

# Stitching: Pen Gestures that Span Multiple Displays

Ken Hinckley<sup>1</sup>, Gonzalo Ramos<sup>1,2</sup>, Francois Guimbretiere<sup>1,3</sup>, Patrick Baudisch<sup>1</sup>, and Marc Smith<sup>1</sup>

<sup>1</sup>Microsoft Research  
One Microsoft Way  
Redmond, WA 98052  
+1 425 703 9065

<sup>2</sup>University of Toronto  
40 St. George St.  
Toronto, ON M5S 2E4  
+1 (416) 946-8491

<sup>3</sup>University of Maryland  
3267 AV Williams Bldg  
College Park, MD 20742  
+1 (301) 405 7952

{kenh, baudisch, masmith}  
@microsoft.com

bonzo@dgp.toronto.edu

francois@cs.umd.edu

## ABSTRACT

Stitching is a new interaction technique that allows users to combine pen-operated mobile devices with wireless networking by using pen gestures that span multiple displays. To stitch, a user starts moving the pen on one screen, crosses over the bezel, and finishes the stroke on the screen of a nearby device. Properties of each portion of the pen stroke are observed by the participating devices, synchronized via wireless network communication, and recognized as a unitary act performed by one user, thus binding together the devices. We identify the general requirements of stitching and describe a prototype photo sharing application that uses stitching to allow users to copy images from one tablet to another that is nearby, expand an image across multiple screens, establish a persistent shared workspace, or use one tablet to present images that a user selects from another tablet. We also discuss design issues that arise from proxemics, that is, the sociological implications of users collaborating in close quarters.

## Categories and Subject Descriptors

H.5.2 [Information Interfaces and Presentation]: Input Devices and Strategies, Interaction Styles

## General Terms

Human Factors, Design

## Keywords

pen computing, mobile devices, co-located collaboration, proxemics, synchronous gestures, spontaneous device sharing

## 1. INTRODUCTION

With wirelessly networked mobile devices rapidly becoming ubiquitous, it is increasingly important to find effective methods for dynamically binding together mobile devices in support of sharing information, collaborating and communicating with other persons, or performing complex tasks involving large documents and multiple sources of information. The *spontaneous device sharing problem* poses the following question: how can a user dynamically forge a purposeful connection between two or more

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

AVI '04, May 25-28, 2004, Gallipoli (LE), Italy  
© 2004 ACM 1-58113-867-9/04/0500..\$5.00

target devices that do not *a priori* know one another's network address? This is not only a significant research problem for system implementation, interaction design, and social and behavioral observation, but also is becoming a problem with a pressing need for practical solutions.

Researchers have recently discovered a new class of techniques that use synchronous user actions (known as *synchronous gestures*) to address the spontaneous device sharing problem without forcing the user to manually enter network addresses. Examples of these techniques include holding two devices together and shaking them [13], bumping a pair of devices together [12], or simultaneously pressing a button on each device [19]. However, the design space of synchronous gestures is still not well understood and needs further exploration.

