

DISEÑO DE SISTEMA DE INTERACCION HUMANO-MÁQUINA MULTIENTRADA
PARA FACILITAR EL CONTROL DE SOFTWARE EN COMPUTADOR
PERSONAL



Juan Camilo Estela Villegas

UNIVERSIDAD AUTONOMA DE OCCIDENTE
FACULTAD DE INGENIERIA
PROGRAMA DE INGENIERIA MECATRONICA
SANTIAGO DE CALI
2009

DISEÑO DE SISTEMA DE INTERACCION HUMANO-MÁQUINA MULTIENTRADA
PARA FACILITAR EL CONTROL DE SOFTWARE EN COMPUTADOR
PERSONAL



Juan Camilo Estela Villegas Cód. 2061468

Proyecto de Grado para optar por el título de

Ingeniero Mecatrónico

Modalidad: Proyecto de Grado

Director

Drago Dussich Muñoz

Ingeniero Mecatrónico

UNIVERSIDAD AUTONOMA DE OCCIDENTE
FACULTAD DE INGENIERIA
PROGRAMA DE INGENIERIA MECATRONICA
SANTIAGO DE CALI
2009

Nota de aceptación:

Aprobado por el Comité de Grado en cumplimiento de los requisitos exigidos por la Universidad Autónoma de Occidente para optar al título de Ingeniero Mecatronico.

CARLOS HECTOR CRUZ VERGARA
Jurado

ANDRES FELIPE NAVAS ESCOBAR
Jurado

Santiago de Cali, 04 de Diciembre de 2009

CONTENIDO

| | Pág. |
|---|------|
| GLOSARIO | 9 |
| RESUMEN | 10 |
| INTRODUCCIÓN | 11 |
| 1.PROBLEMÁTICA | 12 |
| 2.ANTECEDENTES | 16 |
| 3.MARCO TEORICO | 17 |
| 3.1 SUPERFICIE TÁCTIL O PANTALLA TÁCTIL | 17 |
| 3.2 ENTRADA DISCRETA O CONTINUA | 18 |
| 3.3 GRADOS DE LIBERTAD | 18 |
| 3.4 TAMAÑO Y ORIENTACIÓN | 18 |
| 3.5 VARIABLES MEDIDAS | 19 |
| 3.6 MÚLTIPLES ESTÍMULOS | 19 |
| 3.7 INFORMACIÓN RECOLECTADA | 19 |
| 4.OBJETIVOS | 20 |
| 4.1 OBJETIVO GENERAL | 20 |
| 4.2 OBJETIVOS ESPECÍFICOS | 20 |
| 5.EL SISTEMA | 21 |
| 5.1 SENSOR | 21 |
| 5.1.1 ÓPTICOS | 21 |
| 5.1.2 PRESIÓN | 23 |
| 5.1.3 DISCRETO | 23 |
| 5.1.4 CAPACITIVO | 24 |
| 5.1.5 RFID | 24 |
| 5.2 FILTRADO | 24 |
| 5.3 PROCESAMIENTO | 25 |
| 5.4 APLICACIÓN | 28 |
| 5.5 DISPLAY | 29 |
| 6. BENCHMARK | 30 |
| 7.CONCEPTOS | 32 |
| 8.DISEÑO DEL SISTEMA | 35 |
| 8.1 SENSOR | 36 |
| 8.1.1 Resistencia Variable a la presión | 36 |
| 8.1.2 Tinta Resistiva | 39 |

| | |
|-------------------------------|----|
| 8.1.3 Celda de Presión | 42 |
| 8.1.4 Construcción del Sensor | 45 |
| 8.2 INTERFAZ | 53 |
| 8.3 PROCESAMIENTO | 57 |
| 8.3.1 Calibración | 57 |
| 8.3.2 Interpolación | 59 |
| 8.3.3 Extracción de Regiones | 60 |
| 8.3.4 Sincronización | 61 |
| 8.3.5 Aplicación | 63 |
| 9 RESULTADOS | 66 |
| 10 APLICACIONES | 69 |
| 11 CONCLUSIONES | 71 |
| BIBLIOGRAFIA | 74 |

LISTA DE TABLAS

| | Pág. |
|--|-------------|
| Tabla 1. Comparación sistemas actuales v.s. Multitoque | 15 |
| Tabla 2. Comparación eventos actuales v.s. Multitoque | 27 |
| Tabla 3. Tiempos de rotación, traslación y escalado (Básicos) | 30 |
| Tabla 4. Tiempos de rotación, traslación y escalado (Promedio) | 31 |
| Tabla 5. Tiempos de rotación, traslación y escalado (Avanzados) | 31 |
| Tabla 6. Conceptos de Diseño | 32 |
| Tabla 7. Resultado mezclas de compuesto resistivo | 40 |
| Tabla 8. Resistencia por Milímetro de cada mezcla | 41 |
| Tabla 9. Peso aplicado v.s. Resistencia en RVP | 42 |
| Tabla 10. Tiempos de rotación, traslación y escalado simulados (Básicos) | 67 |
| Tabla 11. Tiempos de rotación, traslación y escalado simulados (Promedio) | 67 |
| Tabla 12. Tiempos de rotación, traslación y escalado simulados (Avanzados) | 67 |

LISTA DE FIGURAS

| | Pág. |
|--|-------------|
| Figura 1. Grados de libertad de los sistemas actuales | 12 |
| Figura 2. Secuencia de escalado, rotación y traslación de una imagen con métodos actuales. | 14 |
| Figura 3. Diagrama Subsistemas | 21 |
| Figura 4. Diagrama de proporciones en la mezcla conductiva | 37 |
| Figura 5. Diagrama comportamiento bajo compresión de la mezcla | 38 |
| Figura 6. Efectos por falta de homogeneidad en la mezcla | 41 |
| Figura 7. Circuito de lectura de la celda de presión | 44 |
| Figura 8. Comportamiento del circuito de lectura | 44 |
| Figura 9. Sensor con matriz de electrodos | 45 |
| Figura 10. Esquema eléctrico de sensor discreto | 46 |
| Figura 11. Sensor con capa de resistencia variable a la presión | 46 |
| Figura 12. Esquema eléctrico de intersección con RVP | 47 |
| Figura 13. Sensor con RVP y tinta conductiva | 48 |
| Figura 14. Diagrama sin toque | 49 |
| Figura 15. Diagrama con toque | 50 |
| Figura 16. Esquema eléctrico intersección con toque | 51 |
| Figura 17. Diagrama de posiciones interpoladas | 53 |
| Figura 18. Diagrama micro controlador propuesto | 54 |
| Figura 19. Código de lectura del sensor | 56 |
| Figura 20. Diagrama de flujo del algoritmo de calibración | 58 |
| Figura 21. Resultado interpolación | 59 |
| Figura 22. Resultado etiquetado | 60 |
| Figura 23. Proceso de identificación de regiones | 61 |
| Figura 24. Flujo de información desde el circuito hasta la aplicación | 63 |
| Figura 25. Listado de gestos multitoque | 65 |
| Figura 26. Prototipo para prueba de concepto | 71 |
| Figura 27. Resultados obtenidos con el prototipo | 72 |

LISTA DE ANEXOS

| | Pág. |
|--|-------------|
| Anexo A. Código implementado en C# para identificación de regiones | 75 |
| Anexo B. Código implementado en C# para la calibración del sensor | 76 |

GLOSARIO

CELDA DE PRESIÓN: unidad básica para la medición de presión

CELDA TÁCTIL: unidad básica requerida para detectar la posición de un toque

GESTO: acción física efectuada por el usuario que corresponde a una acción en el sistema computacional.

MULTITOQUE: interacción con múltiples toques o puntos de contacto.

RVP: resistencia Variable a la Presión

TOQUE: punto en el cual la herramienta de interacción con la maquina hace contacto intencional con la superficie del sensor.

RESUMEN

Cada día la tecnología avanza un paso más para lograr máquinas más poderosas y rápidas. Sin embargo la tecnología en este momento es capaz de lograr tareas a velocidades sorprendentes pero el ingreso de información es realizado por personas. Este ingreso de información está limitado por la cantidad de información que la persona puede entregarle al sistema con base en el dispositivo de comunicación. En este trabajo se plantea un sistema que permita entregarle más información a un computador personal de manera más rápida, natural y eficiente para lograr una mejor interacción humano-máquina y lograr explotar la capacidad de los sistemas actuales de cómputo.

INTRODUCCION

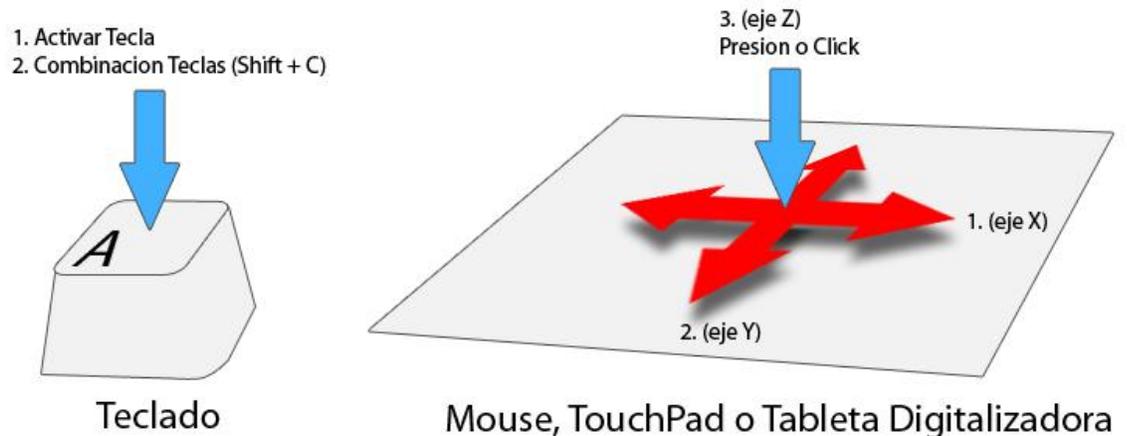
Los avances en tecnología permiten cada día tener máquinas más veloces y eficientes, sin embargo hay casos en los que la velocidad de desempeño de la máquina se ve desperdiciada pues el usuario no puede entregarle información suficientemente rápido al sistema. Para ilustrar el caso, es fácil ver como programas de edición de texto o de imágenes corriendo en un computador personal de hoy en día pueden procesar miles de palabras o cientos de fotos por minuto. Pero el usuario está limitado a ingresar la información al sistema por medio de un mouse o un teclado, lo cual reduce la cantidad de información que se le entrega a este por unidad de tiempo generando una subutilización del equipo y una ejecución más lenta de las tareas requeridas por el usuario. Este problema se ha tratado previamente con la implementación de pantallas VGA, teclado, mouse, joystick y otros dispositivos de interacción humano – máquina. Sólo es hasta hoy que se tiene el poder computacional para aplicar viablemente otros tipos de interacción entre humanos y máquinas para poder hacer que el uso de éstas sea más rápido, eficiente e intuitivo. En este caso se diseñará una interfaz humano-máquina compuesta de software y hardware que permita lograr una interacción más rápida y eficiente al permitir al usuario entregarle más información al sistema con menos acciones.

1. PROBLEMÁTICA

Para realizar una tarea sencilla con los dispositivos de interacción actuales se requieren demasiados y repetitivos pasos, no intuitivos para el usuario no capacitado. Se diseñará un sistema que permita introducir más información al sistema con menos pasos, reducir el estrés causado por movimientos repetitivos no naturales y reducir el tiempo de capacitación para utilizar dicho sistema.

Los sistemas comerciales de interacción humano-máquina disponibles hoy en día solo permiten un máximo de 3 grados de libertad lo cual limita la cantidad de información que recibe el computador por cada movimiento del usuario. Esto implica que el usuario debe realizar más movimientos, restringidos a los grados de libertad disponibles que en la mayoría de los casos no son intuitivos, para lograr una tarea que en el mundo físico se puede realizar sin capacitación previa y de manera instantánea sobre la superficie de trabajo. Como ejemplo de uno de los casos en los que el aumento de grados de libertad hará más eficiente la interacción humano máquina es al girar y/o escalar una imagen. En los sistemas comúnmente disponibles podemos efectuar estas acciones por medio del teclado, de un mouse, de una tarjeta digitalizadora o de un touchscreen.

Figura 1. – Grados de libertad en métodos actuales de interacción humano-máquina.



Teclado: 2 grados de libertad. Mouse: 3 grados de libertad, uno de estos binario.

Una tarea sencilla realizable en un computador personal es el giro y escalado de una imagen. Esta tarea se utilizará como referencia para el planteamiento y

valoración del desempeño del sistema. En el caso de realizar un giro y/o desplazamiento de una imagen con el teclado actualmente se puede lograr de la manera más eficiente así:

- Tecla Flecha hacia Arriba: Desplazamiento en incrementos predefinidos sobre el eje y.
- Tecla Flecha hacia la Derecha: Desplazamiento en incrementos predefinidos sobre el eje x.
- Tecla Flecha hacia Abajo: Desplazamiento en decrementos predefinidos sobre el eje y.
- Tecla Flecha hacia la Izquierda: Desplazamiento en decrementos predefinidos sobre el eje x.
- Tecla +: Giro sobre el centro geométrico en incrementos predefinidos.
- Tecla -: Giro sobre el centro geométrico en decrementos predefinidos.

Con esta metodología se puede ver claramente que la cantidad de veces que se deben presionar las teclas en secuencia depende de la magnitud de los valores deseados y el tiempo en que se tarda en ejecutar una transformación en una única imagen en pantalla tarda más de lo que tarda rotar y trasladar en la misma magnitud una hoja de papel en el mundo físico.

En el caso de realizar un giro y/o desplazamiento de una imagen con el mouse, tableta digitalizadora o superficie táctil hasta el momento de manera más eficiente se puede lograr así:

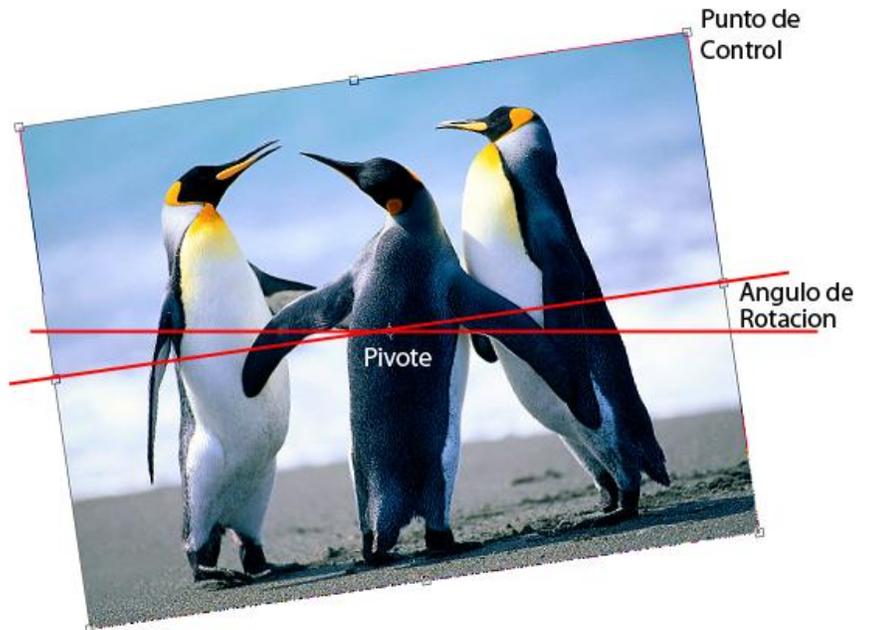
- Llevar mano hasta el mouse
- Llevar puntero hasta la imagen
- Seleccionar imagen
- Arrastrar imagen hasta ubicación deseada
- Llevar puntero hasta el pivote
- Arrastrar pivote hasta el centro de giro deseado
- Llevar puntero al borde de la imagen
- Arrastrar el borde de la imagen hasta lograr el ángulo de giro deseado

Con esta metodología se puede ver fácilmente que la transformación de la imagen en pantalla requiere de varios desplazamientos de un único puntero sobre una superficie en 2 dimensiones. Esto hace que ejecutar la transformación en el mundo digital sea más engorroso y lento que en el mundo real.

Figura 2. – Secuencia de escalado, rotación y traslación de una imagen con métodos actuales.

Proceso escalado, rotación y traslación de una imagen con metodos de interaccion actuales

1. Llevar mano hasta el mouse.
2. Llevar puntero hasta la imagen a procesar.
3. Seleccionar imagen.
4. Llevar puntero hasta el pivote de rotación.
5. Llevar pivote hasta la posición deseada.
6. Llevar puntero a un punto de control de la imagen(Normalmente una esquina de la imagen).
7. Arrastrar punto de control para definir escala y rotación.
8. Llevar puntero sobre la imagen.
9. Arrastrar imagen hasta posición deseada.
10. Aceptar transformación.



El problema radica en la relación entre la cantidad de información que se le entrega al sistema por unidad de tiempo y cantidad de acciones. Un sistema que permita entregar más información al sistema en menos pasos permitirá ejecutar acciones dentro del computador mucho más rápido. La entrega de información al sistema es intuitiva, más semejante a la forma en que las personas manipulan objetos físicos para así no solo reducir el tiempo de uso por tarea pero también el tiempo requerido para la capacitación sobre el uso de este dispositivo.

