Touchless Gesture Goes Mainstream

Francis MacDougall
Sr.Director, Gesture Technology

QUALCOMM®

QUALCOMM






Fortune 500 Company

USD$105 Billion Market Cap

Celebrating more than 25 years of driving the evolution of wireless communications.

Over 23,000 employees worldwide.

World's largest fabless semiconductor company, #1 in wireless.

Member of the S&P 100 & 500 Indexes

GREAT PLACE TO WORK® INSTITUTE INDIA — TOP 25 BEST companies to work for 2010

FORTUNE® 500 2011

FAST COMPANY 2010 MOST INNOVATIVE COMPANIES

Institutional Investor

2011 All-America Executive Team Award Winner

Forbes 2010 WORLD'S LEADING COMPANIES GLOBAL 2000

FORTUNE 100 BEST COMPANIES TO WORK FOR 2011

COMPUTERWORLD 100 BEST PLACES TO WORK IN IT 2010

FORTUNE WORLD'S MOST ADMIRED COMPANIES® 2011 #1 - NETWORK AND OTHER COMMUNICATIONS EQUIPMENT

Touchless Gesture Recognition: Overview

- Next complimentary evolution of UI beyond touchscreens
- Specific use cases targeted are touch free operations, for example:
  - Driving
  - Leaning back from a docked tablet device
  - Page flipping and Media Player control
  - Gaming
  - Remote media/TV control
  - Mouse

Which Gestures To Use ?

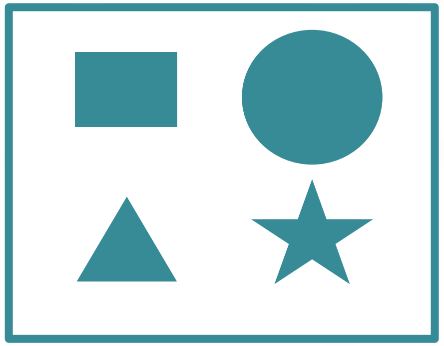
Technology Driven Gestures

- Certain technologies allow for specific hand motions and poses to be tracked
- Build a set of gestures based on what the technology can do
- Problem:
  - Resulting gesture set may be limited or unnatural or difficult to learn

User Driven or "Natural" Gestures

- What are the intuitive gestures that a person would use to perform a specific task?
- Several studies have been done generating an Alphabet of intuitive gestures
- Problem:
  - What a user does "naturally" may be extremely difficult to track
  - Differentiating between a large "natural" gesture set can be difficult, even for the user

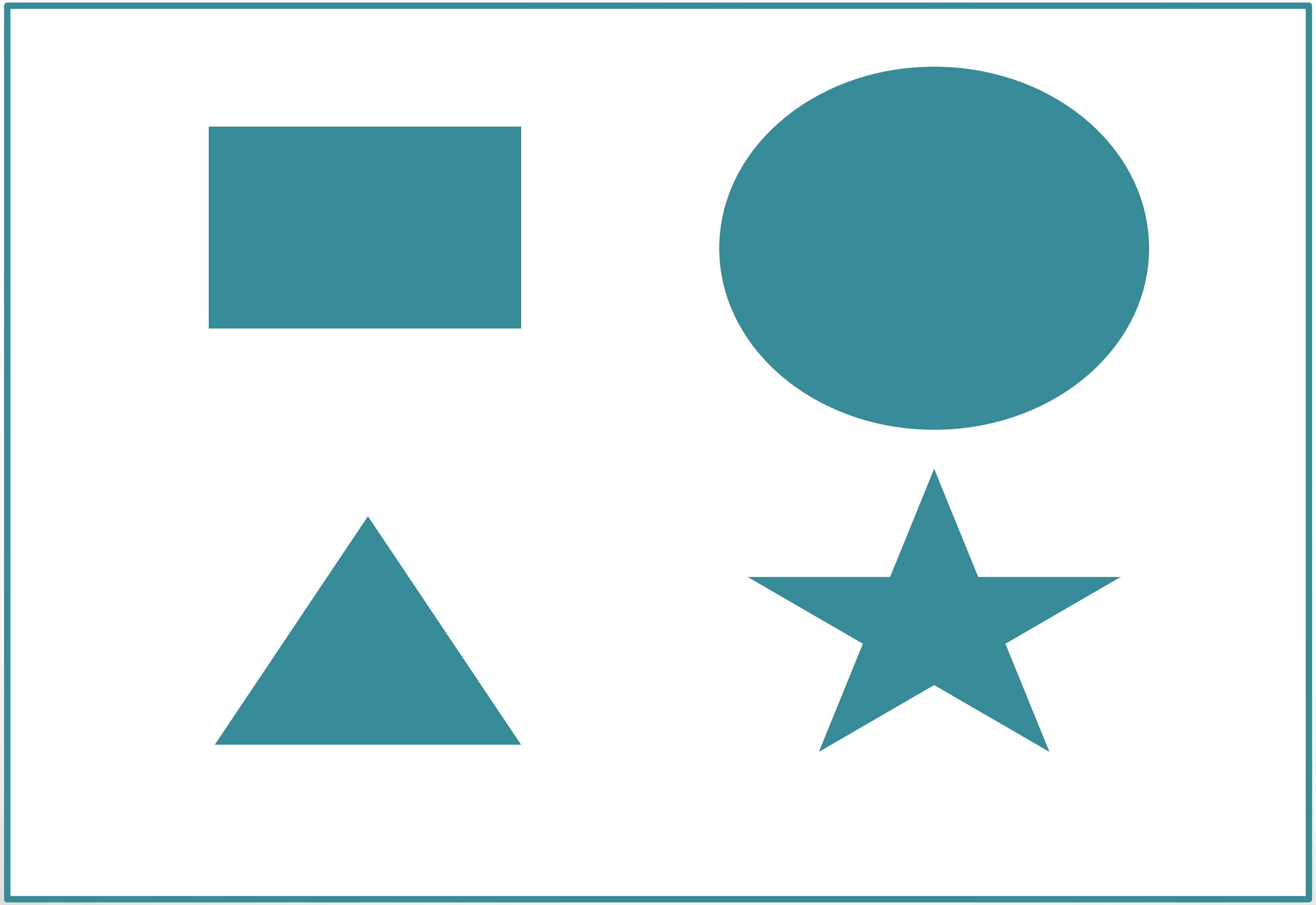
Example Natural Gesture Study: “Target Selection”



Instruction to user:

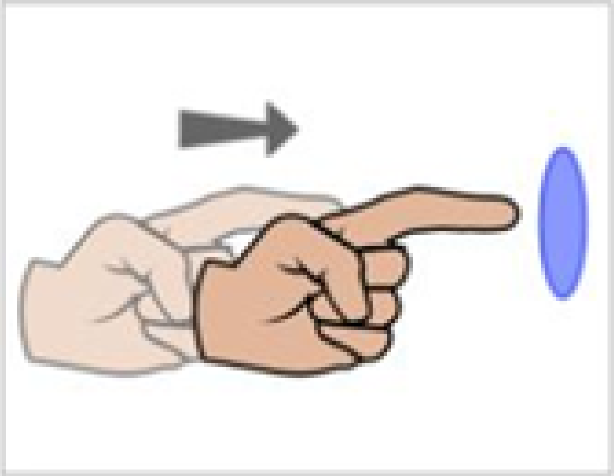
“Select the biggest shape”

(they are aware they cannot touch the screen)

