To Swipe or Not to Swipe

Ahsan Javed, Silicon Laboratories, www.silabs.com

"We respond to gestures with an extreme alertness and, one might say, in accordance with an elaborate secret code that is written nowhere, known by none, and understood by all."

– Edward Sapir, Anthropologist-Linguist[1]

Gestures are one of our most fundamental and universal forms of nonverbal communication. Whether it's pointing to select an object out of a group, waving to say hello or curling your fingers to beckon someone to come closer, gestures play an important role in how we interact with others. The advent of human interface technologies has extended gestures to include interaction with electronics systems as well, and the explosion in smart phone growth over the past decade has placed this technology into the hands of the masses.

Unlike human-to-human gesturing, human-machine gesturing does not have a foundation of thousands of years of history to use as a context. This poses a challenge for human interface system designers. If a user performs a gesture, or a "voluntary motion of hands and/or arms to express a clear action,"[2] how does the machine (i.e., smart phone, tablet computer or point-of-sale device) know how to react in a manner that is commensurate with the user's intentions? Is the action easy to replicate, and does it produce repeatable results? How can gestures be designed to be intuitively easy to comprehend rather than frustrating? How does a system designer minimize the pain of "learning" gestures for the user?

Pinch-to-zoom is one of the most universal gestures on handsets and tablets today. This gesture is used to enlarge documents, making them more readable, and to zoom in on photos, revealing additional details. In the physical world, if one was asked to "zoom in" on a photograph, the instinctive response is to move the photo closer to one's face. Obviously the closer something is, the easier it is to discern features and minutiae. Pinch-to-zoom is now ubiquitous and is an expected feature on all modern touchscreen devices, but this was not always the case. This feature

was subtly taught to consumers, and unlike other less intuitive gestures, it stuck and is now second nature.

Action-Response

B.F. Skinner, the famous psychologist, articulated the idea of behaviorism and reinforcers. The basic concept is that a stimulus, negative or positive, can either encourage or discourage a behavior. This concept can be applied to human interface systems and gesture design and is one of the reasons that the “swipe to scroll” gesture is so intuitive. When one is reading a book or magazine the action to “scroll” to the next page is identical to the actions required on a modern touchscreen device. Since the device operates in a predictable manner, the gesture is reinforced and learned easily. This is one reason why toddlers find it easy to interact with touchscreen systems: the action and response is predictable and instantaneous with minimal learning required and little frustration.

In contrast to current touchscreen interfaces, when the now defunct Palm introduced its Graffiti text entry system on the popular range of Palm Pilot Personal Digital Assistants (PDAs), the technology was hailed as revolutionary. The system was much more accurate than state-of-the-art handwriting recognition systems at the time and allowed relatively fast text entry without the use of a keyboard. What Palm did was to simplify the alphabet and numbers into a series of gestures that were relatively easy to remember and could be consistently recognized by their PDAs. The main issue with Graffiti was that the gestures had to start and end in a particular method and they had to be learned. Every Palm Pilot shipped with a cheat-sheet that constantly reminded users of what the correct method for entry was in order to “type” a certain character. Over time Graffiti became relatively popular, although the learning cycle was relatively complex and remained a barrier to entry for some users.