Tabla 1. Comparación entre sistemas actuales y multientrada

| | Mouse/TouchPad/TouchScreen | Sistema Multientrada |
|--|---|--|
| Grados de libertad | 3 (Max) | 3 o mas |
| Información entregada al computador | Coordenadas X,Y y clic para un solo mouse. 2 variables continuas y una binaria. | Coordenadas X,Y, y Tamaño para cada toque. 3 variables continuas por cada toque. |
| Objeto desplazado por el usuario | Mouse | Sus propias manos / Stylus / Marcador / etc. |

Estas acciones con las interfaces actuales como teclado y mouse permiten este nivel máximo de rendimiento y solo para un usuario a la vez. En el ambiente de trabajo que cambia y avanza al ritmo de la tecnología el mantener equipos de trabajo aislados en cada estación de trabajo entorpece el proceso de colaboración. Casos tan simples como el de trabajo en equipo en una sala de juntas implica recolectar el material para posteriormente ejecutar lo discutido en estaciones personales de trabajo. Si los sistemas de interacción con los computadores permitieran suficientes grados de libertad como para que no sólo una persona sino varias pudieran colaborar sobre una misma aplicación, se generaría un mejor y más eficiente flujo y manejo de la información. Esto representa reducción de tiempos desperdiciados en tareas lentas y repetitivas, reducción en tiempos de capacitación y reducción de lesiones por estrés repetitivo al tener que limitar los movimientos del usuario a los grados de libertad que restringe el dispositivo de entrada. Con base en base a esto podemos decir que los sistemas de interacción humano-máquina actuales tienen las siguientes características a mejorar:

- Tiempo de entrega de información e instrucciones del usuario al computador.
- Estrés por movimientos repetitivos y no naturales en el usuario.
- Tiempos de capacitación al hacer que la usabilidad sea más intuitiva.
- Cantidad de usuarios que pueden utilizar el dispositivo a la vez.
- Cantidad de elementos requeridos para poder utilizar la interfaz. (ej. Mouse, stylus, guante, etc.)

2. ANTECEDENTES

A nivel internacional se ha estimulado el desarrollo en este campo desde el año 1981 en el cual Jack Rebman de Lord Corporation patentó un arreglo de sensores táctiles para que un robot pudiera sentir posición, forma, orientación, etc. del objeto analizado.

En 1982 Nimish Mehta de la Universidad de Toronto es la primera persona en pasarle información multitoque a un computador. En ese momento una cámara filmaba la parte posterior de un vidrio opalizado y detectaba los puntos donde los dedos hacían contacto con el vidrio.

Para el año 1984 Bell Labs desarrolla la primera pantalla CRT con sensores capacitivos para detectar más de un toque en su superficie.

En 1985, el año en que el primer computador Mac sale al mercado ya el Input Research Group de la Universidad de Toronto había desarrollado un panel capacitivo para detección de múltiples toques en 3D. Carnegie Mellon University desarrolla un sistema similar pero con sensores ópticos y una matriz de luz laser. Esta se ve afectada al reconocer múltiples toques por el fenómeno de oclusión.

En 1991 Xerox construye un sistema multitoque basado en eco localización proyectado sobre una mesa. Permite interacción multitoque para escalar y desplazar documentos.

En los próximos años surge Active Desk (1997), Tactex controls (1998), Portfolio Wall (1999), Diamond touch (2001), Universidad de Toronto (2003), Jazz Mutant (2003), Touchlight (2004), perceptive pixel (2006), Synaptics (2006), iPhone (2007) y Microsoft Surface (2007). Todos estos han implementado la tecnología multitoque desde varios enfoques pero solo el iPhone ha sido la aplicación de esta tecnología que ha llegado al consumidor promedio. Las demás implementaciones no han sido comercializadas aun por requerimientos de poder de procesamiento, costo y ambientes de desarrollo de software no estandarizados.

Recientemente se ha generado una iniciativa de código libre que pretende lograr los mismos resultados en comunidades de "hágalo usted mismo". Entre las más reconocidas esta nuigroup, comunidad que trabaja sobre el portal nuigroup.com en el cual se comparten distintos métodos para realizar este tipo de superficies por aficionados a nivel mundial. Sin embargo estos permanecen como integraciones de subsistemas no viables para uso comercial pues no cumplen con una velocidad de respuesta o precisión adecuada.

Se han utilizado distintos métodos para la adquisición de información multitoque en todo estos sistemas. Paneles resistivos permiten captar 2 toques en superficies planas. Paneles capacitivos permiten capturar más de 2 toques. Pero el método que resulta más viable hasta la fecha es el de capturar la información por medios ópticos. Esto se logra por medio de cámaras trabajando usualmente en el espectro infrarrojo con técnicas de iluminación difusa o FTIR. La iluminación difusa consta en iluminar desde atrás de la superficie con luz infrarroja, de esta forma los objetos que están cerca a la superficie serán iluminados y serán captados por la cámara. La iluminación FTIR (Frustrated Total Internal Reflection) es un método en el cual se aprovecha el fenómeno de reflexión interna de los cristales. La luz es proyectada dentro del panel de cristal. Al ver a través del vidrio no se ve la luz, hasta que un objeto con índice de refracción distinto al del aire entra en contacto con el cristal. Este interrumpe la reflexión interna reflejando la luz infrarroja en el sentido de la normal del plano de cristal. Esto hace que la cámara vea solo los puntos donde hay contacto con la superficie.

3. MARCO TEORICO

Dada la gran cantidad de aplicaciones y métodos para lograr un tipo de superficie que capture más información de entrada se clasificará el marco de referencia con base en las distintas variables manejadas para lograr el objetivo. Entre los dispositivos de interacción con computadores personales se pueden clasificar o analizar bajo las siguientes características:

3.1 SUPERFICIE TÁCTIL O PANTALLA TÁCTIL

El sensor consta de cualquier superficie, plana o curva que reciba información para interactuar con el computador. La pantalla es la misma superficie pero existe una continuación entre la superficie táctil y lo que el usuario esta visualizando. Ejemplo de esto es el touchpad en computadores portátiles v.s. la pantalla táctil en tabletPCs. El touchpad permite interactuar pero la información la recibe el usuario en la pantalla ubicada en otra posición distinta a la superficie. El tabletPC tiene la superficie táctil sobre la pantalla permitiendo que el usuario tenga una experiencia más intuitiva.

3.2 ENTRADA DISCRETA O CONTINUA

Los teclados son ejemplos de entradas discretas, a pesar de permitir combinaciones de teclas simultáneas siguen siendo sistemas discretos. El mouse o superficie táctil puede actuar como sistema discreto (clic) o continuo (arrastrar).

3.3 GRADOS DE LIBERTAD

El mouse o touchpad tienen dos grados de libertad al permitir desplazamiento sobre el eje X y Y. Algunos sistemas como tabletas digitalizadoras o paneles con sensores capacitivos permiten sentir la posición en Z/presión ejercida. Sin embargo esto implica que el computador sigue actuando con base en la información de un único puntero. Al permitir que la superficie sienta múltiples puntos de estimulación cada punto permite al menos 2 grados de libertad. Es decir que al utilizar 2 dedos o 2 stylus en una misma superficie se tienen 4 grados de libertad. Este aumento en grados de libertad permite tener una experiencia mucho más eficiente y semejante a la del mundo real.

3.4 TAMAÑO Y ORIENTACIÓN

El tamaño de la superficie se ve afectado por factores de viabilidad de construcción, instalación y configuración. En el caso de trabajar con sensores capacitivos o resistivos el sensor debe ser del tamaño de la superficie y no permite mayor curvatura en esta. Al usar sensores ópticos perpendiculares a la superficie se puede escalar más fácil y se puede tener curvaturas en la superficie. El tamaño de la superficie también es condicional para la resolución y el poder de procesamiento requerido para lograrlo. En caso de utilizar los dedos la resolución será mucho más baja que si se espera trabajar con la punta de un stylus. La orientación de la superficie también presenta retos para lograr una interacción fluida. En caso de tener superficies horizontales se debe considerar los puntos de apoyo de la geometría humana, es decir que al escribir sobre la superficie la palma de la mano debe ser ignorada y solo considerar la punta de la herramienta (dedo o stylus).

3.5 VARIABLES MEDIDAS

Dependiendo del sensor empleado se puede medir distintos tipos de variables para cada punto de estímulo tales como posición, tamaño, orientación, presión.

3.6 MÚLTIPLES ESTÍMULOS

Cada estímulo puede ser generado por una herramienta distinta. Puede ser la punta de un dedo, la punta de un stylus, un objeto reconocible por el sistema. A su vez cada herramienta puede generar un gesto o movimiento específico para ejecutar una acción. Cada punto de contacto puede ser reconocido como individual o puede ser parte de una mano/usuario. Con base en este tipo de reconocimiento avanzado es posible crear sistemas multiusuario. Este tipo de reconocimiento y clasificación de estímulos se ve claramente definido en el sistema multitoque del iPhone de Apple Computers. Este dispositivo por ser reducido en tamaño y poder de procesamiento ignora el 3er y siguientes puntos de contacto pues el cálculo se reduce al procesar el área de un rectángulo mas no las coordenadas de cada punto. Esto se puede lograr sin afectar la experiencia pues la superficie es tan pequeña que físicamente hace incomodo utilizar más puntos de contacto y se asume que solo es una persona la que utiliza el dispositivo. La pared interactiva de Jeff Han sí recibe múltiples toques de múltiples usuarios lo cual hace que el sistema sirva para colaboración simultanea con varios usuarios.

3.7 INFORMACIÓN RECOLECTADA

En todos los casos de interacción multientrada en superficies la información mínima requerida es una matriz de 2 dimensiones con valores correspondientes al estímulo. Dada la naturaleza de la estructura de datos recibida del sensor se debe filtrar la señal y procesar como una imagen, por lo cual el procesamiento de imágenes digitales se aplica en este caso para reconocer los objetos, localizarlos, describirlos y finalmente entregar la interpretación de la información de entrada al computador.

4. OBJETIVOS

La problemática tratada y las herramientas disponibles para tratarla son de fácil acceso y manejo. Requieren de una integración entre sistemas para lograr una mejora en este campo de acción. Es por esto que se define como objetivos claros y específicos lo siguiente:

4.1 OBJETIVO GENERAL

Diseñar una superficie plana que capture más de 3 grados de libertad para permitirle al usuario interactuar con una aplicación de visualización de fotos en un computador personal en la cual podrá rotar, trasladar y escalar una imagen más rápido, con menos pasos, menos estrés repetitivo y con menos capacitación que con los métodos tradicionales (Mouse, Teclado).

4.2 OBJETIVOS ESPECÍFICOS

Disminuir el tiempo requerido para llevar a cabo una tarea en un computador personal. Se utilizará como tarea de referencia el escalar, rotar y trasladar una imagen.

Disminuir la cantidad de pasos requeridos para llevar a cabo una tarea en un computador personal. Se utilizará como tarea de referencia el escalar, rotar y trasladar una imagen.

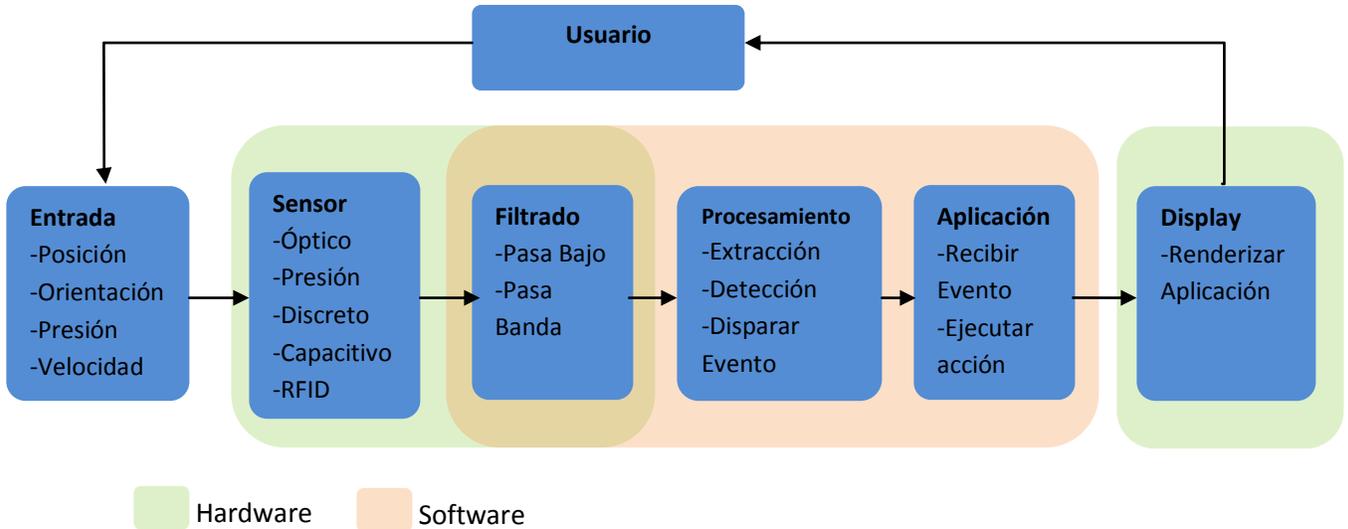
Disminuir la capacitación requerida para llevar a cabo una tarea sencilla en un computador personal. Se utilizará como tarea de referencia el escalar, rotar y trasladar una imagen.

Ofrecer un tiempo de respuesta ágil para asegurar fluidez en la experiencia.

5. EL SISTEMA

El sistema como un todo deberá cumplir con los objetivos planteados. Este estará compuesto de diversos subsistemas, cada uno concebido para lograr el funcionamiento óptimo del sistema global.

Figura 3. – Diagrama de Subsistemas



5.1 SENSOR

Las diversas formas de adquirir una señal para esta aplicación permiten utilizar varios tipos de sensores. Nótese que la mayor cantidad de desarrollos hacen uso de sensores ópticos ya que el paquete de información que debe recibir el sistema de procesamiento es un arreglo de 2 dimensiones estandarizado como una imagen. Estos sensores se pueden categorizar de la siguiente manera:

5.1.1 ÓPTICOS

Los sensores ópticos utilizados para esta aplicación son los fotodiodos, fototransistores, leds tradicionales o cámaras con sensores CMOS o CCD. Basado en la configuración de estos se logran distintos resultados en el sistema global.

Configuraciones

La configuración más famosa puede ser la configuración FTIR (Frustrated Total Internal Reflection) por Jeff Han. Esta configuración emplea una cámara enfocando una superficie translúcida la cual es iluminada con leds infrarrojos desde el contorno hacia el interior de la superficie. Este posicionamiento logra que la luz viaje dentro del cristal hasta el extremo opuesto aprovechando el fenómeno de reflexión total interna. Al hacer contacto en la superficie con un objeto blando como la piel, el índice de refracción fuera del cristal cambia haciendo que la luz se refleje perpendicularmente a la superficie. Esto genera un destello de luz donde se genera un contacto en la superficie. La luz no es visible por estar fuera del espectro visible para los humanos pero si es detectada por las cámaras digitales modernas. De esta forma la cámara obtiene un arreglo de dos dimensiones en el cual la intensidad de cada pixel denota un punto de toque. Este sistema al trabajar con luz infrarroja es muy sensible a la iluminación externa ya que cualquier bombillo incandescente o luz solar afectará negativamente el funcionamiento de este. Adicionalmente se requiere de una arquitectura con una profundidad considerable pues la cámara debe ser posicionada a una distancia proporcional a su campo de visión respecto a la superficie.

La configuración DI (Diffuse Illumination) consta de una superficie translúcida iluminada desde atrás por emisores infrarrojos. La luz atraviesa el cristal pero en el momento que un objeto se acerque a la superficie, la luz es reflejada de vuelta y es captada de la misma manera que el FTIR. Es más sensible a la luz ambiental pero en ciertas ocasiones permite detectar la proximidad del punto de toque a pequeñas distancias sobre la superficie.

Los lasers infrarrojos se utilizan para crear una capa de luz infrarroja sobre la superficie. En esta configuración cuando un objeto esta cerca de la superficie este interrumpe el haz de luz y genera un reflejo difuso en su contorno. Este es captado por una cámara de la misma manera que el FTIR. Es susceptible a sufrir de oclusión y ruido por iluminación ambiental.

La configuración de Cortina emplea una matriz de leds emisores y receptores en el perímetro de la superficie. Esto genera una cortina de luz infrarroja que puede ser activada por coordenadas individuales. Cuando un objeto se acerca a la superficie el haz de luz es interrumpido y la información es recolectada por un microprocesador. Es muy susceptible a sufrir de oclusión y detecta el toque sin tener contacto alguno con la superficie.

La Matriz de Leds se utiliza como display y sensor a la vez. El led actúa como display al emitir luz pero si la emisión es modulada, el led puede detectar

intensidad de luz durante el periodo de tiempo que este está apagado. Este fenómeno es aprovechado para esta problemática pues al acercar un objeto a la superficie hecha de una matriz de leds, este refleja la luz emitida por el led debajo de este. El led detecta el cambio en intensidad de luz reflejada y la información se recolecta por un microprocesador. La resolución de este tipo de sistema es determinada por el tamaño y la cantidad de leds en el sensor. Adicionalmente es sensible a iluminación ambiental.