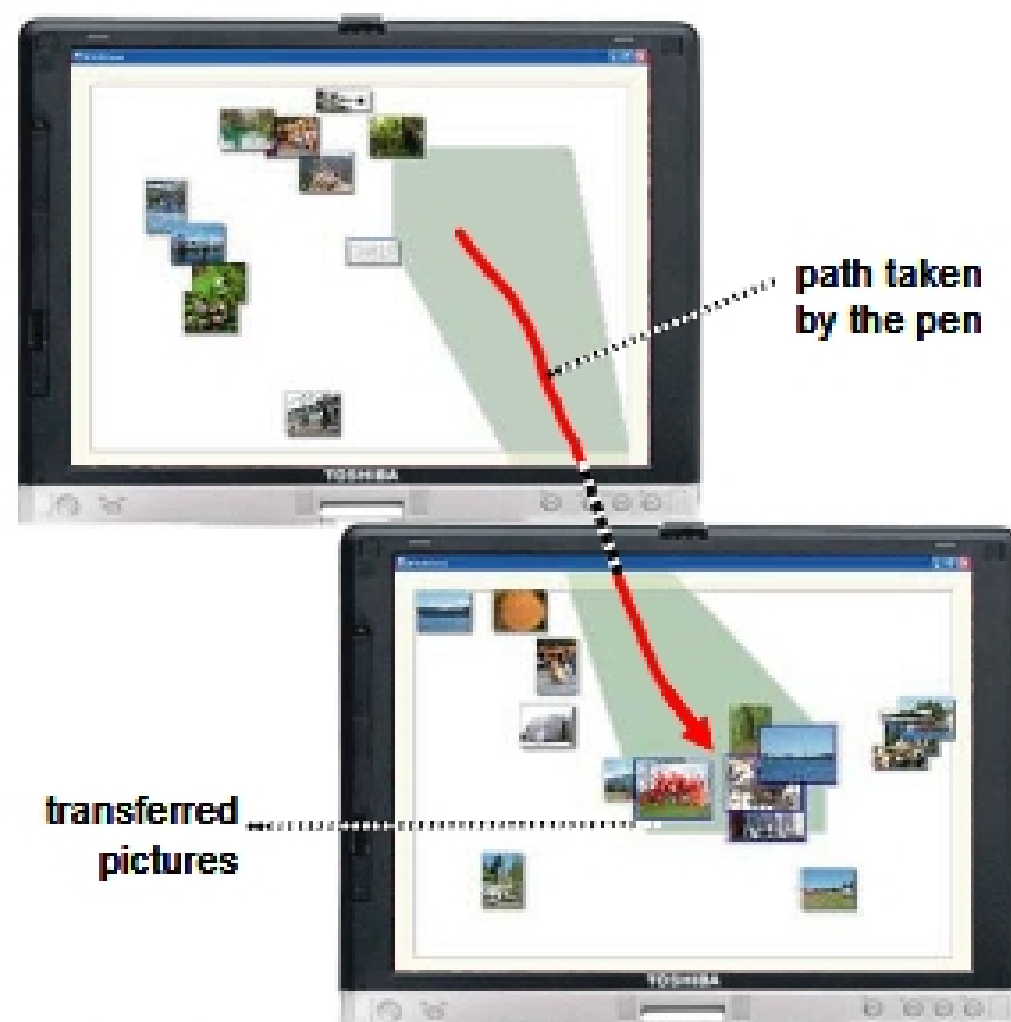


Figure 1. Here, a user gives some photos to another user by stitching from the top tablet to the bottom tablet.

Here, we contribute *stitching*, a new type of synchronous gesture that uses commonplace pen input capabilities to establish and manage serendipitous connections between pen-operated mobile devices. A stitching gesture is a pen gesture that spans multiple displays, consisting of a continuous pen motion that starts on one device, skips over the bezel of the screen, and ends on the screen of another device (Fig. 1). We show how stitching serves as the basis for a flexible connection architecture that can be

implemented on a variety of pen-operated devices. To explore the pragmatics of stitching, we implement a prototype photo sharing application for the Tablet PC that supports operations such as copying images from one tablet to another that is nearby, establishing a persistent shared workspace for collaboration, expanding an image across multiple screens, or using one tablet to display a slideshow of images that a user selects from another tablet. Preliminary usability testing suggests that users can easily make pen gestures that span displays, and find it compelling to have a straightforward means to perform cross-device operations.

In addition to the new technique of stitching, more generally we contribute six requirements for the spontaneous device sharing problem, phrased as design questions:

- **Connection:** How is a connection established?
- **Command:** What type of connection is required?
- **Operands:** What information is shared?
- **Geometry:** What is the spatial relationship between devices?
- **Coexistence:** How do connection gestures coexist with traditional interactions or naturally occurring user behaviors?
- **Proxemics:** How do users share physical space?

The literature presently lacks such a discussion of general requirements. In this sense our research represents a framework that considers cross-device operations in general, including the system, interaction, and social issues that arise.

## 2. RELATED WORK

Our work is the first to address all six requirements listed above. Several previous systems foster collaboration between ubiquitous devices, but require special hardware such as overhead cameras, visual identification tags, or radio-frequency identification (RFID) tags on each device [15,20,27,28]. Some systems have focused on new ways to use devices together, but may require manual configuration and calibration, including manual entry of network addresses [17] and the geometry of displays [14,26].

Some existing standards are relevant to the spontaneous device sharing problem. Bluetooth ([www.bluetooth.org](http://www.bluetooth.org)) supports device discovery, but offers no way to distinguish a target device from other devices within radio range: the user must search through a potentially long list of devices to find a symbolic name. Infrared Data Association (IrDA, [www.irda.org](http://www.irda.org)) allows data transfer between two devices. However, IrDA requires that devices remain still during communication, and provides no flexibility in the relative position and orientation of the devices.