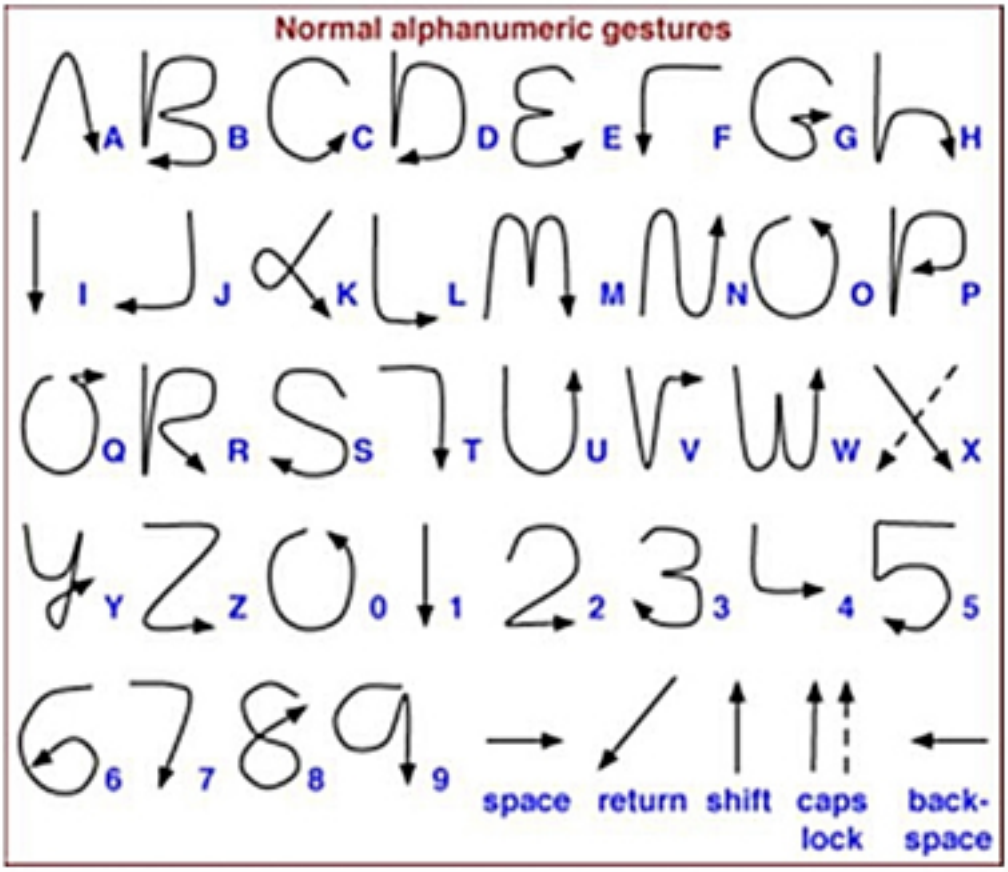

Figure 1. Examples of alphanumeric gestures for PDAs

Unlike Palm's Graffiti, effective gestures are easy to remember and reproduce while providing a meaningful function that is superior to using a simple toggle or button. Most modern touchscreen handsets still use a standard QWERTY keyboard for text entry rather than using a gesture-based input system or handwriting recognition since these techniques are less efficient than tried-and-true keystroke typing.

Tuning Gestures

Gesture sensing is not automatic. Every gesture, whether touch or touchless, has to be tuned to the requirements of the system and the use case. Similarly, and almost as important, the system needs to be tuned to ignore inadvertent movements or false gestures. Automatic faucets and towel dispensers in public restrooms often incorporate a short delay mechanism to ensure that a hand is actually in front of the system waiting for a stream of water or a paper towel. This brief delay ensures that resources are saved and that the system does not respond erroneously to people walking by or other false inputs. On touchscreen devices, when navigating through a selection of photos, for example, scrolling through the images can be achieved with a finger but not with a palm or the side of a hand. The system is designed to recognize that a finger swipe is a valid gesture while a palm swipe is most likely an errant touch.

Many factors go into defining a gesture and the resulting action including the speed, motion and surface area. In the case of touchless gestures, using infrared proximity or other optical sensor technologies, there are other considerations such as the

reflectivity surface of the detectable object, ambient light conditions and object depth. Using Silicon Labs' IRslider2EK evaluation board as an example, the board can recognize left and right swipe gestures and also recognize set points when pausing over a certain location. Alternately if a user waves his hand slowly over the board, neither the set point nor the swipe gesture is recognized. Each of these actions has the potential to have a user interface function associated with it.

Figure 2. silicon Labs' IRslider2EK,

Touch gestures have become commonplace as the popularity of smart phones has increased. Multi-touch gestures, or gestures involving two or more fingers, allow simple and easy navigation of the user interfaces of these increasingly complex devices. These multi-touch gestures include many input mechanics such as tap, swipe, pinch and un-pinch, scroll, drag, rotate and flick.

Touchless gestures, beyond simple proximity sensing, are relatively new and are implemented using slightly different techniques. While many methods such as optical imaging can be employed to achieve touchless gesture sensing, one of the most computationally and energy efficient gesture sensing techniques involves the use of an active infrared sensor and infrared LEDs (irLEDs).Touchless gesture sensing using an IR-based approach can be accomplished using a position-based approach or a phase-based approach.

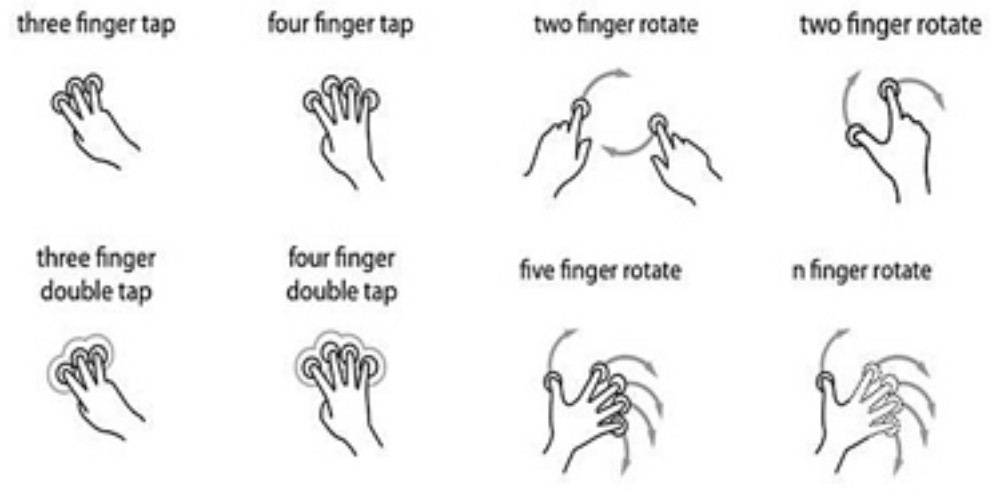

Figure 3. Examples of multi-touch gestures.[3]

Position-based motion-sensing involves three primary steps:
* First, raw proximity data inputs are converted into usable distance data.