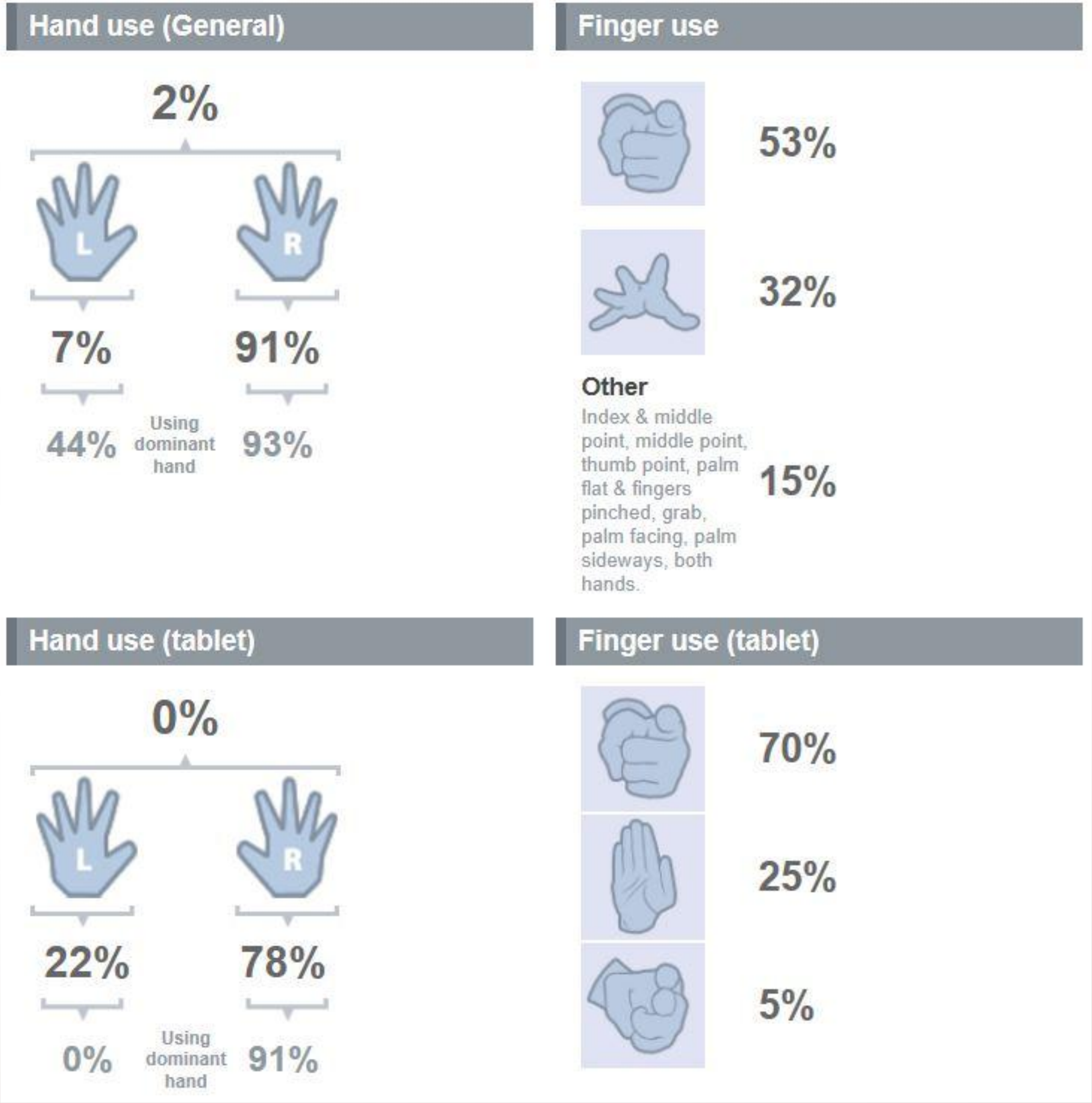


Gesture to select the biggest shape.

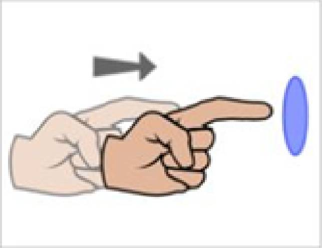
Study Result:



“Poke”

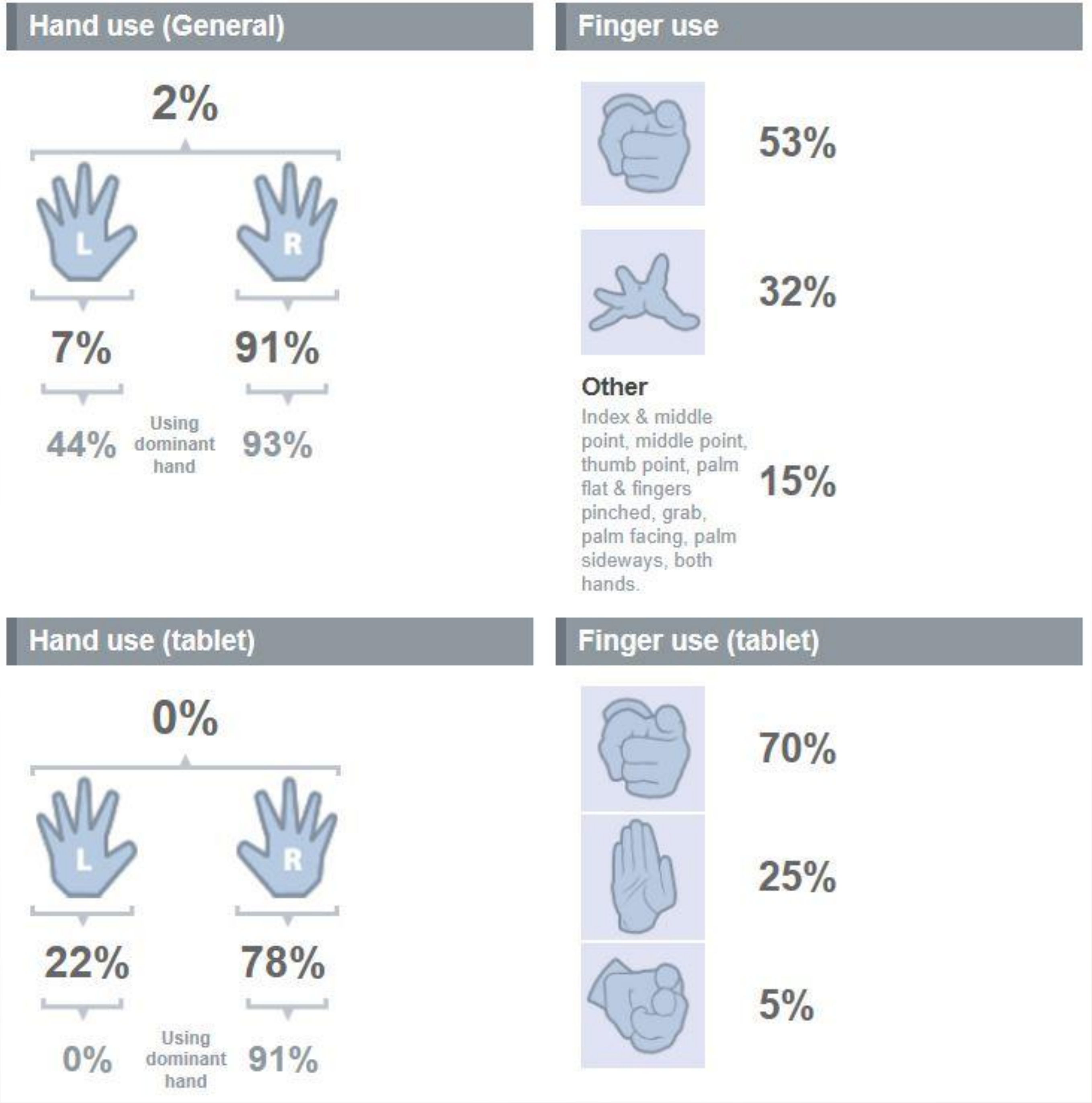


Poke:

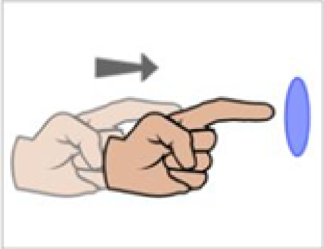


Tracking technology solution for this gesture needs to handle:

- To get 85%, one or more fingers in forward position (what if we want to save the “multiple fingers” for a different action?)
- To get 95%, full vertical palm tap or single finger poke (very different gestures that could be otherwise useful)
- What about finger angle versus the direction of the hand motion?
- What about the problem that the user moves their selection point as they poke?



