the region beyond the display.

In our work to date we have focused on the use of proximity sensing in a situations where the device is resting on a surface. Not only does this enable bi-manual interaction, but means that hands and fingers are ergonomically supported which may result in less fatigue in use compared with free-space gesturing or scenarios where the mobile device is held. However, we also wish to explore scenarios where the user may be holding a SideSight device in one hand whilst gesturing in mid-air using the other for more immediate, albeit coarser grained interaction. We are also interested to explore using the sensors to determine how the device is being gripped and supporting interactive input by ‘stroking’ near or on the sides where the grip is not occluding proximity sensing – potentially making use of the top and bottom edges of the device and employing higher resolution sensing if required. SideSight is also capable of sensing passive IR reflective or active IR objects (rather than just fingers) and would for example support a multi-user pong game where two physical paddles are placed on either

side of a device to interact with a virtual bouncing ball. SideSight is still in its infancy and there are many interesting aspects to explore.

Power consumption is a critical issue for mobile device use and the current prototype is not optimized – being derived from a larger ThinSight display implementation [3]. We are exploring ways to reduce the number of IR LEDs (the main consumer of power) used compared to photodiodes, lowering the inactive update rate to sample only infrequently when no objects are detected, and redesigning sampling hardware from the charge-discharge technique in [3] to a buffered ADC approach which can also significantly reduce the time required for LED illumination during each sampling cycle.

We are also looking at other technologies for providing more accurate sensing at greater distances from the device. In the future we believe that it may be possible to print organic electronic versions of such sensors, and so we are also interested in exploring a SideSight configuration that has the entire casing covered in this type of proximity sensing material.

ACKNOWLEDGMENTS

The authors would like to acknowledge the helpful discussions and input of many people at MSR, including Nic Villar, Mike Sinclair, Bill Buxton and Ken Hinckley.

REFERENCES

1. Arnaut, L.Y. & Greenstein, J.S. Human Factors Considerations in the Design and Selection of Computer Input Devices. In Sol Sherr (Ed.), *Input Devices*. Boston: Academic Press, 73–74. 1988.
2. Hinckley, K., Pierce, J., Sinclair, M., Horvitz, E., Sensing Techniques for Mobile Interaction, In *Proceedings of UIST 2000*, CHI Letters 2 (2), pp. 91-100.
3. Hodges, S., Izadi, S., Butler, A., Rrustemi, A., Buxton, B., ThinSight: Versatile Multitouch Sensing For Thin Form Factor Displays, In *Proceedings of UIST 2007*.
4. Hollemans, G. et al., Entertaible: Multi-user multi-object concurrent input, *UIST '06 Adjunct Proceedings*.
5. LightGlove, <http://www.lightglove.com/White%20Paper.htm>
6. RPO, http://www.rpo.biz/DWT_White_Paper.pdf
7. Smith, J., White, T., Dodge, C., Paradiso, J., Gershenfeld, N., Electric Field Sensing For Graphics Interfaces, *IEEE Computer Graphics & Applns*, May/June 1998.
8. Virtual Keyboard, <http://www.vkb-support.com/index.php>
9. Vogel, D. and Baudisch, P., Shift: A Technique for Operating Pen-Based Interfaces Using Touch, In *CHI 2007*, pp. 657–666.
10. Wigdor, D., Forlines, C., Baudisch, P., Barnwell, J., Shen, C., LucidTouch: A See-Through Mobile Device, In *Proceedings of UIST '07*.