La configuración libre emplea una o más cámaras enfocadas a la superficie y delega toda la labor de reconocimiento al sistema procesador, ya sea por segmentación de color, morfología, marcadores, correspondencia estereoscópica, análisis de sombras y otros tipos de métodos de análisis de imágenes. Con este tipo de sistema se busca que el sistema de procesamiento interprete lo que las cámaras captan. Sin embargo es muy susceptible a variaciones por calibración y tiempos de respuesta altos. Adicionalmente requiere de gran poder computacional solo para cumplir tareas de reconocimiento lo cual hace que este sistema no sea práctico para el usuario común.

5.1.2 PRESIÓN

Recientemente se ha empleado el uso de resistencias sensibles a la presión en aplicaciones multientrada. Se hace uso de tintas resistivas o arreglos de pequeños sensores de presión para detectar la presión aplicada sobre un punto en una superficie en una o dos dimensiones. La tinta resistiva es aplicada a una matriz de electrodos verticales y a otra horizontales. Posteriormente las capas resistivas se unen y un micro controlador hace un barrido por los pares de electrodos detectando los cambios de voltaje correspondientes a cambios de presión sobre cada intersección de los electrodos. Es fácil ver que el sensor es leído de la misma manera en que se lee un CCD de una cámara digital por lo cual no sufre de oclusión.

5.1.3 DISCRETO

La configuración con pantallas táctiles resistivas es posible hoy en día para múltiples entradas. Esta consta de una capa de soporte transparente, una capa de electrodos, micro espaciadores, otra capa de electrodos y otra capa de soporte transparente. Esta arquitectura permite obtener puntos de contacto discretos en los cuales no se obtiene información sobre la presión con la cual se está tocando la superficie.

5.1.4 CAPACITIVO

Los sensores capacitivos permiten múltiples entradas pero con restricciones. El uso de sensores capacitivos multientrada es más reconocido en el iPhone de Apple Computers. Este dispositivo de mano emplea múltiples entradas para eliminar la necesidad de un teclado tradicional. Sin embargo está restringido a 2 puntos de entrada por factores de diseño, poder de procesamiento y dimensiones físicas. Este tipo de sensor no sufre de oclusión pero está restringido a superficies planas y rígidas. Dada la naturaleza del sensor tampoco detecta puntos de contacto de objetos no dieléctricos como guantes, stylus y demás. Puede medir una presión virtual con base en la deformación del objeto que hace contacto con la superficie o por medio de la variación de la capacitancia entre capas de material compresible.

5.1.5 RFID

La marca Wacom hace uso de la tecnología RFID para sus tabletas digitalizadoras. Estas no detectan las manos del usuario pero si detectan con gran precisión hasta 2 stylus o mouse de su propia marca ya que establecen una comunicación directa entre la superficie y la herramienta.

5.2 FILTRADO

El filtrado de la señal obtenida elimina ruidos y puede realizarse en el hardware o en el software. En sensores que emplean un micro controlador para la adquisición de datos el filtrado puede realizarse en el mismo circuito. Para sensores ópticos como cámaras el filtrado se realiza en software. Sin embargo es recomendable filtrar en ambas etapas pues el filtro en hardware por ser dedicado elimina tiempo de procesamiento y permite una etapa de filtrado más flexible en el software.

La señal que se debe obtener del sensor es un arreglo en dos dimensiones de valores continuos o discretos. En el caso de valores continuos la magnitud de estos contiene información respecto a la naturaleza del toque como presión, distancia de proximidad o quizás temperatura. Esta señal deberá ser filtrada inicialmente para eliminar ruidos de alta frecuencia tanto en el tiempo como en el espacio.

Métodos de filtrado en el tiempo incluyen el filtro de media móvil o filtros FIR o IIR pasa bajos con frecuencias de corte definidas por la aplicación que se le dará al sensor. Esta frecuencia para aplicaciones multimedia controladas por humanos se puede definir con base en la frecuencia de muestreo del sistema óptico. El sistema óptico humano detecta en promedio 24 cuadros por segundo. Ya que el ser humano no puede procesar mas información de esta por segundo es seguro decir que no puede ejecutar acciones consientes a mayor frecuencia para aplicaciones de uso general. Si la frecuencia de muestreo del sensor permite utilizar filtros con frecuencia de corte superior a 24Hz se puede considerar el uso de este sensor para otras aplicaciones como sintetizadores o simuladores de instrumentos musicales.

Los métodos de filtrado espacial permiten eliminar ruido con base en la morfología del objeto que hace contacto con la superficie. Para esto se pueden implementar filtros de suavizado o realzado utilizados en procesamiento de imágenes. Estos filtros realizan una convolución entre una vecindad de pixeles y una máscara discreta o gaussiana para obtener un promedio ponderado para cada pixel en la imagen. De esta forma se elimina el ruido de alta frecuencia.

5.3 PROCESAMIENTO

El procesamiento que debe recibir la señal depende de la configuración y sensor utilizado. Sin embargo todos los enfoques deben seguir el siguiente orden en el manejo de la información: filtrar->segmentar->extraer->analizar->disparar evento en aplicación.

El sistema de procesamiento recibe la información filtrada para eliminar ruidos de alta frecuencia. De aquí en adelante los procesos aplicados a la información dependen de la arquitectura y configuración del sistema. Ya que todos los sensores entregan una estructura de datos semejante a una imagen podemos tratar la información como tal. Esta puede ser en escala de grises como la que entrega una cámara en escala de grises o una matriz de sensores resistivos. También puede ser un set de arreglos como en el caso de una cámara a color donde cada arreglo representa un color, siendo el espacio de color más común el RGB. Con base en esto se clasifica el procesamiento según la información recibida, o una imagen a color o una imagen en escala de grises.

Los sistemas a color hacen uso del color de los objetos a reconocer para llevar a cabo la tarea de segmentación y extracción de puntos de interés. Por ejemplo

sistemas de reconocimiento facial usaran segmentación de colores con valores cercanos al de la piel humana. De esta forma se logra extraer la cara del fondo para continuar con el procesamiento y eliminar la información que no es de interés para el sistema. Este tipo de procesamiento usualmente requiere de mucho mas poder computacional y trabaja a menores frecuencias de muestreo.

Los sistemas en escala de grises realizan la segmentación con procesos de umbral donde cada pixel es clasificado como objeto de interés o no haciendo referencia a su valor respecto al valor del umbral. Estos sistemas pueden estar apoyados por procesos de detección de bordes, erosión o dilatación.

Una vez ejecutado el proceso de extracción de los puntos de interés se realiza un análisis de morfología sobre esta información. Se realiza un análisis de conectividad. Este proceso identifica cada grupo de pixeles para que estos puedan ser analizados por separado. De esta forma a cada objeto se le calcula la masa, o cantidad de pixeles en él. Con operaciones geométricas se obtiene el centroide, ubicación, perímetro y orientación. Con este proceso se logra tener un listado de objetos con sus correspondientes características. En base a esta información se realiza un filtrado donde los objetos de menores dimensiones serán ignorados. Como por ejemplo puntos de contacto más grandes o pequeños que el corte transversal de un dedo humano serán ignorados. Esto se torna útil al eliminar falsos toques con la palma de la mano al apoyar la mano sobre la superficie.

Hasta este punto el análisis ha sido aplicado sólo a un cuadro de la secuencia de imágenes entregada por el sensor. Este listado de objetos tiene una identificación asignada a cada uno. En el cuadro posterior cada objeto puede o no haber cambiado de ubicación. Apoyándose en esto se realizan cálculos para encontrar la menor distancia entre los objetos del cuadro actual con el anterior. Así se establece una identificación más probable de cada objeto a lo largo del tiempo. Este proceso incrementa su precisión proporcionalmente a la velocidad de muestreo pues las distancias de desplazamiento entre cuadro y cuadro serán menores.

Una vez establecida la identidad de cada objeto se calcula su velocidad y al usar procesos espaciales avanzados como el análisis de posición con redes neuronales o sumatoria de desplazamientos, se identifican patrones de movimiento denominados: gestos. Estos gestos constan de un toque en la superficie que denota un movimiento particular como por ejemplo hacer scroll en un documento al desplazar dos dedos hacia arriba o abajo. Quizás el más común es el gesto de pinchar, donde dos dedos se acercan o se alejan generando un efecto de

escalado en el contenido. Una vez terminado el gesto, el objeto deja de tener contacto con la superficie y el sistema de identificación detecta que el objeto ya no está presente, de esta forma disparando un evento *TouchUp* semejante al *MouseRelease* en lenguajes de programación tradicionales. Ya que el sistema presenta una alta cantidad de grados de libertad la cantidad posible de gestos es infinita y solo está condicionada por la aplicación. Los gestos más básicos y versátiles, con su análogo en sistemas controlados por un mouse, son los siguientes.

Tabla 2 – Comparación Eventos Sistemas Multientrada v.s. Sistemas con Mouse

| Sistemas Multientrada | Sistemas con Mouse | Descripción |
|------------------------------|---------------------------|---|
| TouchDown | MouseDown | Evento disparado cuando se inicia un toque |
| TouchDrag | MouseMove | Evento disparado cuando un toque presenta movimiento |
| TouchUp | MouseUp | Evento disparado cuando un toque finaliza (el objeto deja de tocar la superficie) |
| Pinch | N.D. | Evento disparado cuando se presenta un gesto de pinchazo utilizado para hacer zoom o escalado |
| Swipe | Scroll | Evento disparado cuando se presenta un gesto lineal utilizado para hacer scroll horizontal o vertical |

Existe una gran variedad de sistemas de procesamiento para este propósito, sin embargo todos son para sistemas que utilizan cámaras como sensor. Intel provee la librería OpenCV la cual contiene muchas operaciones previamente implementadas, desde la adquisición de datos de la cámara hasta la detección de rostros y manos. Grupos independientes han creado soluciones de código libre como touchlib programada en c++ y OpenCV o CCV (Community Core Vision). Reactable, otra marca fabricante de equipos multientrada permite descargar su código de forma gratuita desde su página web. Touchless es otro sistema de procesamiento de equipos multientrada hecha por Mike Wasserman. También está la opción de utilizar DirectShow en Microsoft Windows para la adquisición de datos de una cámara, v4l en sistemas Linux o Cocoa en sistemas Mac. Estos sistemas proveen módulos de procesamiento para sistemas basados en cámaras de forma gratuita pero su uso permanece experimental y/o publicitario. Para

sistemas que no utilizan cámara no hay software disponible pues incluye un desarrollo de hardware para cada instalación. Por lo tanto la solución de procesamiento deberá ser diseñada a la medida.

Dado que ésta tecnología está avanzando a pasos agigantados gracias al poder computacional disponible hoy en día. Existen comunidades que han desarrollado sistemas de procesamiento para procesar múltiples entradas con una cámara. Estos sistemas de procesamiento son distribuidos gratuitamente bajo licencias de código libre. Estos son los de mejor desempeño:

Beespoke

Escrito en C# hace uso del .NET framework por lo cual solo trabaja en Windows. Permite múltiples entradas, reconocimiento de gestos, marcadores y comunicación por OSC(Open Sound Protocol). Actualmente se trabaja en la versión Linux que corre sobre el runtime de .NET para este sistema operativo llamado Mono.

Reactivision

Código libre desarrollado en C++ lo cual le permite trabajar sobre cualquier plataforma. Permite el reconocimiento de múltiples entradas y marcadores.

Community Core Vision (CCV)

Plataforma en código libre que funciona en cualquier sistema operativo. Reconoce múltiples entradas.

Touche

Código libre con procesamiento de múltiples entradas para la plataforma Mac.

Touchlib

Sistema de procesamiento para múltiples entradas en Windows. Actualmente su desarrollo se encuentra inactivo.

Para mantener la flexibilidad en el sistema solo se consideraran los procesadores multi-plataforma para así permitir que el sistema funcione en los sistemas operativos más usados.

5.4 APLICACIÓN

La información enviada por el sistema de procesamiento es recibida por la aplicación que el usuario está utilizando. Este subsistema es clave para el éxito de este sistema pues los programas actuales no fueron diseñados para recibir

múltiples entradas. Esto limita a gran medida la facilidad de adopción de este sistema a nivel comercial pues se estarían dedicando recursos a un sistema que en última instancia no puede ser utilizado en su totalidad. Es por esto que la aplicación también debe ser desarrollada para recibir eventos multientrada y explotar todo el potencial de este método de interacción. La información del evento para los casos de aplicaciones en código libre se transmite en el protocolo TUIO (Touch User Interface Object) el cual está ganando bastante aceptación entre la comunidad de desarrolladores. No obstante se debe tener en cuenta que los sistemas operativos a finales de este año ya todos emplearan su propio protocolo multientrada y no existen estándares definidos por lo cual la batalla para lograr el mejor protocolo de comunicación multientrada aun está vigente.

La aplicación puede recibir la comunicación vía sockets al correr como una instancia individual o se puede emplear una comunicación vía javascript en aplicaciones desarrolladas en Adobe Flash/Actionscript embebidas sobre la aplicación madre.

5.5 DISPLAY

El display para entregarle la información al usuario y observar el resultado de su entrada al sistema también depende de la configuración de este. Sin embargo los avances en el desarrollo de conductores transparentes hechos con ITO (Indium Tin Oxide) permiten crear superficies con sensores capacitivos o resistivos transparentes que pueden ser sobrepuestos sobre cualquier tipo de pantalla, o se puede proyectar sobre estos.

No hay restricciones si la pantalla es un LCD, CRT o Proyectada, pero sí afecta la calidad de la imagen entregada al usuario y la inmunidad al ruido del sistema.

6. BENCHMARK

El benchmark para este sistema no se realizará para cada subsistema dada la gran variedad de configuraciones posibles. En este caso se hará un benchmark respecto a la tarea de referencia planteada en los objetivos: rotar, escalar y trasladar una imagen.

Para realizar esta tarea de referencia se dispone de distintos métodos, siendo la función de rotar 90 grados con un solo botón la más básica hasta la función de transformación en Adobe Photoshop CS la más compleja.

La prueba se realiza utilizando Adobe Photoshop, en el cual se medirá el tiempo requerido para rotar, escalar y trasladar la imagen a una posición central en la pantalla a un ángulo de 45 grados, usando un mouse tradicional, iniciando con el cursor e imagen en posiciones aleatorias en la pantalla. Se realiza la prueba para usuarios de distinto desempeño en el uso de computadores personales con base en las horas de uso semanales y se obtiene el tiempo requerido por cada usuario para ejecutar los cambios en las tres variables de la imagen (rotación, traslación y escalado).

Usuarios Básicos son aquellos que usan un computador entre 0 y 1 hora semanal. Los usuarios promedio son aquellos que usan un computador entre 1 y 5 horas semanales. Los usuarios avanzados son aquellos que usan un computador 6 horas o más a la semana.

Tabla 3 - Tiempos de Rotación, traslación y escalado para usuarios BASICOS

| Tiempo(s) | Prueba # | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | Promedio Usuario(s) |
|-----------|-------------------------|-----------|------|----------------------|------|------|------|--------------------------|------|---------------------|
| | | Usuario 1 | 34.4 | 32.5 | 32.8 | 35.8 | 33.4 | 33.8 | 33.6 | 29 |
| | Usuario 2 | 30.5 | 35 | 33.3 | 33.8 | 37 | 34.7 | 32.2 | 31.2 | 33.5 |
| | Usuario 3 | 29.2 | 33.3 | 36.3 | 28 | 35.2 | 33 | 30.1 | 34.2 | 32.4 |
| | Usuario 4 | 30.5 | 36 | 29.6 | 31 | 32.4 | 28.1 | 31.4 | 26.8 | 30.7 |
| | Usuario 5 | 33.7 | 35.6 | 27.9 | 35.9 | 28.6 | 31.9 | 28.2 | 37.3 | 32.4 |
| | Usuario 6 | 27.7 | 35.6 | 30.8 | 28.6 | 34.6 | 32.1 | 34 | 32.6 | 32.0 |
| | Usuario 7 | 34.6 | 35.3 | 28.4 | 32.1 | 29.5 | 30.9 | 31.6 | 30.7 | 31.6 |
| | Usuario 8 | 25.9 | 26.7 | 28.6 | 28.1 | 27 | 27.2 | 31.2 | 35 | 28.7 |
| | Tiempo Minimo(s) | | 25.9 | Tiempo Max(s) | | 37.3 | | Promedio Total(s) | | 31.8 |

Tabla 4 - Tiempos de Rotación, traslación y escalado para usuarios PROMEDIO

| Tiempo(s) | Prueba # | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | Promedio Usuario(s) |
|-----------|-------------------------|-----------|------|----------------------|------|------|------|--------------------------|------|---------------------|
| | | Usuario 1 | 19.2 | 20.7 | 18.5 | 16 | 22.3 | 20.1 | 20.2 | 15 |
| | Usuario 2 | 19.3 | 18 | 20.6 | 17.5 | 22.5 | 19.8 | 16.4 | 22.5 | 19.6 |
| | Usuario 3 | 21.7 | 19.5 | 22.6 | 15.7 | 15.7 | 22.5 | 16.3 | 16.3 | 18.8 |
| | Usuario 4 | 14.9 | 22.3 | 19.4 | 18.9 | 17.2 | 15.8 | 15.7 | 17.6 | 17.7 |
| | Usuario 5 | 20.6 | 20.2 | 19.9 | 22.1 | 17 | 22.7 | 18.7 | 16.1 | 19.7 |
| | Usuario 6 | 20.1 | 22.4 | 15.9 | 21.8 | 17.3 | 19.9 | 17.7 | 21.4 | 19.6 |
| | Usuario 7 | 21 | 22.4 | 20.2 | 18.2 | 20.7 | 21.5 | 20.2 | 20 | 20.5 |
| | Usuario 8 | 17.5 | 15.5 | 18.4 | 15 | 19.7 | 20.5 | 21.2 | 16.2 | 18.0 |
| | Tiempo Minimo(s) | | 14.9 | Tiempo Max(s) | | 22.7 | | Promedio Total(s) | | 19.1 |