Poke:

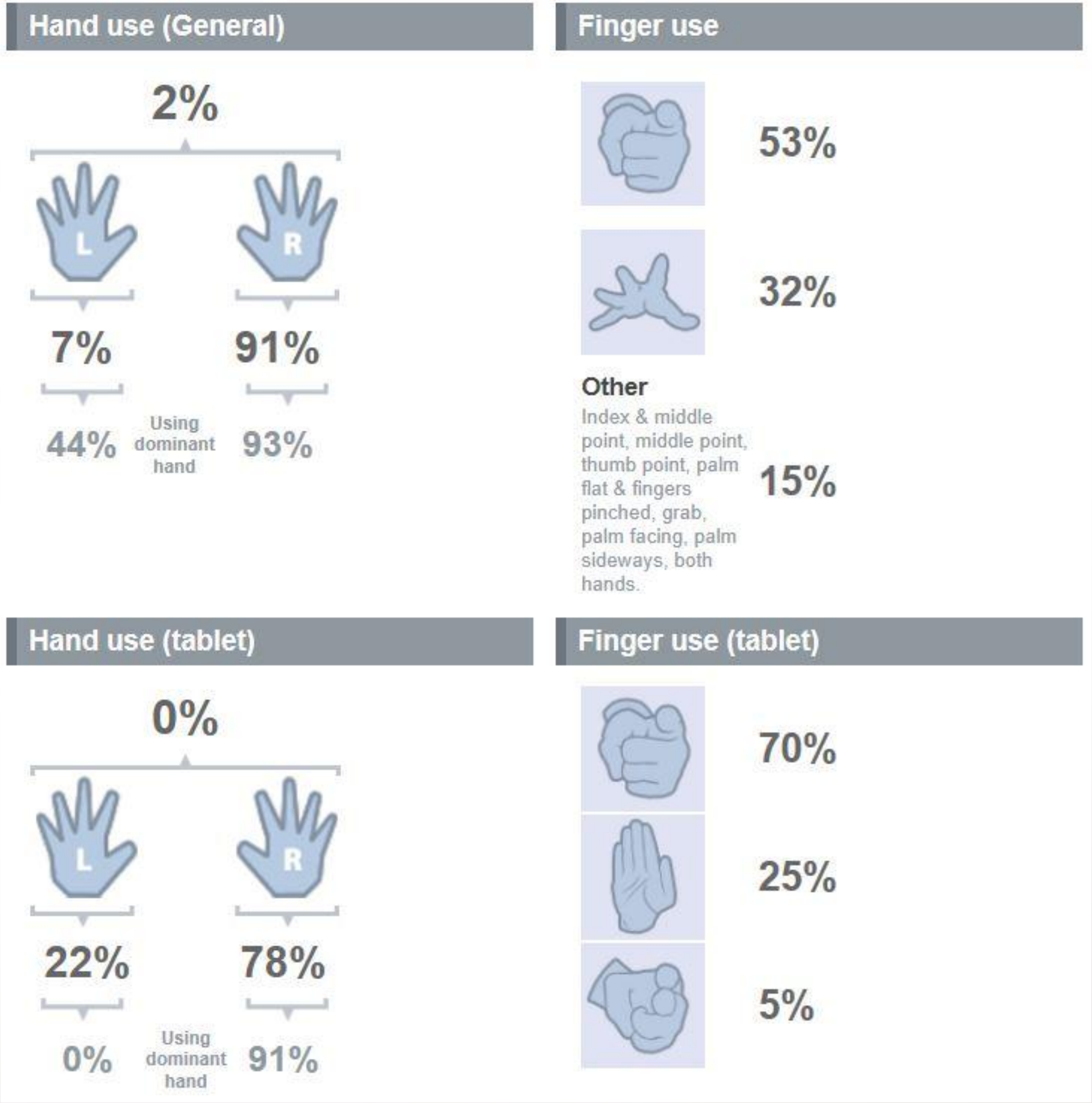


When users were told to restrict to a specific gesture:

- Single finger poke was considered natural and intuitive and easy to learn and remember

Findings:

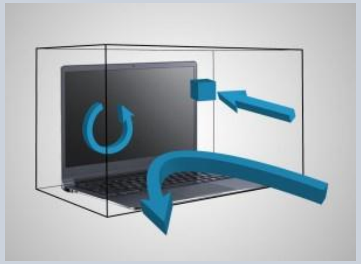
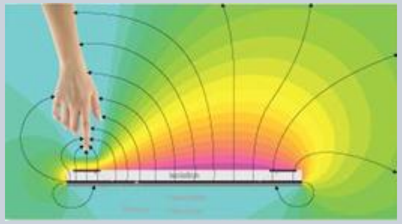
- Find gestures that can be tracked but are “easy to learn” and users will conform to them
- The UI can significantly guide the user to the “right” gesture



Gesture Use Cases

Use Case	Device	User Context	Applications
No-Look (< 1 foot)	Smartphone Embedded Car	Driving Working Out Desktop Belt holster	Voice Call Answer Mail Reader Media Player
Touch Free (< 1 foot)	Smartphone / Tablet Embedded Car Laptop	Mounted Desktop Driving	eReading Presentations Media Player Mail Reader Photo Album Viewer Navigation / App switching Gaming
Lean Back (1 to 5 feet)	Tablet Embedded Car	Airplane Couch Lying in bed Kitchen Back Seat of Car	eReading Presentations Media Player Mail Reader Photo Album Viewer Video Call Gaming
Full Body (>5 foot)	Set Top Box Embedded TV	Living Room	Gaming Remote Control TV Media Player

Hardware Technologies For Touch Free Gesture Detection

2D Image Sensors	3D Image Sensors	Other
		  
Built-in user facing camera Infrared camera sensors	Kinect Structured Light PMD Time Of Flight	Ultrasound Near Field Imaging
Existing 2D sensors are available today and are low cost and widely deployed. Infrared to follow: 2014	Cost of 3D sensor solutions delays CE deployment 2013: Accessories 2014: Embedded	Low cost and low power solutions will be deployed as gesture specific hardware: Ultrasound / NFI: 2013

Products

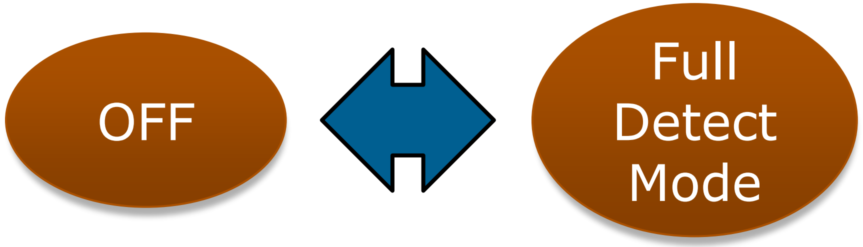
Mass Commercialization Devices for Touch Free Gesture Control

MARKET LAUNCHED DEVICES – PANTECH VEGA LTE SMART PHONE

Gesture Application Types – Event Driven Applications

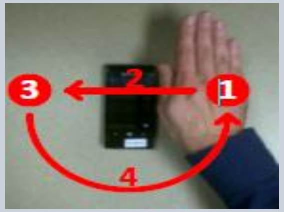
This is an example of an "***Event Driven***" application:

- Can run sensor at full speed for short duration and detect swipe without requiring engagement gesture
- Could choose to only run if phone is on its back or at rest based on motion sensor input



Example: "No-Touch" Call Answer

- Right swipe: *Answer and turn on speakerphone*
- Left swipe: *Discard / send to voice mail*
- In call:
  - Down/Up swipe: *Mute/unmute microphone*
  - Left swipe: *Hang up call*

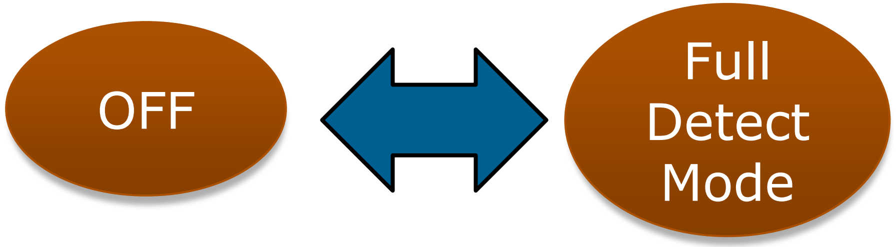
Gesture	Description	Uses
Near Swipe 	**Tracks a hand passing within 6 inches in front of the camera sensor. Sends Swipe Left and Right messages.**	**Touch-free call answer, in-car gesture mail "no-look" control, page turns, etc.**

MARKET LAUNCHED DEVICES – Samsung MV900F Digital Camera

Gesture Application Types – User Mode Driven Applications

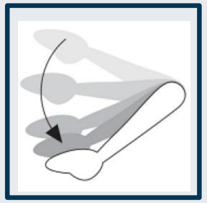
User explicitly turns on gesture mode:

- In this case, all tracking can be performed continuously until user turns off the mode

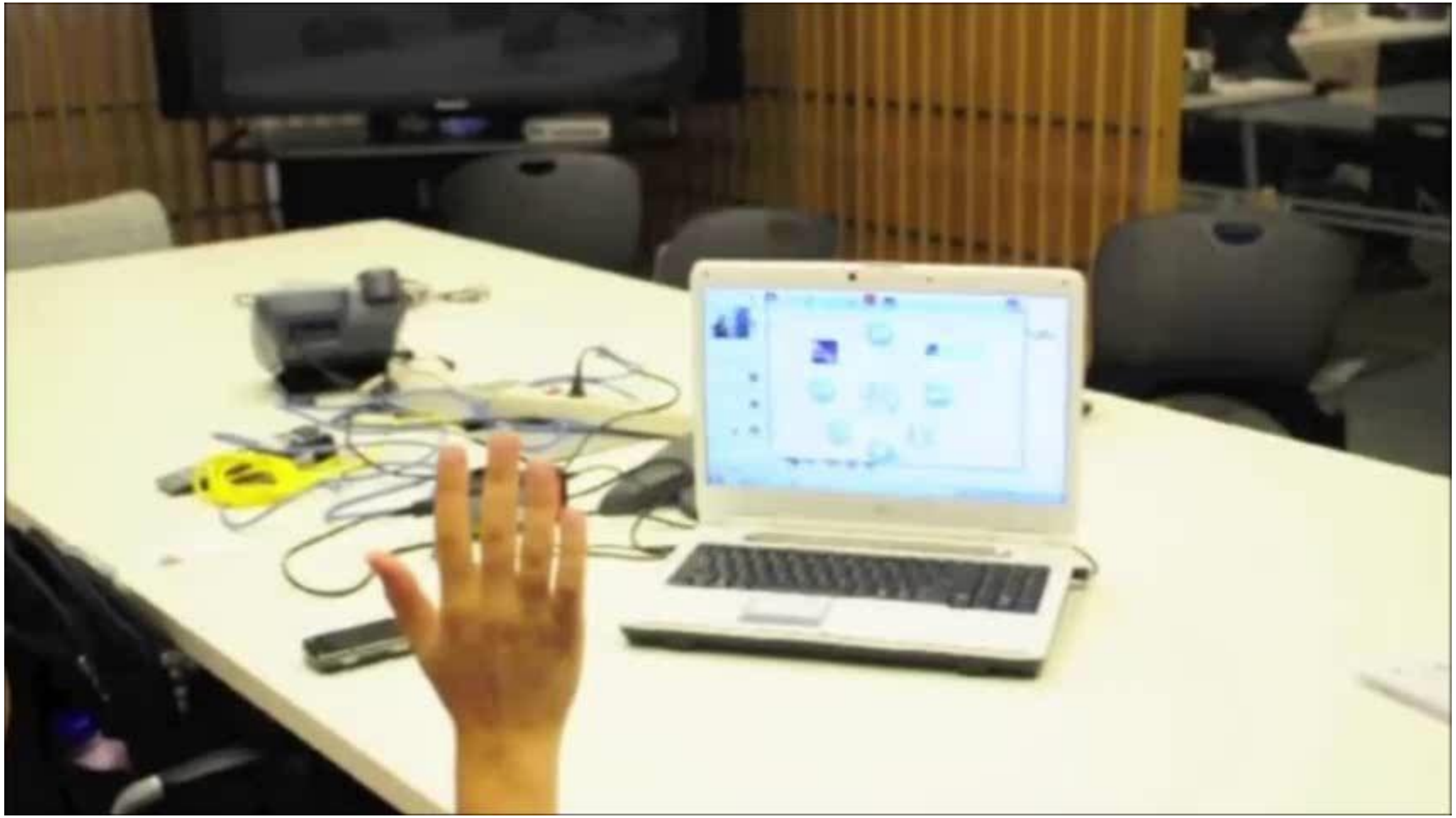


Example: "Picture Taking and Zoom"

- Clockwise Circle: *Zoom Out*
- Counter Clockwise Circle: *Zoom In*
- Up/Down large wave: Take photo

Gesture	Description	Uses
Circle 	**Tracks a hand moving in either a clockwise or counter clockwise circle. One trigger event per circle.**	**Zoom in/out for photos, GPS, and other apps.**
Up/Down Wave 	**Hand must move down and up in a repeating manner. One trigger event after three waves detected.**	**Start photo timer.**

MARKET LAUNCHED DEVICES – FUJITSU LIFEBOOK AH530

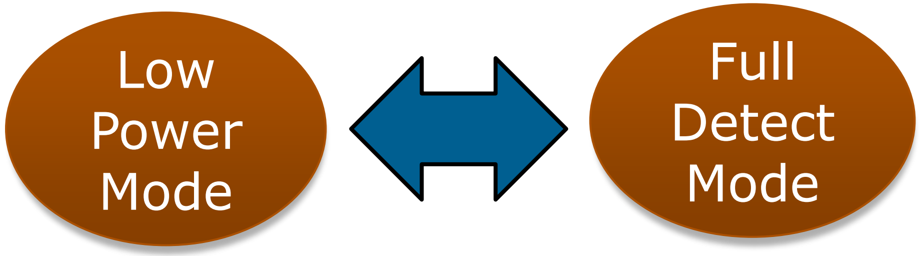


Gesture Application Types – Custom Desktop UI Control

- "Gesture Wheel" style of UI is gesture friendly.
- Provides rapid access to commonly desired controls.
- Can be implemented in an "overlay" that does not interfere with the underlying application.
- Allows user customization and "most recently used" access.

Gesture Application Types – Custom Desktop UI Control

Engagement gesture followed by gesture tracking in custom UI.

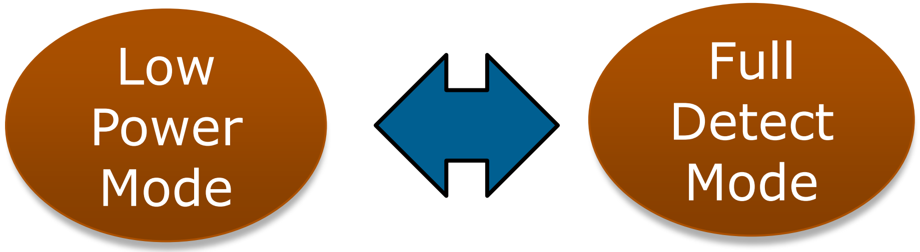


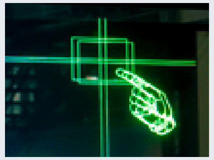
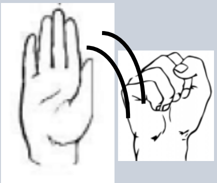
Gesture	Description	Uses
Hand Pose Detection / Engagement 	**Detects a forward facing palm pose in the camera view.**	**Initial engagement of gesture system.**
Progressive Circle 	**Tracks a hand moving in either a clockwise or counter clockwise circle. Sends an event every few degrees around the circle.**	**Volume control, scrolling, and other variable control operations.**
Hand position Tracking 	**Hand position is tracked and provides (x,y) positions.**	**Rapid selection of custom designed "gesture wheel" UI.**

MARKET LAUNCHED DEVICES – SAMSUNG ES Series Televisions

Gesture Application Types – “Always On” Applications

Engagement gesture followed by mouse emulation with click.