### 2.1 Synchronous Gestures

Synchronous gestures are distributed patterns of user activity that occur together in time, or in a specific sequence in time [12]. The literature currently describes three examples of synchronous gestures: bumping [12], Smart-Its Friends [13] and SyncTap [19]. Bumping devices together can be sensed and used to create a shared display that spans two or more devices [12]. Smart-Its Friends and SyncTap recognize when a gesture occurs on two devices at the same time. Smart-Its Friends infers a connection when two devices are held together and shaken. If a “friend” device previously identified in this manner comes within range of another device’s low-power radio, the device beeps, thus creating awareness of other nearby users. No other features are supported.

With SyncTap, a user can simultaneously press “Sync” buttons on two separate devices to connect them. SyncTap’s main strength is that it offers a lowest common denominator solution, since nearly all mobile devices could include a suitable Sync button. However, actually doing anything with the connection requires additional steps. For example, to establish cursor migration between a laptop and a desktop PC, the user must first perform a SyncTap gesture, and then switch to the mouse on the desktop computer to indicate the edge of the screen that should be linked to the other device. Stitching can support similar types of operations in a single fluid pen-based command phrase [4], without multiple steps that require switching devices. SyncTap also discusses the need to avoid *collisions* of multiple overlapping Sync actions; if a collision occurs, the user must repeat the gesture. This is a less critical issue for stitching since stitches can only collide if they exhibit the same timing *and* the same geometrical properties.

### 2.2 Pick and Drop

Pick and Drop [18] allows users to pick (copy) an item from one screen and drop (paste) it onto the screen of another nearby device. Stitching makes significant contributions to the literature beyond Pick and Drop, primarily by recognizing the requirements for a versatile interaction paradigm for combining multiple mobile devices. Our research identifies general design requirements of the spontaneous device sharing problem and illustrates how distributed pen input represents a rich design space that provides some elegant solutions addressing these requirements. Stitching provides users multiple ways to copy or move information between devices, link the displays of devices, and to otherwise use devices together. Stitching uses the geometrical information from the pen to automatically determine the spatial relationship between a pair of devices. This automatic calibration technique has not been previously implemented or reported in the literature. Our research is also the first to recognize that proxemics (how users share physical space) may yield critical design insights for the spontaneous device sharing problem.

Stitching also differs from Pick and Drop in its technical requirements. Pick and Drop requires a pen with an embedded unique ID, whereas stitching uses only a series of time-stamped pen events to infer a connection. This is a subtle technical difference but has tremendous practical implications: to our knowledge, there are *no* mobile devices available that support the pen ID feature required for Pick and Drop, whereas Stitching gestures can be supported between any pen-based mobile devices with compatible pen technologies, as summarized in Fig. 2. This technical difference may become less important if a pen ID capability becomes commonplace on future mobile devices.

	PDA (PocketPC, Palm)	Tablet (Tablet PC)
Stitch from PDA to...	<i>Yes</i> PDA/PDA stitches can use any stylus	<i>Yes</i> Tablet/PDA stitches must use the tablet’s stylus
Stitch from Tablet to...	<i>Yes</i> Tablet/PDA stitches must use the tablet’s stylus	<i>Sometimes</i> Tablet/tablet stitching requires compatible pens
Pen Technology	<i>resistive touchscreen</i> senses any solid object in contact with screen	<i>magneto-inductive</i> active digitizer senses only special pen

Figure 2. Summary of pen technologies used by Tablets and PDA’s, and the implications for stitching.

## 2.3 Proxemics and Social Issues

*Proxemics* is the study of how people use the invisible bubble of space that surrounds an individual [2,11,25]. Sociologist E. T. Hall [11] distinguishes two distances within arm's reach, intimate and personal, with social and public distances beyond that. No single distance between persons is "best" in all situations. Factors such as how well the persons know one another, the gender of each person, and what the persons are trying to accomplish can all influence selection of a social distance that feels comfortable to both persons [2,24]. It is often assumed that users prefer face-to-face interaction, but proxemics research suggests that people's choice of relative body orientation influences communication patterns and changes with the task at hand [7,11,21,24].