Tabla 5 - Tiempos de Rotación, traslación y escalado para usuarios AVANZADOS

| Tiempo(s) | Prueba # | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | Promedio Usuario(s) |
|-----------|-------------------------|-----------|-----|----------------------|-----|-----|-----|--------------------------|-----|---------------------|
| | | Usuario 1 | 9.6 | 9.2 | 9.3 | 8.8 | 7.5 | 8.3 | 9 | 8 |
| | Usuario 2 | 9.1 | 7.4 | 7.2 | 8.5 | 7.7 | 9.5 | 7.8 | 7.3 | 8.1 |
| | Usuario 3 | 8.5 | 9.2 | 7.7 | 8.4 | 7.4 | 9.1 | 7.4 | 8.8 | 8.3 |
| | Usuario 4 | 7.8 | 7.4 | 8.8 | 7.2 | 9 | 7.4 | 8.8 | 7.6 | 8.0 |
| | Usuario 5 | 7 | 7.9 | 8 | 8.6 | 7.9 | 7.8 | 8.7 | 9 | 8.1 |
| | Usuario 6 | 8.6 | 8.2 | 8 | 7.3 | 7.9 | 8.9 | 7.9 | 7 | 8.0 |
| | Usuario 7 | 7.7 | 8.3 | 7.8 | 7.7 | 8.8 | 8.6 | 9 | 9.5 | 8.4 |
| | Usuario 8 | 7.3 | 9 | 8.5 | 7.7 | 9.2 | 8.5 | 8.9 | 9.2 | 8.5 |
| | Tiempo Minimo(s) | | 7 | Tiempo Max(s) | | 9.6 | | Promedio Total(s) | | 8.3 |

Es fácil ver que una tarea tan sencilla en realidad toma más tiempo del esperado. Si se logra optimizar una tarea tan sencilla, los avances que se podrían lograr con esquemas de interacción más complejos pueden hacer que la acción de interactuar con un computador sea más placentera o en tal caso mínima. Cabe notar que el cambio en estas variables bajo este método depende del flujo de operaciones que se da en el programa en base al dispositivo de entrada. Es decir, el tiempo requerido depende de los pasos que deba seguir el usuario en el programa y no depende de la magnitud del ángulo, escalado o traslación.

7. CONCEPTOS

Los distintos subsistemas serán analizados y calificados para establecer el aporte a los objetivos planteados. Dado que la problemática puede ser tratada desde varias perspectivas con distintas tecnologías que en varios casos son mutuamente excluyentes, los subsistemas no serán elegidos según su calificación individual sino en el sistema completo final.

En la generación de conceptos se obtienen distintas configuraciones de sensor-procesador ya que los demás componentes del sistema no tienen mayor efecto sobre el resultado deseado. Estos conceptos surgen en base a sistemas existentes, tecnologías en desarrollo y experimentación bajo una condición: todo sistema multitoque debe entregar un paquete de información semejante a una imagen digital, es decir, un arreglo de 2 o más dimensiones de valores medidos por el sensor. Bajo esta condición es prudente considerar sensores basados en cámaras digitales pues éstas fueron diseñadas para tal propósito; entregar una secuencia de imágenes al PC. Esta secuencia de imágenes debe ser procesada. Para esto se siguen distintas metodologías basadas en procesamiento de imágenes digitales pero en este caso se planteara conceptos utilizando sistemas de procesamiento de código libre como CCV o OpenCV desarrollados para este propósito. Se considerara el concepto propuesto por Jeff Han el cual aprovecha la naturaleza dual de un LED para emitir y detectar el reflejo de su propia luz al tener la presencia de un objeto cerca. Este concepto consta de una matriz de LEDs que oscilan entre los estados de recepción y emisión. Los sistemas comerciales como pantallas táctiles resistivas y capacitivas se consideraran como también los más avanzados sistemas de tabletas digitalizadoras Wacom que hacen uso de la tecnología RFID. Se incluye un concepto que se encuentra en etapa experimental que hace uso de tintas especiales para crear un sistema resistivo variable a la presión. Como concepto de referencia se define el mouse tradicional. De esta forma se plantea que los conceptos son los siguientes.

Tabla 6 - Conceptos en base a tipo de sensor y procesador

| Concepto | Sensor | Procesador |
|-----------------|------------------|-----------------------------------|
| A | Cámara(FTIR) | CCV (Community Core Vision) |
| B | Cámara(DI) | CCV (Community Core Vision) |
| C | Cámara(laser) | CCV (Community Core Vision) |
| D | Cortina de leds | Microprocesador |
| E | Cámara | OpenCV (Segmentación Morfológica) |
| F | Cámara (estéreo) | OpenCV(Segmentación |

| | | |
|---|-------------------------------------|-----------------|
| | | Estereoscópica) |
| G | Matriz de leds | Microprocesador |
| H | Resistencia variable a la presión | Microprocesador |
| I | Discreto(pantalla táctil resistiva) | Microprocesador |
| J | Capacitivo | Microprocesador |
| K | RFID | Microprocesador |
| R | Mouse (Concepto de Referencia) | Microprocesador |

Estos conceptos serán analizados de acuerdo a los siguientes criterios:

- Tiempo de respuesta
- Inmunidad al ruido
- Inmunidad a la oclusión
- Volumen Mínimo
- Dimensiones Superficie Máximas
- Entradas Máximas
- Requerimiento de herramientas o marcadores para su uso
- Tiempo de configuración
- Velocidad mínima CPU
- Costo
- Presión requerida para activar la entrada
- Reconocimiento de marcadores

Conceptos

| Criterios | A | B | C | D | E | F | G | H | I | J | K | R |
|--|---|---|----|---|----|----|---|----|----|---|---|---|
| Tiempo de respuesta | - | - | - | + | - | - | + | + | + | + | + | 0 |
| Inmunidad al ruido | 0 | - | - | 0 | 0 | 0 | 0 | + | + | + | + | 0 |
| Inmunidad a la oclusión | + | + | 0 | 0 | 0 | 0 | + | + | - | 0 | 0 | 0 |
| Volumen Mínimo | - | - | - | + | - | - | + | + | + | + | + | 0 |
| Dimensiones Superficie Máximas | + | + | + | + | 0 | 0 | 0 | 0 | + | - | 0 | 0 |
| Entradas Máximas | + | + | 0 | 0 | 0 | 0 | + | + | - | 0 | - | 0 |
| Requerimiento de herramientas o marcadores para su uso | + | + | + | + | + | + | + | + | 0 | 0 | - | 0 |
| Tiempo de configuración | 0 | - | - | 0 | 0 | - | 0 | 0 | 0 | 0 | 0 | 0 |
| Velocidad mínima CPU | - | - | - | 0 | - | - | 0 | 0 | + | + | + | 0 |
| Costo | + | + | 0 | 0 | 0 | - | - | 0 | 0 | - | - | 0 |
| Presión requerida para activar la entrada | - | + | + | + | + | + | + | + | - | + | + | 0 |
| Reconocimiento de marcadores | - | + | 0 | 0 | + | + | 0 | 0 | - | - | + | 0 |
| Positivos | 5 | 7 | 3 | 5 | 3 | 3 | 6 | 7 | 5 | 5 | 5 | |
| Iguales | 2 | 0 | 4 | 7 | 6 | 4 | 5 | 5 | 3 | 4 | 4 | |
| Negativos | 5 | 5 | 5 | 0 | 3 | 5 | 1 | 0 | 4 | 3 | 3 | |
| Total | 0 | 2 | -2 | 5 | 0 | -2 | 5 | 7 | 1 | 2 | 2 | |
| Orden | 8 | 4 | 11 | 3 | 9 | 10 | 2 | 1 | 7 | 5 | 6 | |
| Continuar? | | | si | | si | | | si | si | | | |

Según los criterios seleccionados para cumplir con los objetivos se obtienen 4 conceptos viables en el siguiente orden:

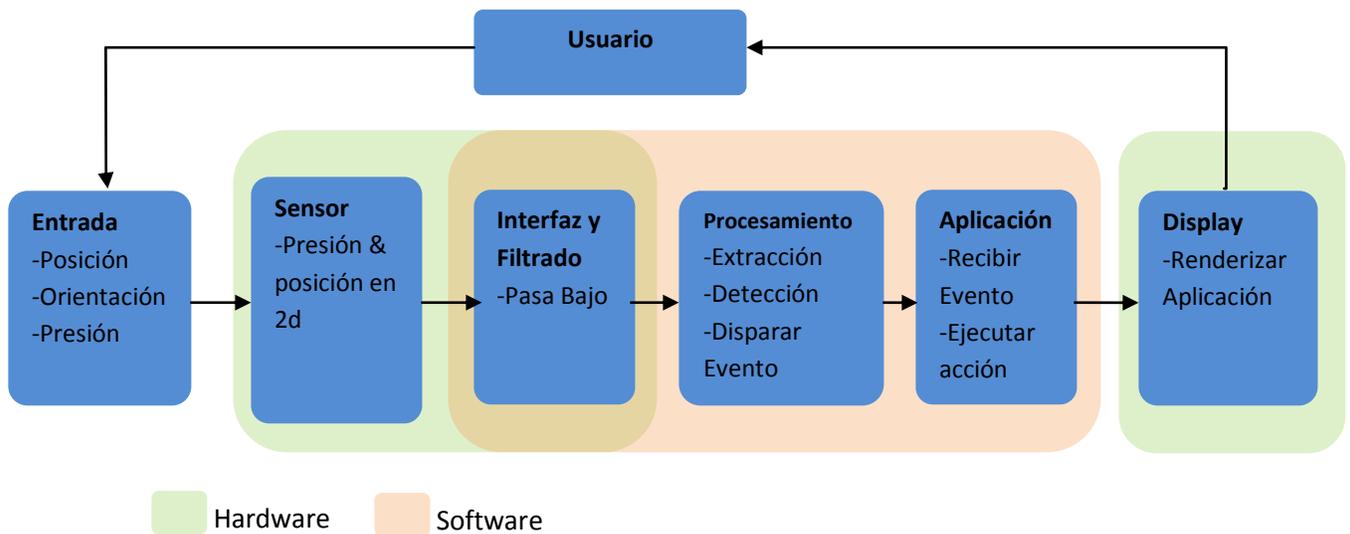
- H - Resistencia variable a la presión
- G- Matriz de Leds
- D- Cortina de Leds
- B- Cámara(DI)

El concepto H- Resistencia Variable a la presión será analizado y simulado para determinar los niveles de cumplimiento con los requerimientos del proyecto.

8. DISEÑO DEL SISTEMA

Tomando el concepto seleccionado se planteará el sistema completo para posteriormente desglosarlo en subsistemas modulares. Al utilizar una arquitectura modular se podrá analizar cada subsistema con base en el cumplimiento de su función en el sistema global y permitirá tener procesos de producción mucho más flexibles.

Figura 3. – Diagrama de Subsistemas



8.1 SENSOR

El sensor utilizado en el concepto seleccionado debe entregarle al sistema información de presión y coordenadas en 2D para cualquier cantidad de estimulaciones que éste reciba. Es más fácil ver la estructura de datos que el sensor debe entregar como una imagen de 8 bits por pixel en la cual cada pixel representa una coordenada x,y específica y la intensidad de dicho pixel puede tomar 256 valores al ser representada por un byte.

Ya que la estructura de datos requerida por el sistema es una imagen, lo más lógico es que el sensor adquiera esa información de la misma forma en que se digitaliza una imagen. Esto es haciendo un barrido x/y sobre la superficie. Al leer la información de esta forma se elimina cualquier problema de oclusión en los momentos en que se está tocando la superficie en más de un punto. Esto implica

que cada celda debe ser eléctricamente independiente de las demás. Con propósitos de prueba y simulación se planteará una superficie con 64 posiciones, es decir, 8 líneas y 8 columnas.

Para este subsistema se evaluarán varias hipótesis para lograr un funcionamiento correcto y un proceso de manufactura económico.

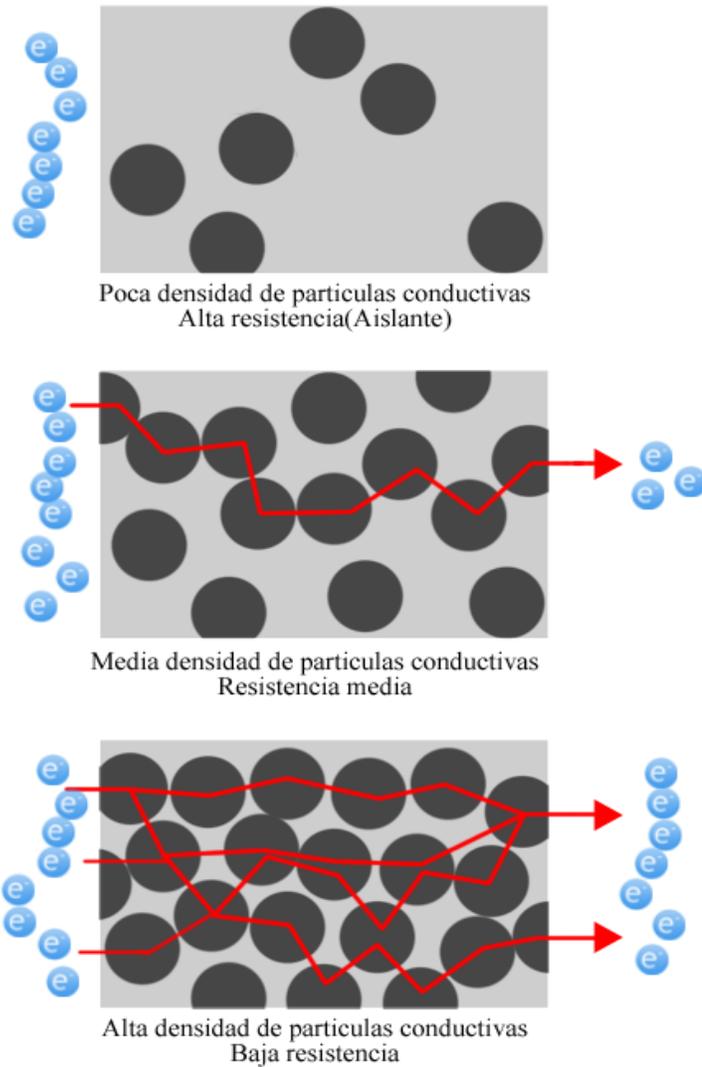
8.1.1 RESISTENCIA VARIABLE

Como módulo básico para el funcionamiento de este sensor se aprovecha el comportamiento de los compuestos resistivos. Se parte del comportamiento de un potenciómetro en el cual la arquitectura del mismo permite por medios mecánicos, variar la distancia entre dos puntos de contacto sobre una pista de material resistivo. Este tipo de material presenta una mayor resistencia a una mayor distancia, obteniendo así una resistencia variable o potenciómetro.

Un análisis más detallado del tipo de material resistivo usado en las pistas de los potenciómetros revela compuestos hechos con base en una matriz de soporte aislante y partículas muy finas de un material conductor. La relación entre las cantidades de estos dos materiales define la resistencia por metro del compuesto. Los materiales más comunes para la fabricación de las pistas de potenciómetros son el grafito, carbono en plástico y una mezcla de metal y cerámica llamada cermet.

A nivel microscópico este tipo de material contiene las partículas conductoras suspendidas sobre el medio de soporte aislante. A mayor proporción de partículas conductoras, más de estas estarán en contacto permitiendo así menor resistencia al flujo de electrones.