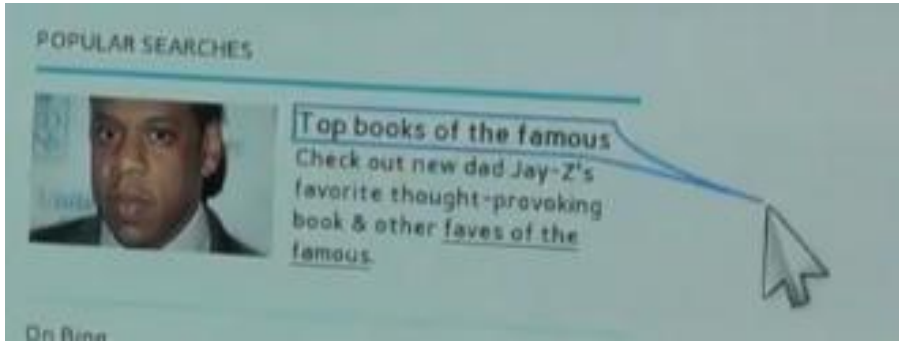


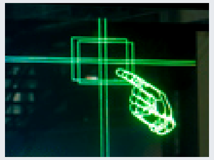
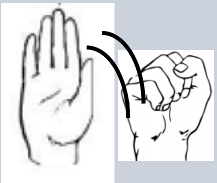
Gesture		Description	Uses
Hand Wave Engagement		**Detects a forward facing palm pose in the camera view.**	**Initial engagement of gesture system.**
Hand position Tracking		**Hand position is tracked and provides (x,y) positions.**	**On-screen mouse cursor control.**
Hand close For “click”		**Hand moves from open palm position to closed fist position to effect a click event.**	**Mouse click emulation.**

MARKET LAUNCHED DEVICES – SAMSUNG ES Series Televisions – Browser Control

Gesture Implementation – TV Browsers

- Voice control combined with gesture for better UX
- Engagement gesture followed by mouse emulation with click for full browser control
- Browser enhanced with "gesture friendly" callouts
- Integration with voice control should be improved to eliminate typing through gestures
- Scrolling should be implemented with gestures, not click

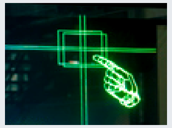
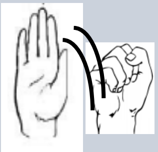
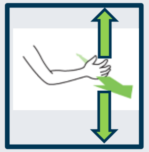
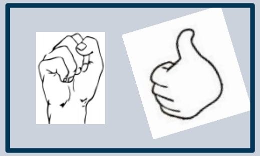


Gesture	Description	Uses
Hand Wave Engagement 	**Detects a forward facing palm pose in the camera view.**	**Initial engagement of gesture system.**
Hand position Tracking 	**Hand position is tracked and provides (x,y) positions.**	**On-screen mouse cursor control.**
Hand close For "click" 	**Hand moves from open palm position to closed fist position to effect a click event.**	**Mouse click emulation.**

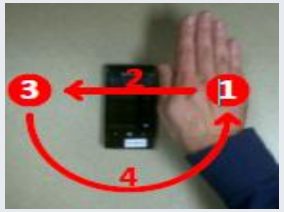
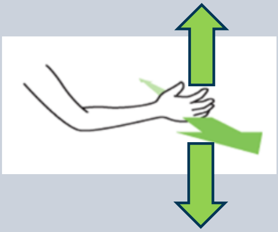
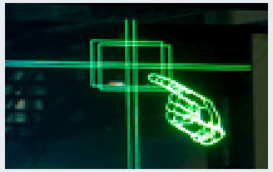
COMING DEVICES – WINDOWS 8 GESTURES

Gesture Implementation – Windows 8 Gestures

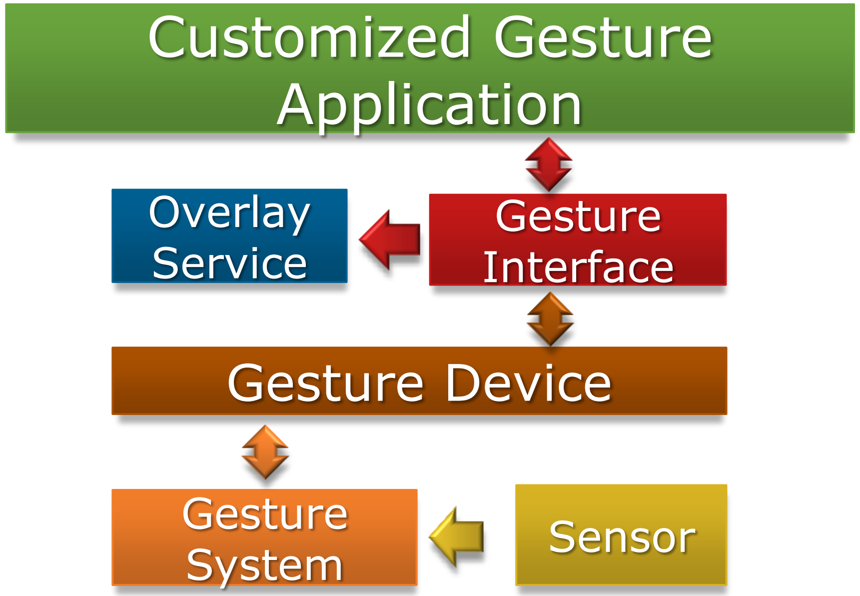
- Engagement gesture followed by mouse emulation with click for full desktop control
- Swipe gestures integrated for gesture UI

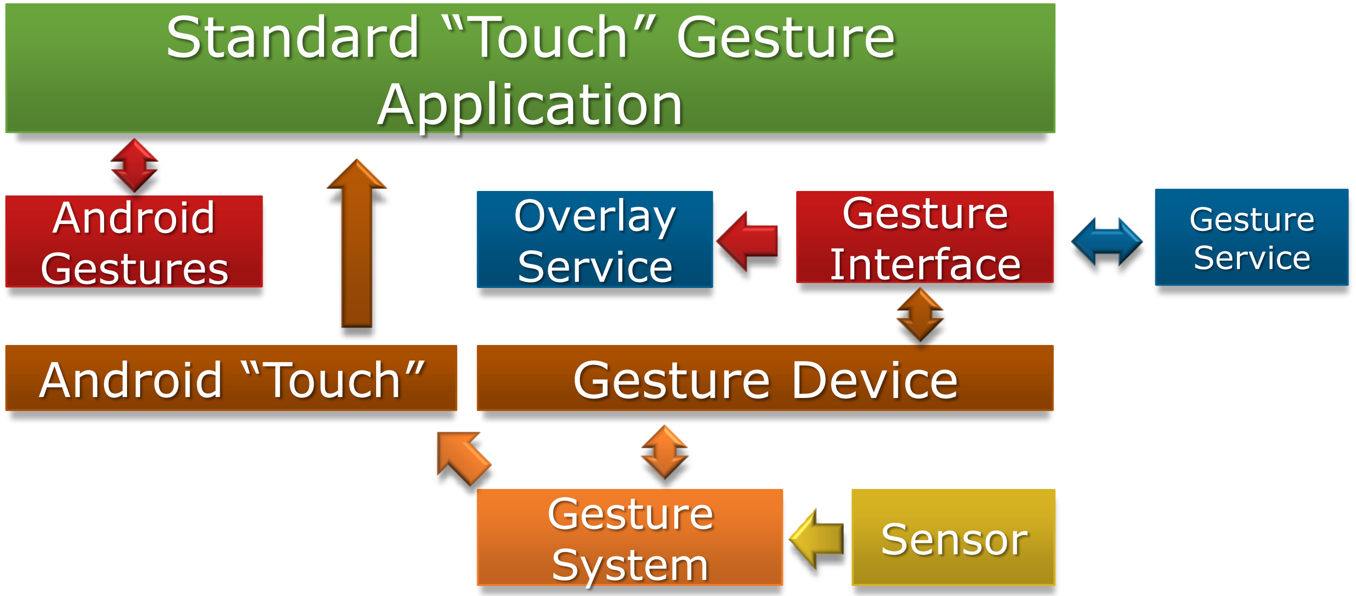
Gesture		Description	Uses
Hand Pose / Wave Engagement / Exit		Detects a forward facing palm pose in the camera view.	Initial engagement of gesture system. Application exit.
Hand position Tracking		Hand position is tracked and provides (x,y) positions.	On-screen mouse cursor control.
Hand Close and Open for "click & drag"		Hand moves from open palm position to closed fist position and is tracked until hand opens again.	Mouse click and drag emulation for more complex user interface tasks.
Far Swipe > 1 foot away		Hand moves rapidly left/right/up or down.	Custom gestural input for "swipe" based interfaces. Page turn, media browser.
Alternative Hand poses		Use thumb up pose and rotate wrist to change thumb angle. Hold hand in "Fist" pose.	Thumb rotation for volume control. Fist pose to exit to Metro UI.