Although co-located collaboration involves users working in close physical proximity, this literature rarely mentions proxemics. Scott proposes territory-based techniques for tabletop interaction [22,23]. Rodden et al. [21] discuss social issues surrounding face-to-face consultations, but do not reference the sociological literature. Thus, we believe our discussion of proxemics is a contribution to co-located collaboration in general, as well as to the more specific problem of spontaneous device sharing.

## 3. THE REQUIREMENTS OF STITCHING

Stitching represents an explicit user command to combine the resources of multiple computers. We wanted stitching to provide a flexible and potentially extensible facility that would support a number of different ways of combining devices, rather than supporting only a single operation or a very limited set of options, as in previous systems (e.g. [13,18,28]). To achieve this, stitching addresses the following central design questions:

- **Connection:** How is a connection established? A user must name the devices that are involved in a multi-machine operation, and the system needs to provide feedback to the user(s) of those devices that a connection has been established.
- **Command:** What type of connection is required? The user needs to be able to choose among several possible ways to combine the devices. Does the user want to copy a file from one device to another? Establish a persistent shared workspace for collaboration? Expand an image across multiple screens? These all represent *multi-device commands* that transcend the barriers between devices.
- **Operands:** What information is shared? Multi-device commands may require operands, such as which file to copy to another computer. Users need mechanisms to select one or more objects as part of a stitching gesture.
- **Geometry:** What is the spatial relationship between the devices? Several previous systems support features, such as cursor migration or combining the screens of two devices, that require knowledge of where one display is relative to another [14,20]. Stitching uses the information provided by the pen to infer the spatial relationship between devices. This also allows us to provide graphical feedback for multi-device operations that appears to span devices, as seen in Fig. 1.
- **Coexistence:** How do connection gestures coexist with traditional interactions or naturally occurring user behaviors? In the case of stitching, connection gestures must coexist with existing uses for the pen including widget interactions,

inking, character entry, and naturally occurring human-human communicative gestures (such as waving the pen near the device while discussing the contents of the screen).

- **Proxemics:** How do users share physical space? Proxemics suggests that the arrangement of spaces can influence communication; as Hall writes, "what is desirable is flexibility... so that there is a variety of spaces, and people can be involved or not, as the occasion and mood demand" ([11], p 110). Interaction techniques that form impromptu associations between mobile devices should likewise support the range from users who know each other well and want to work closely together, to users who are strangers and may want to exchange files while maintaining social distance.

## 4. THE MECHANICS OF STITCHING

The above design questions suggest that stitching represents a new class of interaction techniques that could be implemented in a variety of ways. We now discuss the general concept of stitching in reference to a proof-of-concept photo sharing application, called *StitchMaster*, which we implemented for the Tablet PC. With digital photography becoming widespread, sharing photos with others is a task of interest to many persons. Also, many of the semantics that we wanted to explore with stitching, such as expanding an image across multiple screens or copying objects from one screen to another, represent useful and compelling operations for digital photographs. To begin, each user launches *StitchMaster* on his own tablet, which displays that user's photo collection as a set of thumbnail images.

### 4.1 Establishing a Connection

With stitching, a user names the devices to connect simply by moving the pen across them. Since there is a natural order implied by the gesture, stitching also establishes which machine is the sender of information, and which machine is the receiver. Some connection techniques are inherently bidirectional [13,19,28] and do not naturally provide this information.

In order to recognize a pen gesture that spans displays, a set of co-located mobile devices need to synchronize pen events (consisting of absolute pen  $(x,y)$  location on the screen, whether or not the stylus is in range of the screen, and a timestamp) with one another. There are a variety of well-understood technical means for devices to discover one another, so our research focuses on the *user interaction required to form a purposeful connection between two (or possibly more) specific target devices*, and not the system details of how background communication occurs between a set of co-located candidate devices. For example, *SyncTap* [19] simply multicasts button press timing information to all other devices on the same subnet. We could alternatively use a discovery mechanism such as that included in Bluetooth, or the Intentional Naming System [1]. We are also investigating the use of wireless signal strengths as a way to restrict synchronization communications to other nearby devices [3,12].

In our current prototype system, each participating device sends its pen events to a *stitching server*, which typically is hosted on one of the mobile devices (it may alternatively be hosted in the environment to offload computation from the mobile devices). The stitching server synchronizes time between the devices [9,12] and looks for matching pen traces; when a match is found, the server sends a stitching event that informs the two devices of each