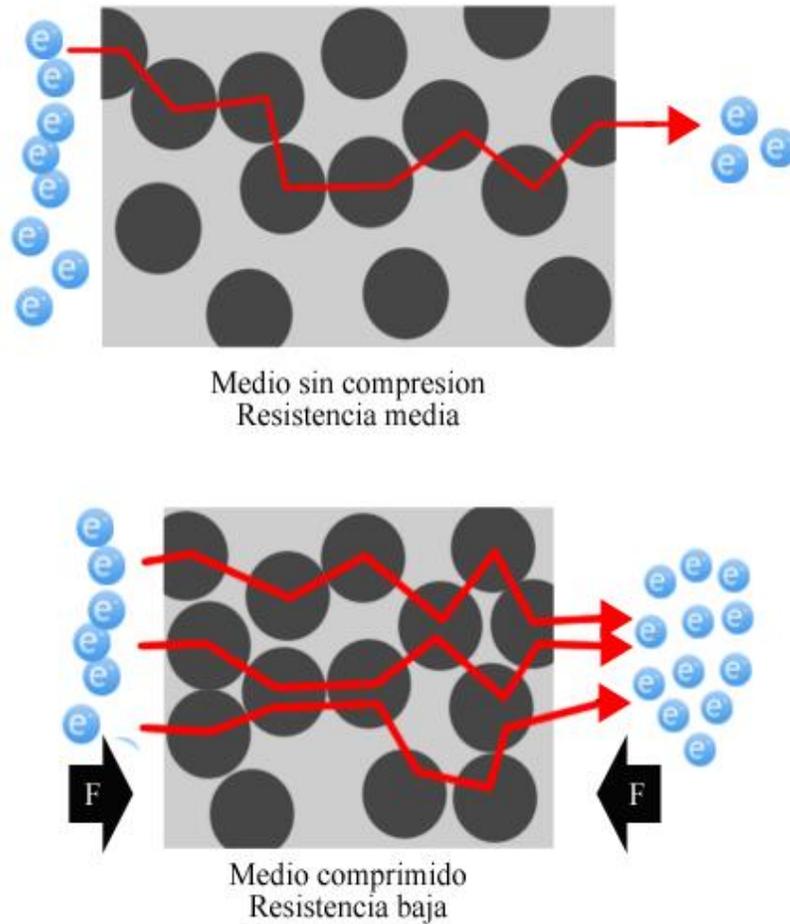
Figura 5. Diagrama de proporciones en la mezcla conductiva



Resultado de distintas proporciones de partículas conductoras suspendidas en el medio de soporte. Al haber más densidad de partículas conductoras se aumenta los puntos de contacto entre partículas por lo cual se presenta menor resistencia al flujo de electrones.

Si el medio de soporte de las partículas es compresible, el material presentara cambios en resistencia no solo en la distancia pero también cuando este está sujeto a una presión externa. Al ser compresible cualquier estímulo externo comprime el medio permitiendo que más partículas conductoras entren en contacto.

Figura 6. Diagrama comportamiento bajo compresión de la mezcla



Comportamiento eléctrico al estar sujeto a una presión externa. Al estar bajo presión las partículas conductivas entran en contacto permitiendo un mayor flujo de electrones.

Aprovechando este fenómeno es posible construir no solo resistencias variables en cuanto a distancia pero también en cuanto a compresión. Con base en este comportamiento se realizan distintas mezclas de compuestos resistivos con materiales de fácil adquisición en el mercado. Se analiza su comportamiento y se establecen parámetros de arquitectura para el sensor multitoque.

Según los objetivos generales del proyecto y la importancia del medio resistivo para el funcionamiento del sensor se tiene como requerimientos para este compuesto lo siguiente:

- Que pueda tomar la forma de una capa delgada.
- Que se adhiera al conductor y a la capa de soporte.
- Que presente un comportamiento eléctricamente resistivo
- Que presente un comportamiento eléctricamente lineal
- Que al haber contacto entre dos superficies de este material, no se adhieran
- Que no sea tóxico
- Que no requiera equipos especiales para aplicar y/o secar

Estos requerimientos los cumplen las tintas utilizadas a nivel experimental a la fecha. Estas trabajan aprovechando su estructura porosa la cual al ser sometida a la presión aumenta el contacto entre las partículas. La arquitectura de este sensor requiere de una capa de aire entre ambas capas de material resistivo, pues es el aire que se utiliza como aislante. Esto condiciona la construcción del sensor en cuanto a que la forma de este debe ser tal que conserva la capa de aire interna. Para este subsistema se plantea una alternativa nueva que será sujeta a experimentación en este trabajo. La tinta resistiva estará impregnada en un medio de soporte poroso. Esto conserva la porosidad requerida para la resistencia variable a la presión pero también permite que el sensor tome cualquier forma deseada ya que el soporte poroso mantiene los bolsillos de aire necesarios para su adecuado funcionamiento. Como soporte poroso se utilizara tela, específicamente franela de algodón.

8.1.2 TINTA RESISTIVA

Como material conductivo se utilizará grafito el cual es de fácil adquisición, presenta cualidades conductivas y su presentación en polvo permite una fácil elaboración del compuesto.

Como matriz de soporte se experimentó con pintura a base de vinilo, tinta textil a base de agua usada para estampado sobre textiles, Pegante para PVC, Pegante a base de agua y Tinta a base de PVC para estampado textil (Plastisol). A cada uno de estos se le agregó grafito logrando una mezcla de 55% soporte y 45% grafito.

Estos materiales fueron seleccionados como punto de partida experimental para poder observar el comportamiento de éste dentro del sistema. De acuerdo a los resultados se podrá utilizar estos mismos materiales en el producto final o se podrá utilizar compuestos diseñados específicamente para este uso aumentando el costo del dispositivo.

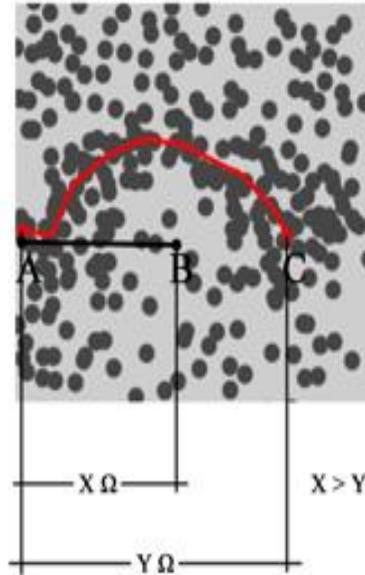
Se definen las mezclas de acuerdo a la siguiente tabla.

Tabla 7 - Resultados Mezclas Compuesto Resistivo

| | Mezcla 1 | Mezcla 2 | Mezcla 3 | Mezcla 4 | Mezcla 5 |
|------------------------------|------------------------------|---------------------------------|---|-----------------------------------|--------------------------------------|
| Descripción | 55% plastisol 45% grafito | 55% tinta textil 45% grafito | 55% pegante base agua 45% grafito | 55% pegante PVC 45% grafito | 55% pintura vinilo 45% grafito |
| Métodos de Aplicación | Screen, Espátula | Screen, Espátula | Screen, Espátula | Screen, Espátula | Screen, Espátula |
| Métodos de Secado | Material Cura a 120 °C | Secado al aire | Secado al aire | Secado al aire | Secado al aire |
| Flexible | Si | Si | Si | Si | Si |
| Compresible | Si | No | Si (Poco) | No | No |

Cada mezcla fue aplicada con espátula sobre una lamina de policarbonato transparente en secciones de 2 por 20 cms. Se aplicó una capa de cada mezcla pero el resultado fue un comportamiento eléctrico altamente no lineal. Las mediciones a intervalos constantes de distancia ocasionalmente resultaban en resistencias menores que las obtenidas en mediciones con puntos de contacto más cercanos. Se tiene como hipótesis a este comportamiento la falta de homogeneidad de la mezcla y el grosor de la capa crean secciones donde la resistencia varía respecto al resto de la mezcla.

Figura 6. Efectos por falta de homogeneidad en la mezcla

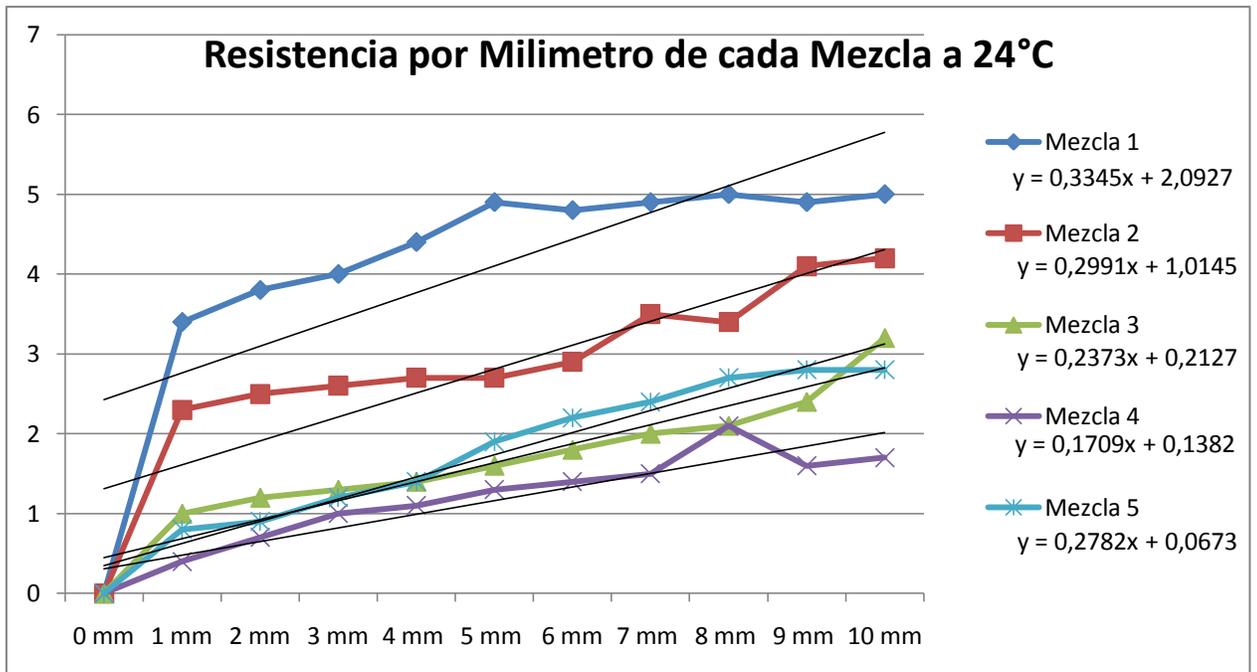


Por falta de homogeneidad en la mezcla. Se generan sectores donde la resistencia es menor a mayor distancia.

Se procede a extender los tiempos de mezclado de cada compuesto para asegurar una mejor homogeneidad y se aplican más capas en cada sección de prueba. Esto mejoró el comportamiento sustancialmente, sin embargo se requiere de procesos industriales para la elaboración de este material para asegurar un comportamiento totalmente lineal. Los resultados obtenidos de acuerdo a la resistencia por milímetro lineal son los siguientes.

Tabla 8 - Resistencia por milímetro (KΩ/mm) para las mezclas de prueba a 24°C

| Mezcla | Descripción | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|--------|--|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| | | mm | m | m | m | m | m | m | m | m | m | mm |
| 1 | 55% plastisol 45% grafito | 0 | 3.4 | 3.8 | 4.0 | 4.4 | 4.9 | 4.8 | 4.9 | 5.0 | 4.9 | 5.0 |
| 2 | 55% tinta textil 45% grafito | 0 | 2.3 | 2.5 | 2.6 | 2.7 | 2.7 | 2.9 | 3.5 | 3.4 | 4.1 | 4.2 |
| 3 | 55% pegamento base agua 45% grafito | 0 | 1.0 | 1.2 | 1.3 | 1.4 | 1.6 | 1.8 | 2.0 | 2.1 | 2.4 | 3.2 |
| 4 | 55% pegamento PVC 45% grafito | 0 | 0.4 | 0.7 | 1.0 | 1.1 | 1.3 | 1.4 | 1.5 | 2.1 | 1.6 | 1.7 |
| 5 | 55% pintura vinilo 45% grafito | 0 | 0.8 | 0.9 | 1.2 | 1.4 | 1.9 | 2.2 | 2.4 | 2.7 | 2.8 | 2.8 |



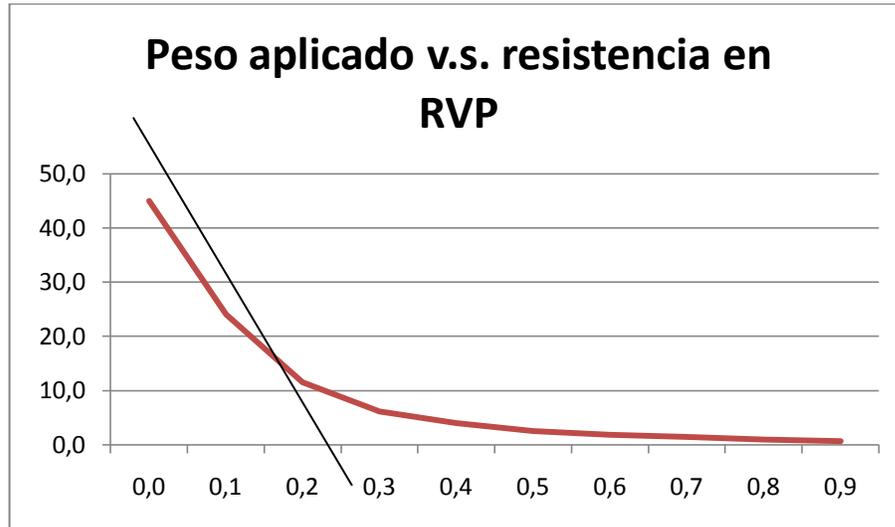
8.1.3 Celda de Presión

Se prevee que una mezcla resistiva con un medio compresible permite tener una resistencia variable a la presión. En las mezclas propuestas sólo aquella que contiene tinta para estampado a base de PVC (Plastisol) es compresible. Una vez curada, ésta se comporta como una lamina de caucho, resistente al agua y flexible. El problema con éste tipo de mezcla para ésta aplicación está en el método de curado pues el material sobre el cual se aplicará la mezcla debe soportar los 120°C. Por esta razón se plantea una alternativa la cual, por ser con materiales de fácil adquisición en el mercado, puede reducir costos de manufactura para el producto final.

A nivel microscópico la franela en algodón presenta características porosas lo que le da al material elasticidad, flexibilidad y más importante, compresibilidad. Se utiliza la mezcla 2 a base de tinta textil para teñir la franela. Esta es sumergida en la mezcla logrando que tanto la tinta como las partículas de grafito queden embebidas en los espacios internos de las fibras de algodón. Posteriormente la franela se deja secar al aire. Efectivamente este método produce una tela con un comportamiento eléctricamente resistivo ortogonal al plano formado por la tela, mas no a lo largo de la superficie. La siguiente tabla presenta las resistencias obtenidas a distintas presiones aplicadas a la tela.

Tabla 9 - Peso aplicado v.s. Resistencia en RVP (Resistencia variable a la presión)

| | | | | | | | | | | | |
|-----------|-----|------|------|------|-----|-----|-----|-----|-----|-----|-----|
| Kg | 0.0 | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 | 1.0 |
| KΩ | inf | 45.0 | 24.0 | 11.5 | 6.1 | 4.0 | 2.5 | 1.8 | 1.4 | 0.9 | 0.7 |



Es fácil ver que se presenta un comportamiento no lineal respecto a la presión. Para ésta aplicación se debe lograr un comportamiento tal que la menor presión posible ejercida por el dedo humano y la mayor presión cómodamente posible, representen los límites de máxima resistencia y menor resistencia respectivamente. Las medidas realizadas sobre la presión promedio realizada por el dedo humano al hacer click en el mouse entrega valores inferiores a los 200 gramos. Al observar la grafica vemos que el comportamiento de la resistencia variable a la presión en ese rango se puede asumir como lineal. La ecuación de la línea asumida para el sistema es

$$R_{rvp}(p) = 55,000 - 155,000 * p$$

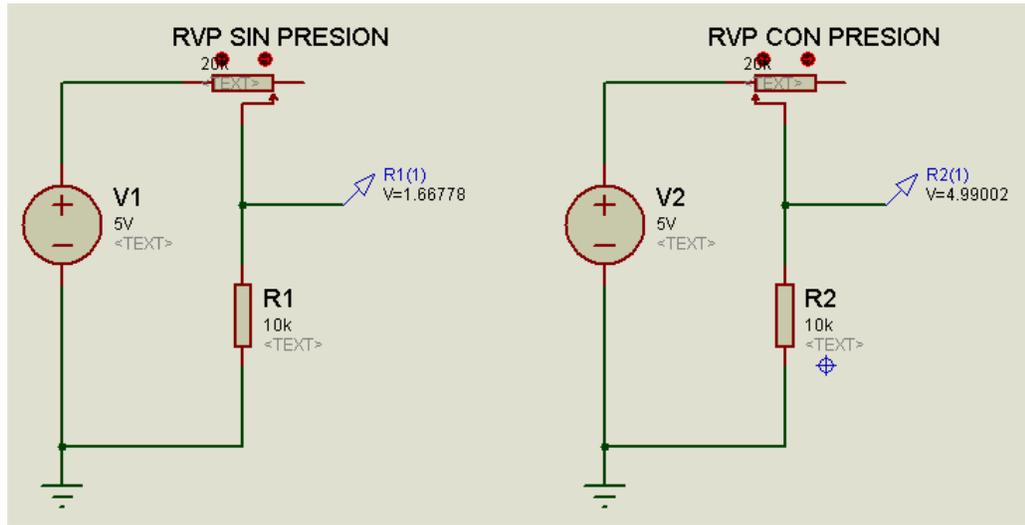
Donde

p = peso aplicado en el punto de contacto en Kg

R_{rvp} = Resistencia con peso p en Ω

Con este material es fácil construir una celda de presión. Basta suspender el material entre dos capas de conductores sujetos a una presión variable. La lectura es efectuada usando un divisor de voltaje logrando así una salida de voltaje variable respecto a la presión.

Figura 7. Circuito de lectura para la celda de presión.