Gesture Feature Set 1.0: Snapdragon ICS Release

Component	Description	Uses
Hand Pose Detection / Engagement  	Detects an open palm hand pose in the camera view. Distinguishes between left and right hands.	Initial engagement of gesture system. Can be used as standalone “ok” gesture.
Near Swipe 	Tracks a hand passing within 6 inches in front of the camera sensor. Sends Swipe Left and Right messages.	Touch-free call answer, in-car gesture mail “no-look” control, page turns, etc.
Far Swipe 	Tracks the gestures which occur after a hand pose detection. Sends Swipe Left, Right, Up and Down messages.	Any swipe style interaction. Allows multiple successive swipes without re-engagement.
Pointer Tracking 	Tracks the hand position after a hand pose detection and outputs hand position as on-screen (x,y) co-ordinates.	“Hand as mouse” tracking for touch screen emulation.

"Direct Connect" vs "Touch Injection" Architecture





"Direct Connection"

- Uses Camera Gesture SDK to connect to the Gesture Device, set the tracking mode of the Gesture System, and receive gesture events

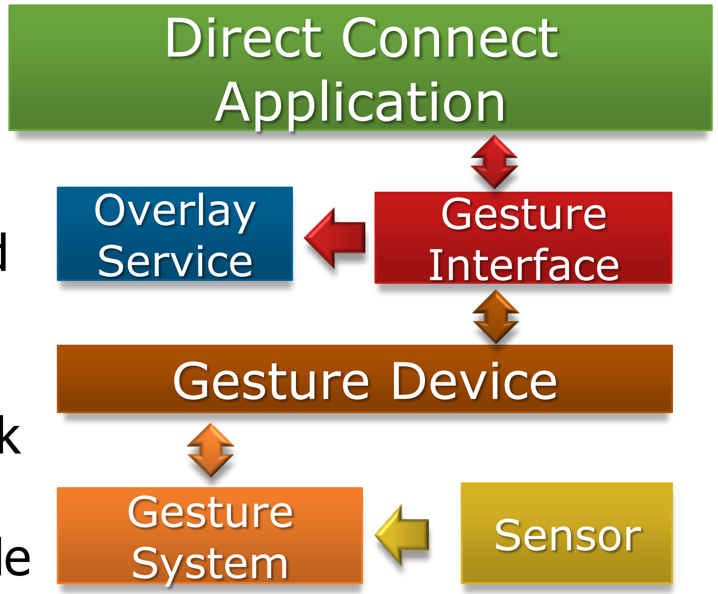
"Touch Injection"

- Allows an existing application to receive gesture events through standard touch system with no change to app.
- Requires Gesture Service to manage the gesture mode through the Gesture Device

Gesture Feature Set 1.0: Snapdragon Gesture Overlay

User Feedback / UX Component

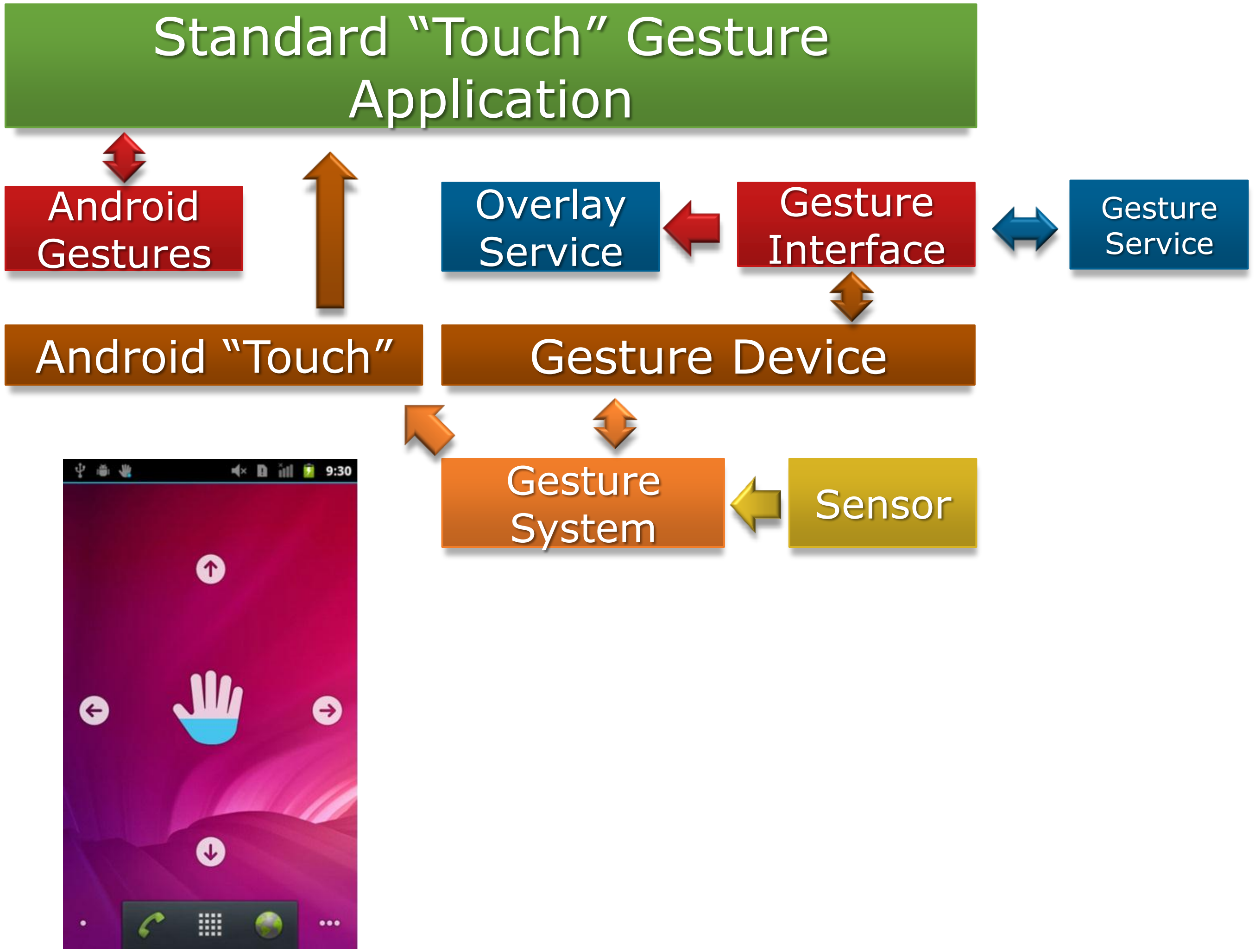
- Is an accessory service that draws overlay imagery in response to:
  - Engagement percentage
  - Swipes
- Provides feedback to the user that their hand is being detected by the system and that they have activated specific gestures
- Service can be replaced by OEMs, and customized to give specific "look and feel" to all gesture UIs
  - Could allow end-user selectable "skins"
- "Direct Connect" applications can disable the overlay and implement their own if desired



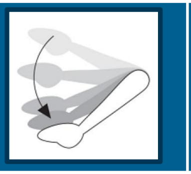
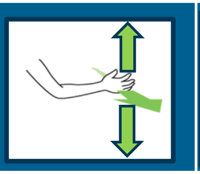
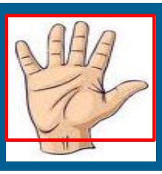
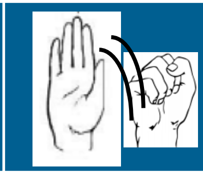
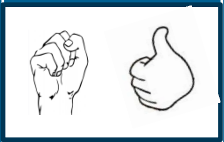
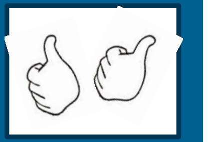
Gesture Feature Set 1.0 – Snapdragon Gesture Service

Allows existing applications to be driven by gestures:

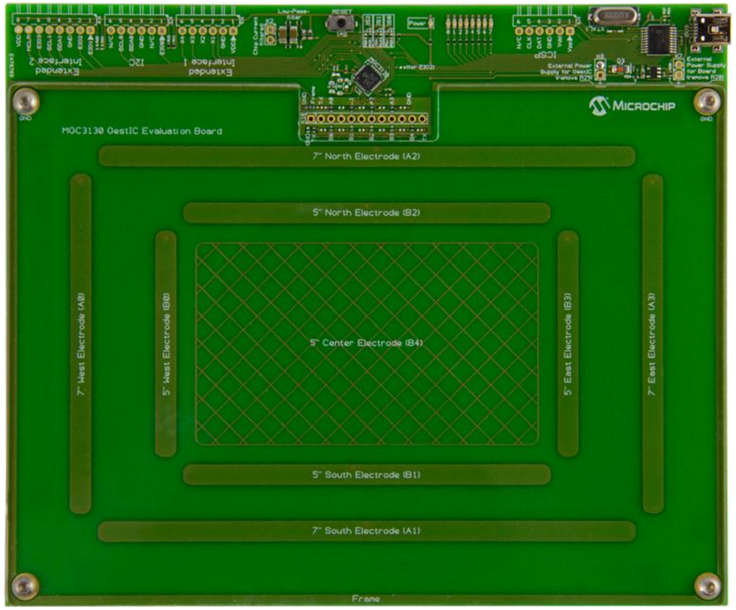
- Existing applications must use the standard "flings" based Android Gestures to be supported
- Service can be configured to support any number of applications, and to enable specific gestures for each
- Service detects when an application comes to the foreground, and controls the gesture system through the Gesture SDK
- All overlay features are supported



Gesture Alphabet Implementation – Comparison

										
	Yes									
		Yes	Yes							
	Yes		Yes		Yes					
					Yes		Yes	Yes		
	?		?	Yes	Yes	Yes	Yes	Yes	Yes	Yes
android	Yes		JB Yes	Yes	JB Yes	Yes	Yes	JB Yes	JB Yes	

GestIC from Microchip – Electrical Near Field Imaging



GestIC® Technology System

The GestIC® technology system is comprised of the respective GestIC® chip which is connected to sensing electrodes to determine the fluctuation pattern of the electrical field (E-field).

Sensing Electrodes

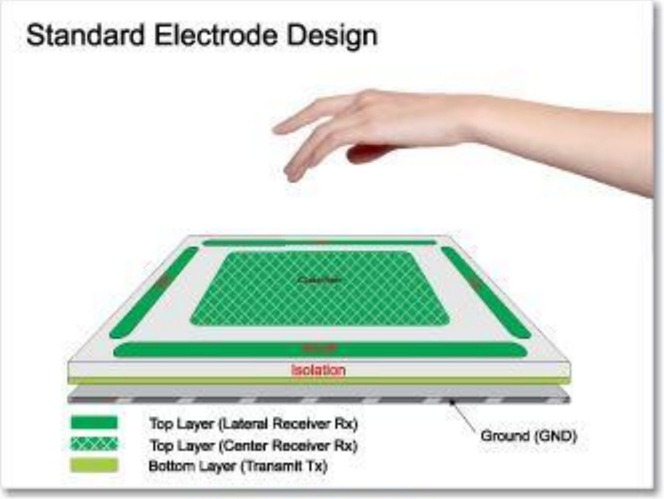
With MGC3130 using a number of typically four receiver (Rx) electrodes, the origin of E-field variations inflicted by the user's hand can be detected.

Those electrode signals are processed by the GestIC® technology device to calculate the hand position relative to the sensing area in x, y, z data points, and classify the movement pattern into gestures in real time.

Electrode Materials

Examples for electrode materials:

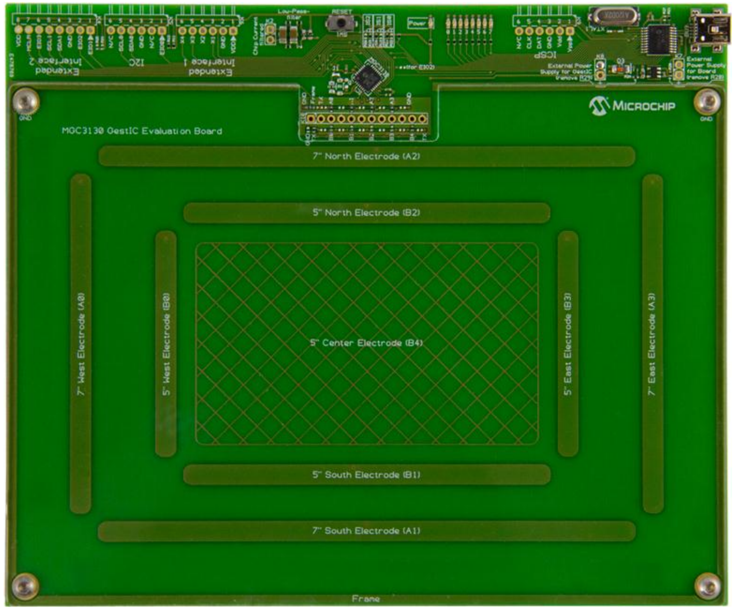
- Rigid printed circuit (PCB) electrodes
- Flexible printed circuit (FPC) electrodes
- Self adhesive electrodes
- Laser direct structured electrodes (LDS)
- Conductive foils
- ITO coating



Click image to enlarge

Microchip's GestIC® technology utilizes low-cost and thin sensing electrodes made of any conductive material. It allows an invisible integration behind the target device's housing while the overall thickness of the product's industrial design is not increased. Even the reuse of existing conductive structures, such as a display's Indium Tin Oxide (ITO) coating, is feasible making GestIC® technololgy a very cost effective system solution.

GestIC from Microchip – Electrical Near Field Imaging



How Does it Work?

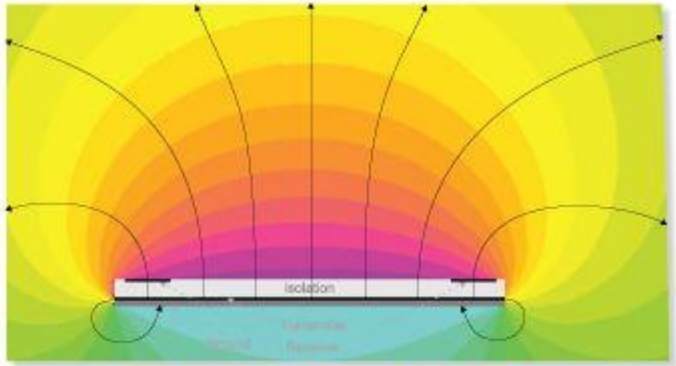


Figure 1: Undisturbed E-field distribution

GestIC® technology uses transmit (Tx) frequencies f in the range of 100 kHz, which reflects a wavelength of about three (3) kilometers. With electrode geometries of typically less than twenty (20) by twenty (20) centimeters, this λ_{Tx} wavelength is much larger in comparison.

Therefore, the magnetic component is practically zero and no wave propagation takes place. The result is a quasi-static electrical near field that can be used for sensing conductive objects such as the human body.

Once a user intrudes the sensing area, the electrical field distribution becomes distorted. The field lines intercepted by the hand are shunted to ground through the conductivity of the human body itself. The simulation results in Figure 1 and Figure 2 show the influence of an earth grounded body to the electric field.

As illustrated, the proximity of the body causes a compression of the equipotential lines and shifts the receiver (Rx) electrode signal levels to a lower potential which is detected by the respective GestIC® technology product.

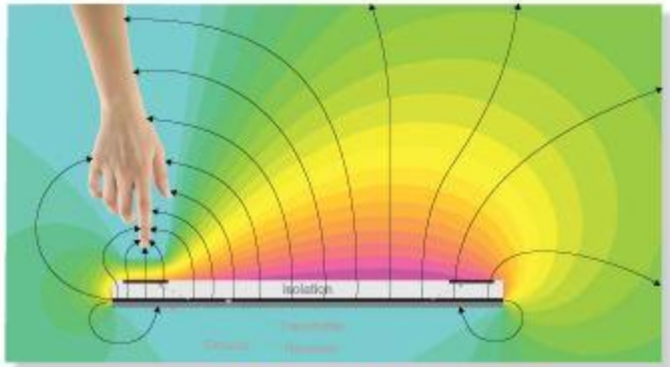
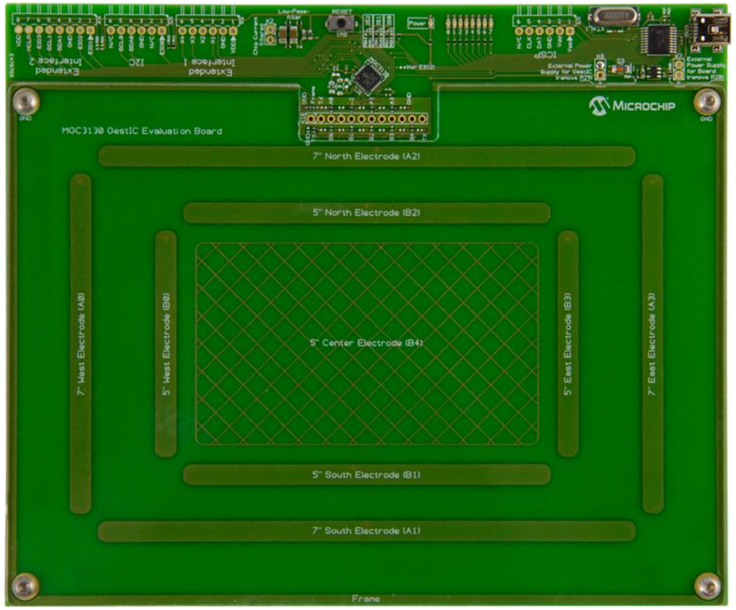