* The second step uses this distance data to estimate the position of the target object.
* The final step compares the timing of the position data to determine whether a gesture has occurred.

When initiating the development of such a system, a target object is held above the sensing array at a fixed distance, and the analog to digital converter (ADC) counts are noted. For example, if a hand is held a set distance away from the sensor and yields x ADC counts, then future readings of x ADC counts imply that a similar object is at that distance from the sensor. Taking these measurements across various distances provides the system designer with a good set of ADC counts to distance data. This data needs to be taken for all the irLEDs in the gesture sensing array, and some tuning will be required to ensure that the LED's light output variation is accounted for across systems. This data is then used to estimate the position of a target above the system based on the relative reflection from the irLEDs. Finally, keeping track of the timing of the target object's movement, direction and speed allows the system to recognize and acknowledge a valid gesture.

In a position-based system, it is critical to keep track of the time stamps for the entry, exit and current positions of the target in the detectable area -- the area around a proximity sensing system where valid measurement can be made. Most gestures can be recognized easily using this timing and position information; however, each system will need to be custom-tuned for each application and the system designer's preference.

Phase-based gesture sensing uses solely timing data and the raw ADC output to detect gestures and never requires distances to be calculated. For a multi-LED proximity sensing system, a phase-based approach uses the changes in reflectance over the sensor and the direction of movement to acknowledge the gesture. The peak reflectance for an irLED will occur when a target object is directly above it so the direction of the swipe can be determined by which LED's reflectance rises first. In the IRslider2EK example, when an object passes from left to right above the board, the left irLED provides peak reflectance before the right irLED so the phase system recognizes that a right swipe gesture has occurred.

Position-based gesture sensing can report back the physical location of a detectable object and thus provide some additional control over a system, such as acknowledging a hover command or "hold-to-select" gesture. This approach can be fooled, however, since the accuracy of the positional calculation is based on the shape of the infrared LED's light output and the shape of the detectable object. For example a finger at 2 cm can reflect the same amount of light as a hand at 15 cm. The phase-based approach provides a more robust method of providing gesture recognition, but does not provide any positional information so hold-to-select gestures cannot be implemented. The best solution is a combination of the position- and phase-based approaches to leverage the benefits of both techniques to develop a truly robust gesture recognition system.

Future electronic devices will incorporate more gestures or even combinations of

gestures to achieve advanced interface functions. Modern tablets and smart phones with built-in touchscreens and cameras already incorporate a plethora of sensors such as capacitive touch sensors, optical sensors, accelerometers and image sensors to interpret multiple forms of human input. Using combinations of these sensors in unison can give rise to innovative human interface experiences such as using the touchscreen to focus an embedded camera while using the proximity sensor to control the zoom level.

The advent of advanced imaging technologies such as Microsoft's Kinect system has ushered in even more sophisticated and exciting applications for gesturing systems. Soon automobiles will be able to recognize tired drivers as their head sags; users will be able to navigate through interfaces using only their eyes; and video billboards will display proximity sensitive messages. Environmentally aware and user-aware electronics will usher in a new era of convenience and ground-breaking enhancements for all types of consumer electronics. The interactions are only limited by the human interface system designer's imagination.

# # #

Ahsan Javed joined Silicon Laboratories in 2008 to help develop the company's emerging digital isolation business prior to transitioning to the company's fast-growing human interface products group. Mr. Javed has more than 10 years of industry experience spanning the automotive, consumer and industrial markets. Prior to joining Silicon Labs, Mr. Javed spent eight years at Freescale Semiconductor (formerly Motorola) where he held product engineering and product marketing roles for Freescale's microcontroller division. Mr. Javed holds a M.E.E. from Rice University and an M.B.A. from the University of Texas at Austin.

1 "Nonverbal Communication in Human Interaction" By Mark L. Knapp, Judith A. Hall, Cengage Learning 2009
http://books.google.com/books?hl=en&lr=&id=j5HIIfRUPm0C&oi=fnd&pg=PR5&dq=human+gesture+communication&ots=LsuaCVPmMZ&sig=gLCJKRsgSVYUAfnfSTwYvx_P23I#v=onepage&q=gesture&f=false
[1]2 Device based gesture recognition Damien Zufferey, Department of Informatics (DIUF), University of Fribourg
http://diuf.unifr.ch/diva/SeminarGest09/reportZufferey.pdf [2]
3 http://gestureworks.com/features/open-source-gestures/
[3]

Source URL (retrieved on *09/18/2014 - 12:46am*):
http://www.ecnmag.com/articles/2011/09/swipe-or-not-swipe?qt-most_popular=0

Links:
[1] http://books.google.com/books?hl=en&lr=&id=j5HIIfRUPm0C&oi=fnd&pg=PR5&dq=human+gesture+communication&ots=LsuaCVPmMZ&sig=gLCJKRsgSVYUAfnfSTwYvx_P23I#v=onepage&q=gesture&f=false

[2] http://diuf.unifr.ch/diva/SeminarGest09/reportZufferey.pdf
[3] http://gestureworks.com/features/open-source-gestures/