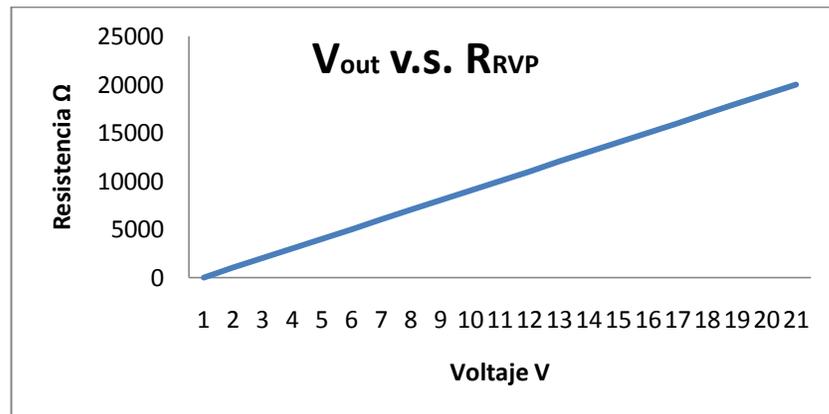


La celda o Resistencia Variable a la Presión (RVP) está representada por un lazo de un potenciómetro. Cuando la celda no tiene presión ejercida sobre si la resistencia es máxima. Al aplicar presión la resistencia disminuye proporcionalmente a la presión.

El comportamiento de la celda de presión está definido por la siguiente ecuación

$$V_{out}(t) = \frac{R2}{RVP(t) + R2} * V_{in}$$

Figura 8. Comportamiento del circuito de lectura

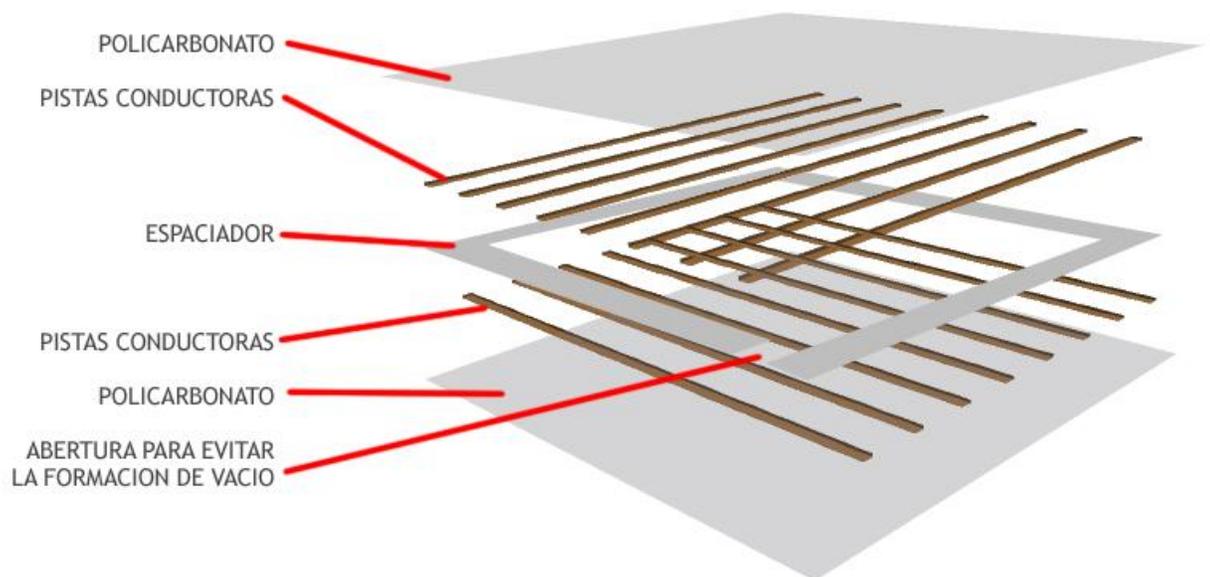


El comportamiento será asumido como lineal para puntos de presión inferiores a los 200 gramos.

8.1.4 Construcción del sensor

Con la celda de presión obtenida es posible construir un arreglo de celdas que obtengan la presión aplicada sobre distintos puntos en la superficie. Se plantea un sensor construido con capas externas de policarbonato el cual es resistente a los rayones y al agua, flexible para transferir la presión sobre un punto al circuito interno y permite deslizar la piel del dedo humano sobre su superficie. En cada capa externa se trazan electrodos, verticales en una capa y horizontales en la otra. Se ubica entre ambas capas un borde espaciador para mantener los electrodos fuera de contacto. Esto forma una matriz de intersecciones de dos dimensiones.

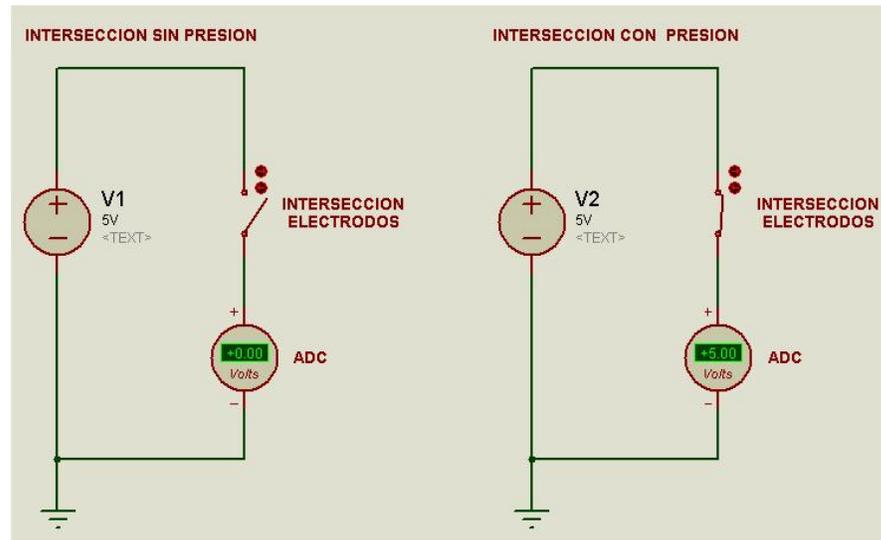
Figura 9. Sensor con matriz de electrodos



Intersecciones formadas por la matriz de electrodos verticales y horizontales.

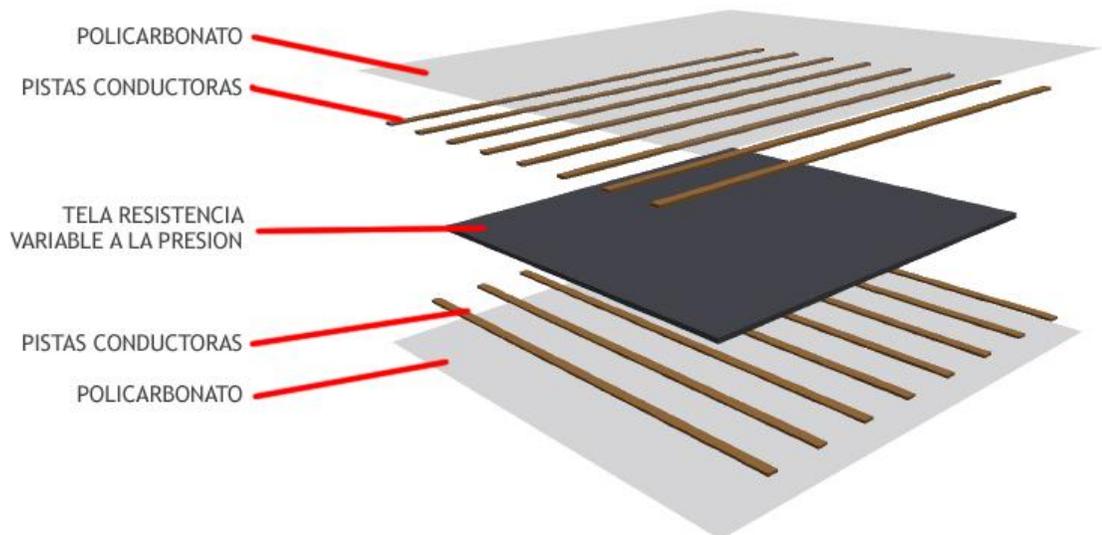
Este sistema da como resultado un sensor con tantas salidas como la suma de electrodos verticales y horizontales para un total de (electrodos verticales * electrodos horizontales) puntos de medida. Estas salidas se conectaran a un circuito de lectura que trabaja de la misma forma en que se lee un sensor CCD de una cámara digital. Este proceso de lectura será descrito con mayor detalle en la sección posterior. Al leer este sensor se obtienen valores discretos, donde el 0 representa un punto donde no hay presión aplicada y el 1 un punto donde si hay presión.

Figura 10. Esquema eléctrico de sensor discreto



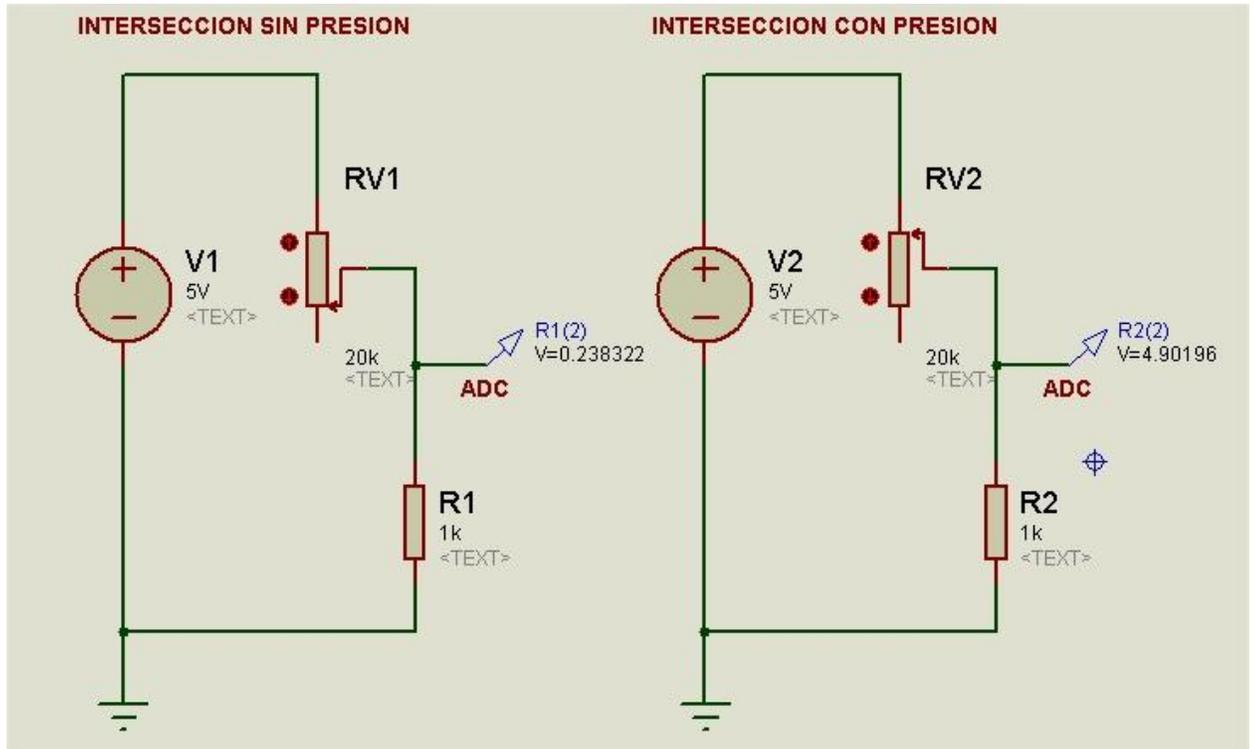
Sin embargo los requerimientos para este sistema es poder medir la presión en cada intersección. Es aquí donde se hace uso de la tela impregnada con la mezcla resistiva. Al mismo sensor descrito antes, se le añade una capa de tela resistiva entre los electrodos.

Figura 11. Sensor con capa de resistencia variable a la presión



Al hacer la lectura de este sensor se obtienen valores continuos, correspondientes a la presión sobre cada intersección de electrodos.

Figura 12. Esquema eléctrico de intersección con RVP

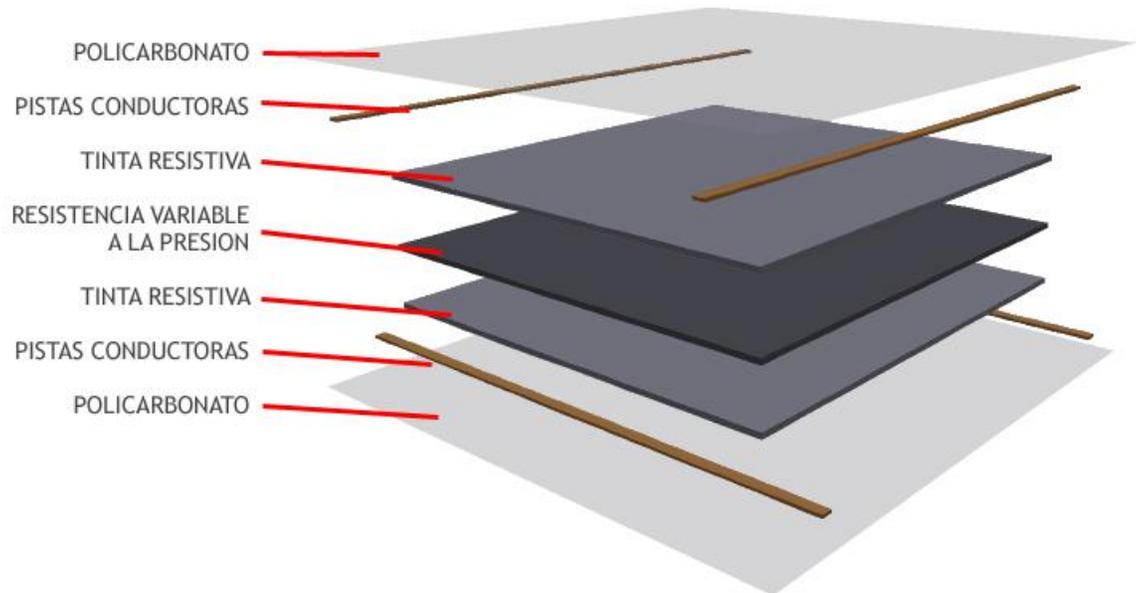


Se describen los dos estados, sin presión y con el máximo de presión.

Con esta configuración cada intersección actúa como un sensor de presión. Según esto es fácil ver que la resolución del sistema está dada por la cantidad de intersecciones por unidad de área. El sistema requiere una resolución adecuada al objeto más pequeño que hará contacto con la superficie. En el caso del dedo humano, tener intersecciones cada 0.5 cm proporciona una resolución adecuada. Sin embargo este mismo sistema no será adecuado para reconocer la presión ejercida por la punta de un stylus. Inicialmente una solución a este inconveniente es reducir la distancia entre electrodos, de esta forma se reduce la distancia entre intersecciones. Pero para mantener el área activa del sensor del mismo tamaño la cantidad de salidas del sensor aumenta drásticamente. De la misma forma si se desea aumentar el área del sensor se requiere de más electrodos y más salidas a procesar.

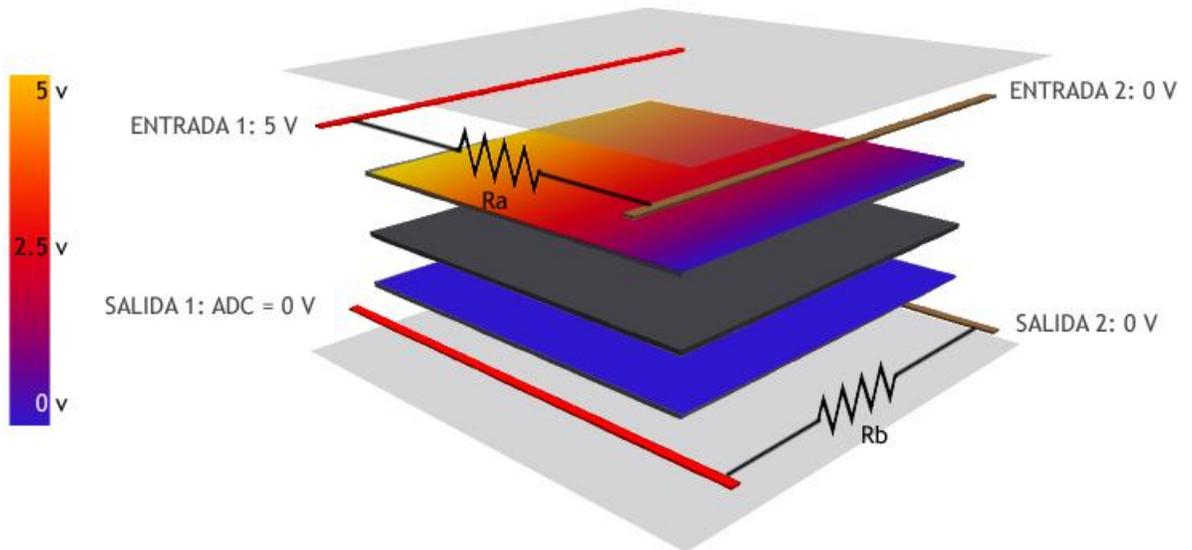
Para lograr capturar información en los espacios entre intersecciones se plantea aplicar material resistivo entre los electrodos. Es más fácil ver el proceso de lectura del sensor observando una sola celda, es decir un sensor con dos electrodos verticales y dos horizontales formando cuatro intersecciones.