Figure 2: E-field distorted by human hand

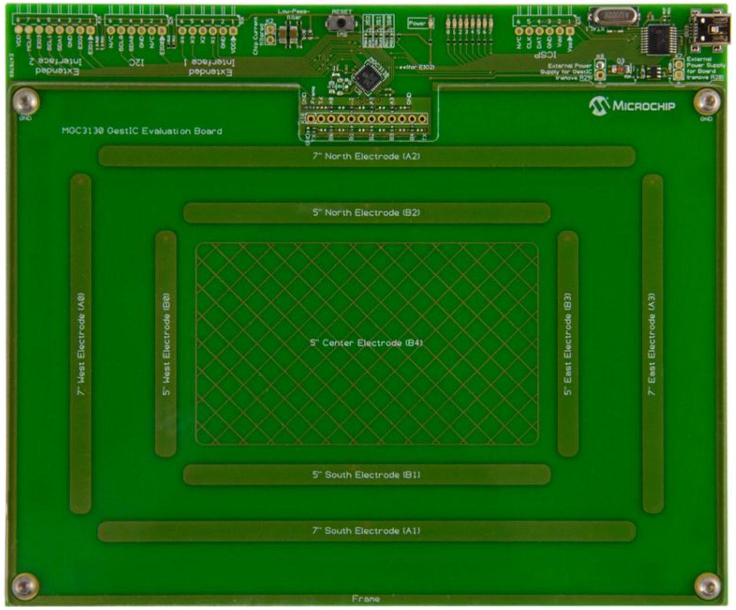
GestIC from Microchip – Electrical Near Field Imaging



MICROCHIP

General Operation Demo

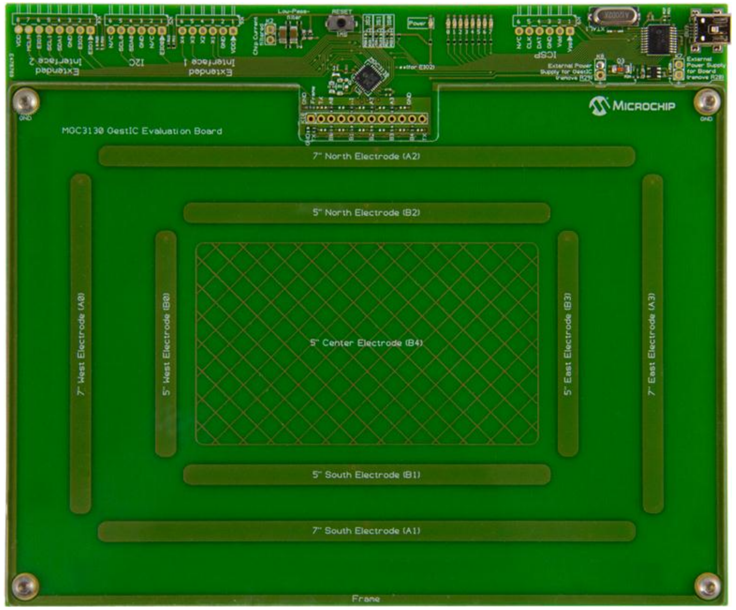
GestIC from Microchip – Electrical Near Field Imaging



MICROCHIP

Gesture Demo

GestIC from Microchip – Electrical Near Field Imaging



Low Power Demo

Alternative Low Power Solutions: Ultrasound

Close to the display: Ultrasound Tracking

- Uses multiple microphones plus emitted sound to determine the hand position when close to the device.
- Can also be used as engagement technique for other tracking mechanisms
- ***<u>Low Cost, Low Power</u>***
- Also enables high fidelity "active" stylus tracking, both onscreen and off.

Elliptic Labs – Ultrasound

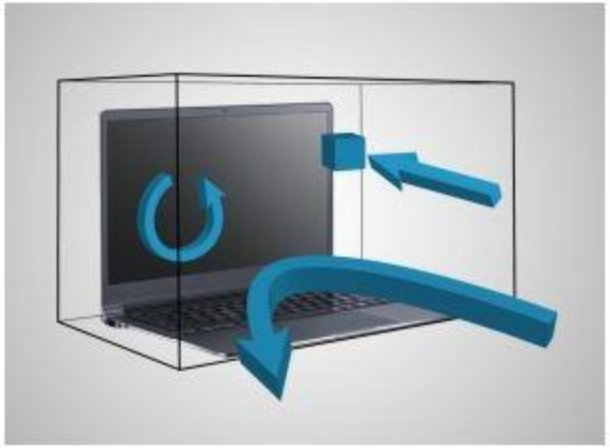
Wide Field of View

Sensors can see both forward and to the sides, enabling an interaction zone extending over the screen and to the sides. Ultrasonic sensors enables gesturing both from a distance and very close to the screen at the same time

ellipticlabs

Distributed Sensing

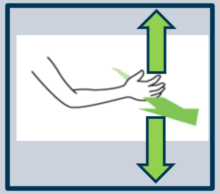
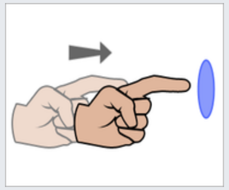
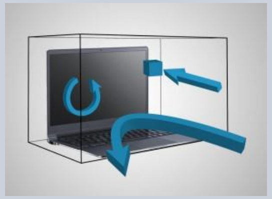
Enables motion capture from multiple angles, avoiding the occlusion of objects or parts of an object. Facilitates robust positioning by giving sufficient sensor baselines. Sensors used are MEMS microphones which can also be used for speech enhancement and recognition.



Elliptic Labs – Ultrasound

ellipticlabs

Elliptic Labs – Ultrasound

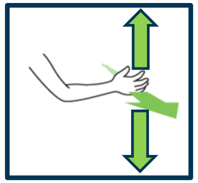
Gesture	Description	Uses
Near / Mid Swipe 	**Hand moves rapidly left/right/up or down.**	**Stepping through images, scrolling browser, etc.**
"Poke" for Click 	**Hand can approach the screen rapidly and activate a click anywhere on the screen.**	**Selection of onscreen objects using a "poke" click. Note that no mouse tracking was shown.**
Edge of screen swipes 	**Hand moves in offscreen area to onscreen from various directions to invoke special controls.**	**Novel selection of various activities including application swapping, menus, etc.**

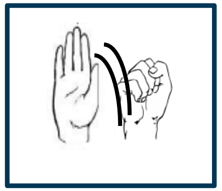
Mouse Emulation versus Gesture Events

- Android OS
  - Supports gesture events natively (“Flings”)
  - Many applications available
- Windows 8:
  - Supports gesture events natively
  - Navigation and other apps starting to adopt
  - Substitute control possible using keyboard events
- Mouse emulation:
  - Can be slower than simply touching the screen
  - Requires greater user concentration
- Gesture events:
  - Can be faster than touch
  - Less concentration required (move in the general area)
  - Provides the “most common actions” very quickly

Moving Forward…

- Keep it simple
  - Apple's "pinch to zoom" was enough to enable a myriad of use cases
  - Use a minimal, robust set of gestures (don't rotate the thumb just because you can!)
  - Be consistent in the use of specific gestures
- UX
  - Provide dynamic feedback to the user at all times
  - For best experience, design custom apps that leverage the gestures
- Backwards application support
  - Using a mouse is ok but augment with gestures to improve experience over touch
  - "Typing in the air" should not be the goal – integrate voice and other techniques
- Investigate low power "always on" alternatives





Thank You

QUALCOMM Incorporated, 5775 Morehouse Drive, San Diego, CA 92121-1714