Figura 13. Sensor con RVP y tinta conductiva



Los electrodos verticales serán denominados como electrodos de entrada. Los electrodos horizontales serán denominados como electrodos de salida o medición. El proceso de lectura inicia en la intersección entre el electrodo de entrada 1 y el electrodo de salida 1. En este momento el electrodo de entrada 1 está a 5 voltios y el electrodo de entrada 2 está a 0 voltios. Al haber material resistivo entre ambos electrodos se genera un gradiente de voltaje entre estos logrando de esta forma que cualquier punto de contacto entre ambos electrodos, contenga información sobre la distancia del punto de contacto entre los electrodos. En la capa inferior se conecta el electrodo de salida 1 a un adc y el electrodo de salida 2 se lleva a 0 voltios. Esto lleva la lectura del adc a 0 voltios excepto cuando existe un punto de contacto con la superficie. Este sistema genera el siguiente circuito.

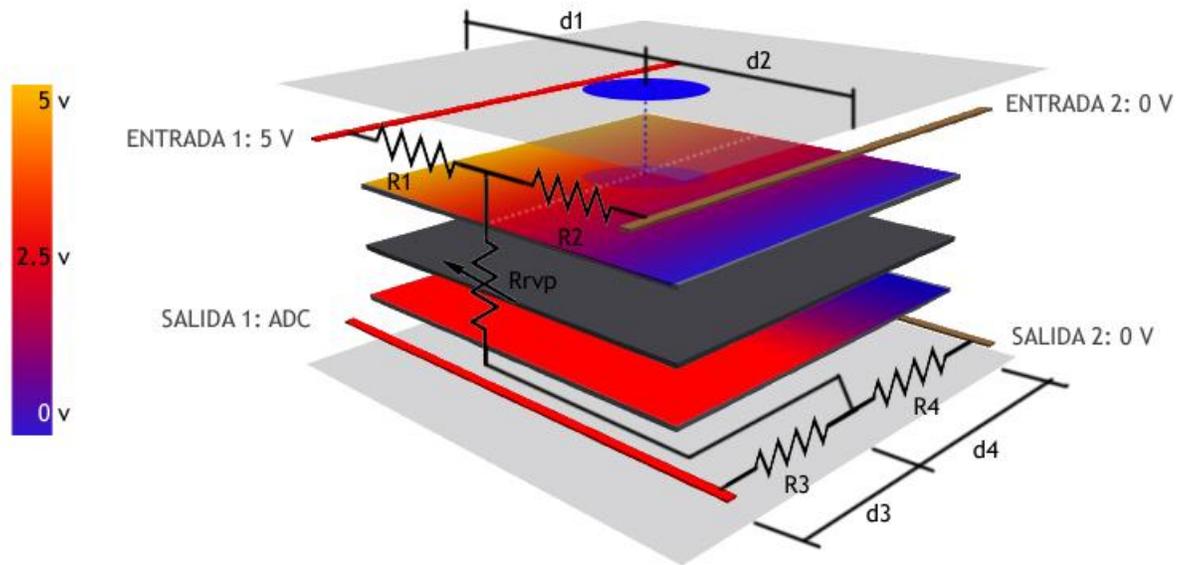
Figura 14. Diagrama sin toque



Se crea un gradiente de voltaje en la capa de tinta resistiva entre los electrodos, llevando así la lectura del adc para este punto a 0 voltios.

Posteriormente se activa el electrodo de entrada 2 y el electrodo de entrada 1 se lleva a 0 voltios. El electrodo de lectura 1 sigue conectado al adc y el electrodo de lectura a 0 voltios. Después se repite el proceso pero activando el electrodo de entrada 2 y llevando el electrodo de entrada 1 a 0 voltios. El proceso se repite iterando sobre cada intersección, llevando el electrodo de entrada a 5 voltios, el electrodo de salida al adc y todos los demás a 0 voltios. Una vez se obtienen los valores para las 4 intersecciones es posible interpolar la posición del punto de contacto entre las 4 intersecciones por medio de la interpolación bilineal.

Figura 15. Diagrama con toque



Estado en el momento de la lectura de la intersección entre la entrada 1 y salida 1 con un punto de contacto representado por el punto azul.

Se tiene que

$$R_A = R_1 + R_2 \quad R_B = R_3 + R_4 \quad R_{rvp} \propto \frac{1}{F_{toque}} \quad R(d) = R_{tinta} * d$$

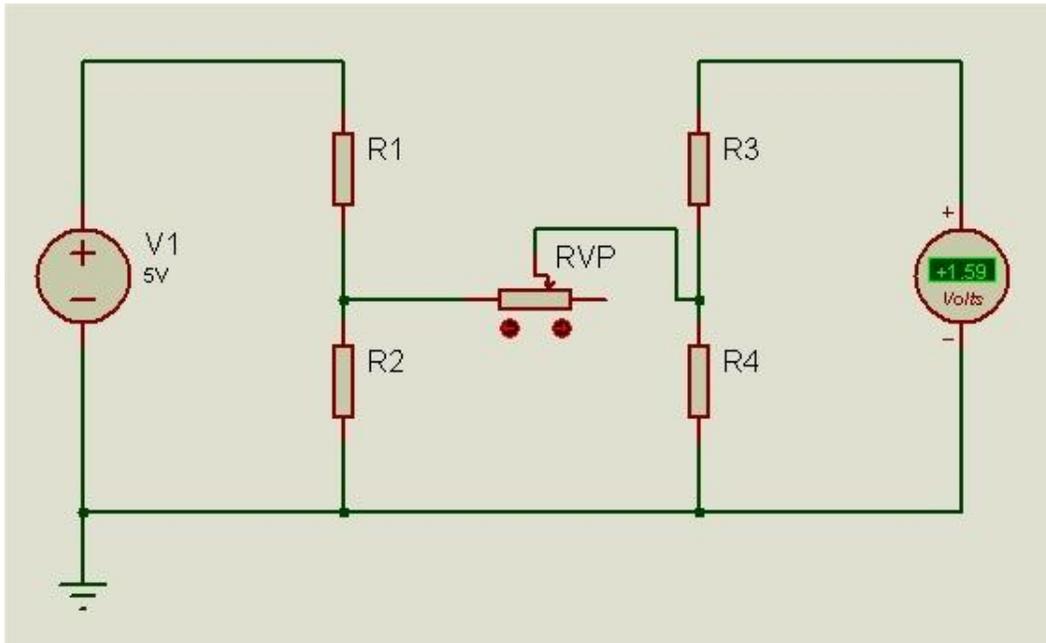
$$d_A = d_1 + d_2 \quad d_B = d_3 + d_4$$

R_1, R_2, R_3 y R_4 son función de la distancia entre electrodos donde se produce el punto de contacto y de la resistencia por metro del compuesto utilizado.

$$R_1 = R(d_1) \quad R_2 = R(d_2) \quad \text{por lo tanto} \quad R_A = R(d_1) + R(d_2)$$

$$R_3 = R(d_3) \quad R_4 = R(d_4) \quad \text{por lo tanto} \quad R_B = R(d_3) + R(d_4)$$

Figura 16. Esquema eléctrico intersección con toque



El modelo de este circuito será de la siguiente forma

$$V_{out} = \frac{R4}{(R_{rvp} + R4)} * \frac{R2}{(R1 + R2)} * V_{in}$$

Sujeta a

$$R_{rvp} + R4 \gg R2$$

Reemplazando los valores de $R1, R2$ y $R4$ ($R3$ no tiene efecto sobre el sistema) obtenemos

$$V_{out} = \frac{R(d4)}{(R_{rvp}(p) + R(d4))} * \frac{R(d2)}{(R(d1) + R(d2))} * V_{in}$$

Donde

$p =$ presión ejercida en el punto de contacto

$d1 =$ distancia desde el electrodo energizado al punto de contacto

$d2 =$ distancia desde el electrodo a tierra en la capa de entrada al punto de contacto

$d4 =$ distancia desde el electrodo a tierra en la capa de salida al punto de contacto

$$R_{rvp}(p) = 55,000 - 155,000 * p$$

De esta forma obtenemos el modelo para una intersección en el sensor en función de la presión y las distancias d1, d2 y d4.

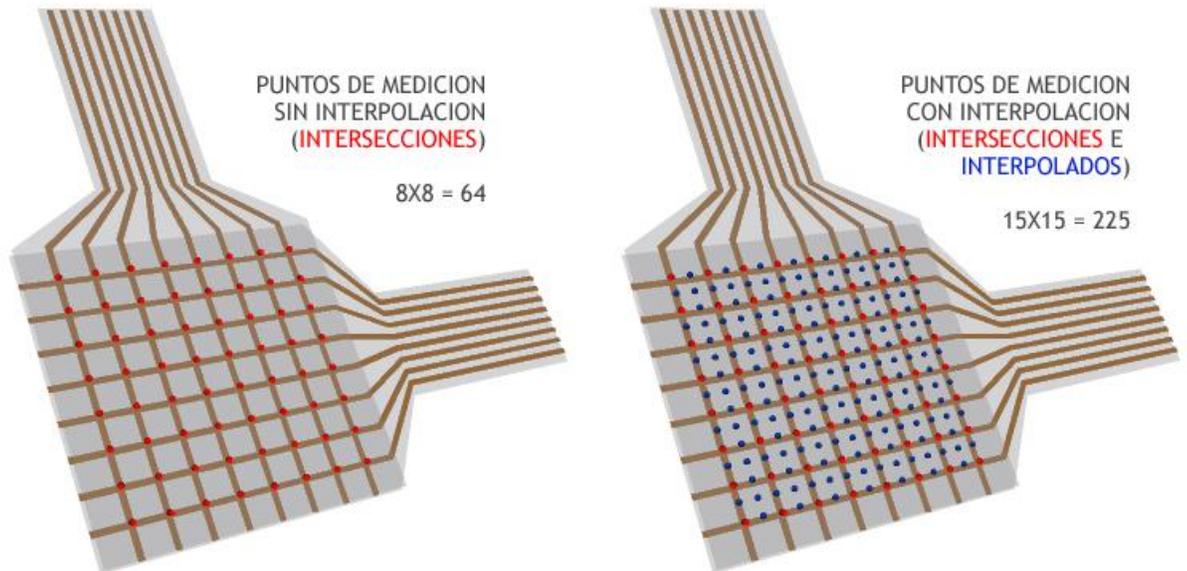
$$V_{interseccion}(p, d1, d2, d4) = \frac{R(d4)}{(R_{rvp}(p) + R(d4))} * \frac{R(d2)}{(R(d1) + R(d2))} * V_{in}$$

En las secciones posteriores se presentará un método de calibración para el sensor. Este método prácticamente elimina la necesidad de determinar valores específicos para las resistencias en el modelo pues no trabaja sobre un marco de referencia establecido, sino sobre diferencias en los estímulos de éste. Los valores de la tinta resistiva y de la resistencia variable a la presión solo condicionan el consumo de potencia del sensor.

Esta celda compuesta por 4 intersecciones actúa de la misma forma de un touchpad convencional. Permite detectar la posición y presión de un punto de contacto entre las 4 intersecciones. Este factor determina la distancia mínima entre electrodos pues cada celda podrá detectar máximo un punto de contacto. Para tener una aplicación multitoque con este tipo de celda se debe garantizar que el área de la celda sea igual o menor al área del objeto que genera el toque. Es por esto que para esta aplicación se determina que las celdas deberán ser de 1 cm por 1 cm, tamaño que representa el área transversal del dedo humano promedio.

Una vez determinado el tamaño de cada celda se procede a generar un arreglo en dos dimensiones del tamaño deseado. En este caso se genera un sensor con 8 electrodos de entrada y 8 de salida, espaciados a 1 cm entre sí. De esta manera se obtiene un sensor de 64 celdas. Gracias a la interpolación calculada en el software con 8 líneas de entrada y 8 de salida se obtienen 15 valores de entrada y 15 de salida para un total de 225 medidas de presión sobre un área de 64 cms².

Figura 17. Diagrama de posiciones interpoladas



Se muestran las posiciones medidas directamente y las interpoladas en un sensor de 8x8 electrodos.

8.2 INTERFAZ

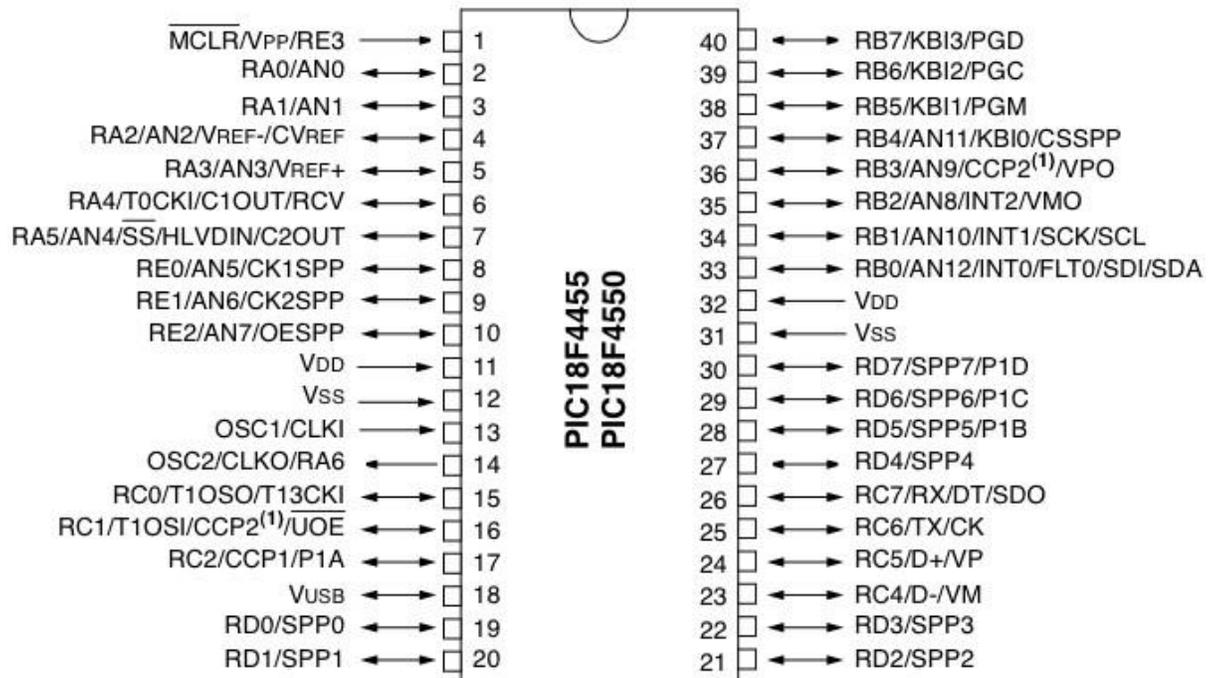
La interfaz a utilizar será la interfaz USB 2.0 de alta velocidad. Esta trabajará sobre el driver genérico Human Interface Device (HID) lo facilita la comunicación con el PC. Se utiliza la interfaz USB por su velocidad, no requiere fuente externa, facilidad de uso para el usuario final y portabilidad entre PCs modernos que ya no manejan puertos seriales o paralelos.

El estándar USB HID permite tener dispositivos de interfaz humana, ya sean mouse, teclados, game pads, tabletas, etc los cuales cumplen la función de entregar información al pc. El protocolo de USB HID permite tener una tasa de transferencia máxima teórica de 64kb por segundo. En esta aplicación al tener una imagen de 64 bytes, el micro controlador lee la imagen de 64 bytes y la entrega como un reporte de entrada al PC. Estos reportes son especificados por el protocolo USB y el tamaño máximo es de 64 bytes. En caso de transmitir una imagen de mayor dimensión esta se entregará en porciones en cada reporte.

Para esta interfaz se puede trabajar con cualquier micro controlador que tenga comunicación USB. En este caso se diseñará sobre el Micro controlador PIC

18f4550 de Microchip trabajando a velocidad 2.0 con un cristal de 20 MHz La comunicación USB está controlada por el modulo USB interno del micro controlador. Se utilizan 8 líneas digitales como las líneas para las entradas del sensor y 8 entradas con convertidor análogo a digital. En el caso de este integrado se pueden aumentar las líneas de entrada a 16 y las líneas de lectura a 13 ya que este es el máximo de líneas digitales y análogas disponibles en el PIC. Este límite máximo de líneas en este pic permitiría tener una superficie para uso con el dedo humano de cerca de 13 x 16 cms y una resolución de 25x31 pixeles. Este micro controlador contiene todo lo necesario para realizar la interfaz y lectura del sensor. Basta de un condensador de 0.22µf y un cable para conexión USB para su funcionamiento. De esta forma el micro controlador hace un barrido por cada intersección del sensor en el cual en un solo instante se encuentra una columna con 5 voltios y una fila leyendo por medio del ADC interno de 10 bits del micro controlador. La lógica de funcionamiento se puede observar en el código en lenguaje C para el micro controlador.

Figura 18. Diagrama micro controlador propuesto



Fuente: Microchip. "PIC 18f4550 Datasheet" [En línea]
<http://ww1.microchip.com/downloads/en/devicedoc/39632c.pdf>, pag. 2[Octubre 2009]

El micro controlador solo requiere de conexiones a Vdd y Vss en los pines 11,12,31,32, el oscilador en los pines 13 y 14, un condensador a Vdd en el pin 18 y 2 líneas de comunicación USB en los pines 23 y 24.

El protocolo USB es manejado por el modulo USB del micro controlador y la comunicación es dirigida por la librería HID de mikroelectronica, disponible de manera gratuita para compilar programas de menos de 2kb en tamaño.

Esta librería ofrece 3 rutinas para trabajar con USB.

Hid_Enable(**unsigned** readbuff, **unsigned** writebuff);

Esta rutina habilita la comunicación USB y prepara los buffers de escritura y lectura.

Hid_Read(**void**);

Con esta rutina se lee la información enviada desde el PC al pic vía USB.

Hid_Write(**unsigned** writebuff, **unsigned short** len);

Se realiza la escritura al buffer de escritura y se envía el largo len en bytes desde el PIC al PC.

El código que corre en el PIC trabaja bajo un sencillo código de iteración sobre las intersecciones de la matriz. En cada iteración lleva todos los pines a tierra excepto el de salida y el de lectura. Realiza la lectura y envía la información vía USB al PC.

Figura 19. Código de lectura del sensor

```
• //definir punteros a los pines
- bit PinSalida[] = {&PORTD.0,&PORTD.1,&PORTD.2,&PORTD.3,&PORTD.4,&PORTD.5,&PORTD.6,&PORTD.7};
• bit ConfigPinLectura[] = {&TRISA.0,&TRISA.2,&TRISA.3,&TRISA.4,&TRISA.5,&TRISA.6,&TRISA.7};
• char PinLectura[] = {0,1,2,3,4,5,6,7};
•
• //loop infinito
150 while(1){
•
• //limpiar el buffer
• for (i=0;i<BUFFER_SIZE;i++){
•     userWR_buffer[i] = 0;
• }
•
• //Iterar sobre la matriz
• t=0;//definir contador en 0 para almacenar en vector
• for (x=0;x<8;x++){ //cada fila
160     PinSalida[x] = 1;
•     for (y=0;y<8;y++){ //cada columna
•         //Poner pin lectura como entrada
•         ConfigPinLectura[y] = 1;
•         //leer pin adc
•         tmp = ADC_Read(PinLectura[y]);
•         //escribir valor en el buffer de escritura USB
•         userWR_buffer[t] = tmp;
•         t++;
•         //Poner pin lectura como salida
170         ConfigPinLectura[y] = 0;
•     }
• }
• while (!HID_Write(&userWR_buffer, BUFFER_SIZE));
• }
```

El resultado obtenido con este módulo de adquisición de datos es un rendimiento excelente en el cual se le entrega a la aplicación la estructura de datos especificada un promedio de 540 veces por segundo. Al comparar este rendimiento con 30 cuadros por segundo que entrega una cámara web estándar vemos un claro aumento de rendimiento sobre otros métodos de adquisición de información.

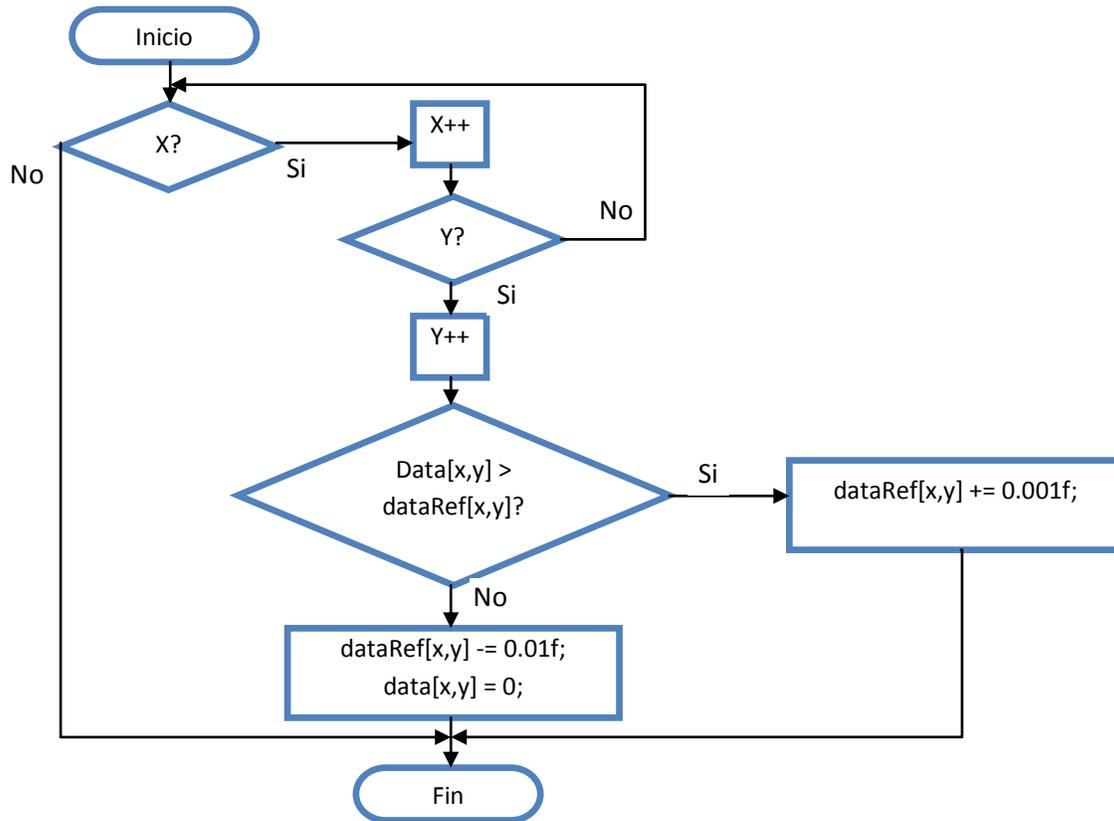
8.3 PROCESAMIENTO

El procesamiento de la estructura de datos recibida se realiza en una aplicación realizada en C# corriendo sobre un PC Windows XP. Esta aplicación recibe un promedio de 540 cuadros por segundo los cuales son analizados con la siguiente secuencia.

8.3.1 Calibración

Ya que se tiene una gran cantidad de celdas trabajando juntas es muy factible que se presenten irregularidades entre celdas en el momento de la lectura por la calibración de cada una. Inicialmente se supone que cada celda debe tener una construcción perfecta para asegurar una calibración idéntica a las de las demás y que no habrá fluctuaciones por cambios de temperatura. Sin embargo otro enfoque presenta una mejor solución al problema de calibración. El modulo de procesamiento debe almacenar un arreglo de las mismas dimensiones del arreglo leído. En este arreglo se almacena el valor que determina que esa celda no está activa. Si el valor leído es mayor al valor de referencia, se determina que la celda está activa y hay un toque. Esto produce un aumento pequeño en el valor de referencia. Igualmente si el valor leído es inferior al valor de referencia se determina que la celda no está activa y se le resta una pequeña cantidad a éste valor de referencia. De esta forma cada celda está en constante búsqueda de equilibrio. Este comportamiento se asemeja al de la piel humana que si permanece en contacto con algo por un periodo de tiempo extendido, ya no es tan sensible a dicho estímulo. Este método de calibración logra sobreponerse a la gran cantidad de inconsistencias presentes en la arquitectura del sensor como la equivalencia entre celdas, cambios de temperatura, etc. Pero se debe tener en cuenta que si se aplica un estímulo idéntico por largos periodos de tiempo eventualmente el sensor no percibirá el estímulo pues lo ha calibrado como un cero. Es ideal que este proceso se lleve a cabo en el micro controlador para reducir la carga en el PC.

Figura 20. Diagrama de flujo del algoritmo de calibración



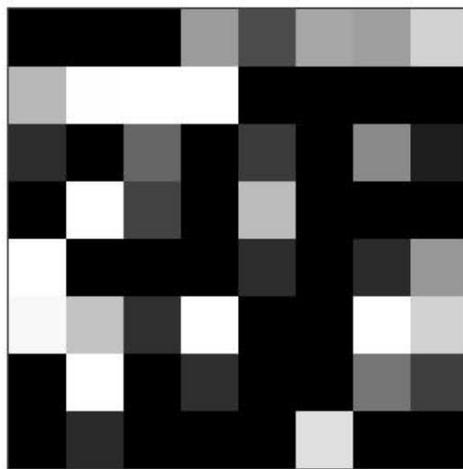
8.3.2 Interpolación

La información obtenida será sometida a interpolación bilineal para obtener puntos intermedios entre los puntos medidos del sensor. Por medio de este algoritmo se aumentará la resolución del sistema de 64 a 225 mediciones en el caso de un sensor de 8x8 electrodos.

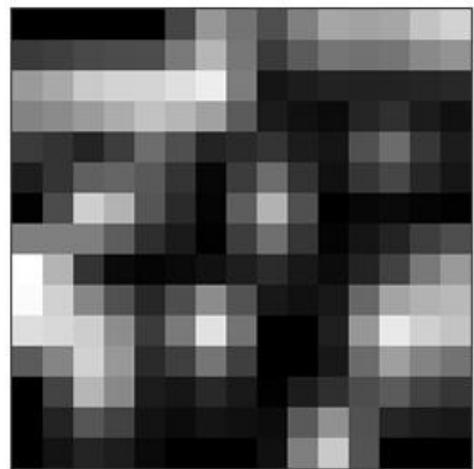
Cualquier punto entre el área definida por un cuadrado unitario puede ser estimada por medio de la formula de interpolación bilineal.

$$f(x,y) = f(0,0)(1-x)(1-y) + f(1,0)x(1-y) + f(0,1)(1-x)y + f(1,1)xy$$

Figura 21. Resultado interpolación



MEDICIONES SENSOR 8X8
(RAW DATA)

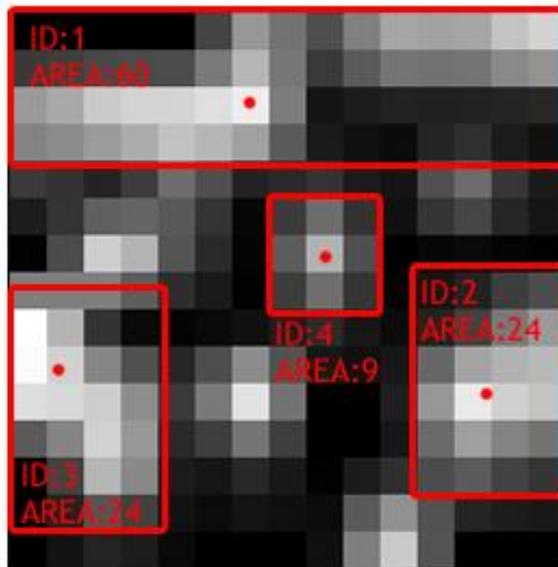


MEDICIONES SENSOR
INTERPOLADO A 15X15

8.3.3 Extracción de regiones

Cada región donde se detecta un punto de contacto es analizada para conectividad, masa, centroide

Figura 22. Resultado etiquetado



El área es calculada en pixeles con la siguiente formula

$$AreaRegion = pixelesAlto \times pixelesAncho$$

El centro es calculado geoméricamente

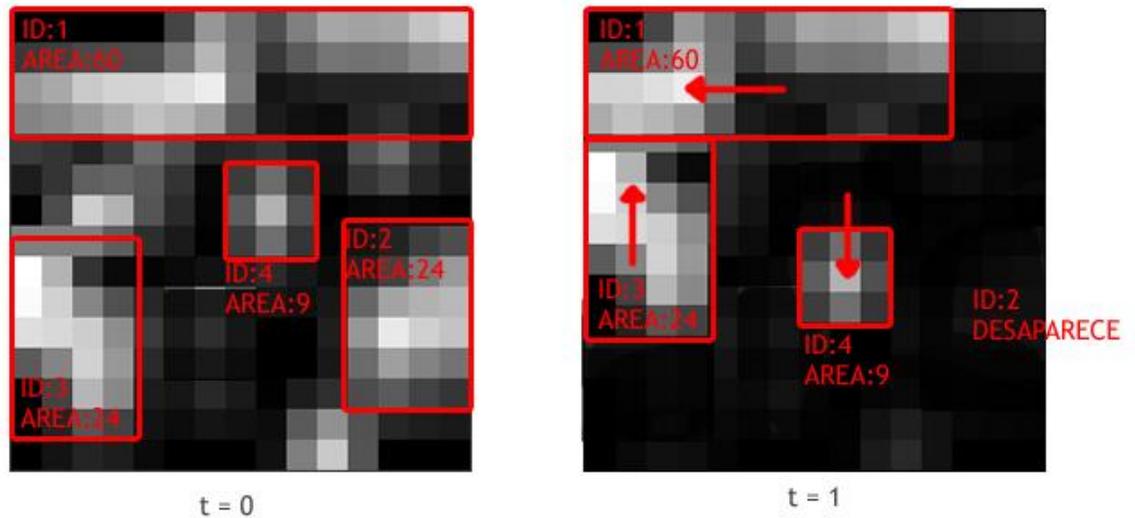
$$Centro_x = Round\left(\frac{pixelesAncho}{2}\right)$$

$$Centro_y = Round\left(\frac{pixelesAlto}{2}\right)$$

8.3.4 Sincronización

Cada región extraída es comparada con las regiones del cuadro anterior para así darle identificación a cada región, velocidad, inicio y final.

Figura 23. Proceso de identificación de regiones



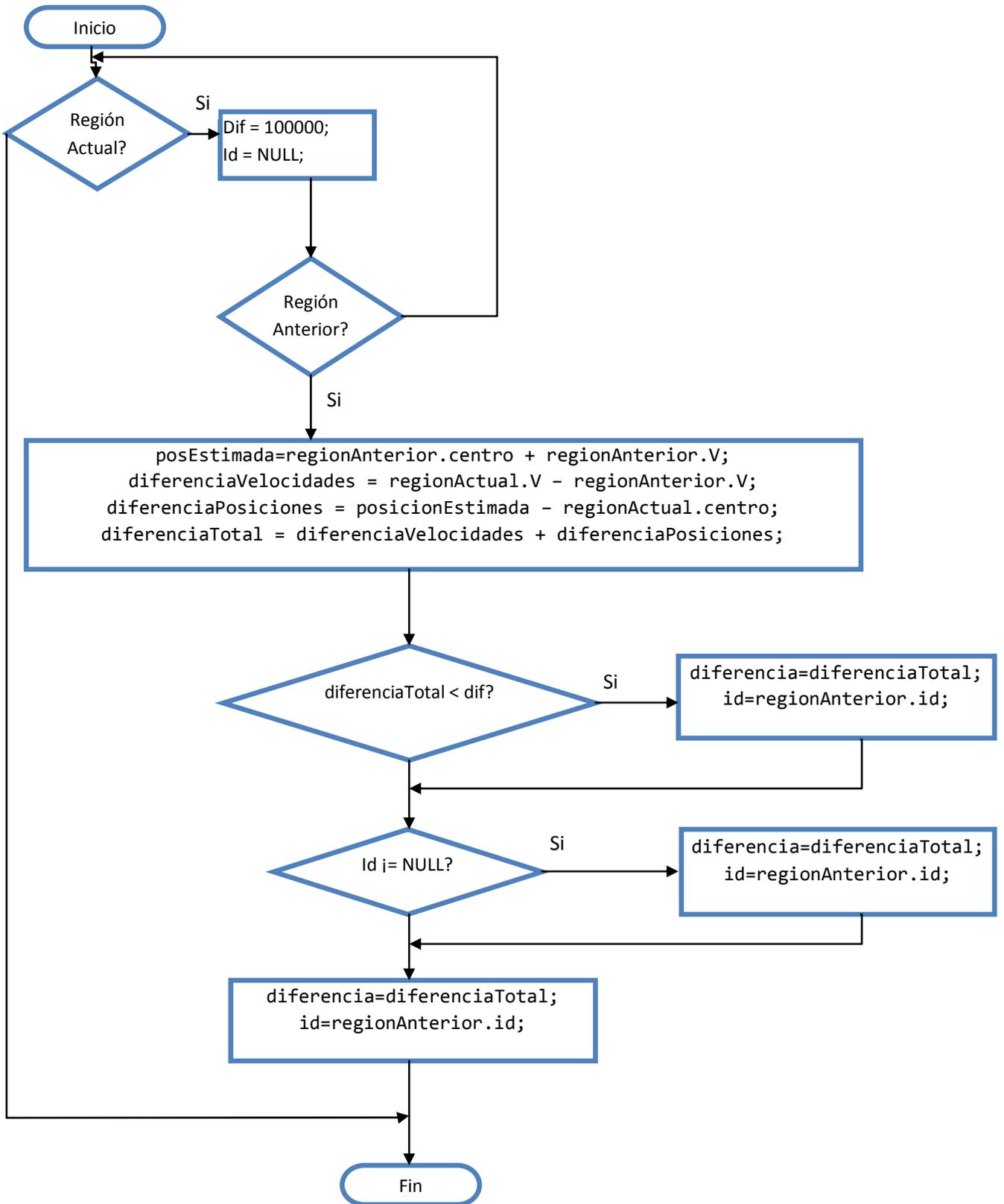
Con cada cuadro recibido las regiones son etiquetadas y el centro, posición y área de cada región es calculada. Al recibir esta información el algoritmo de identificación asigna la identidad de la región más cercana en el cuadro anterior. En el momento en que el cuadro anterior tiene más regiones que el cuadro actual se presenta el momento en el que el toque pierde contacto con la superficie. Cuando hay más regiones en el cuadro actual quiere decir que hay un nuevo punto de contacto.

La velocidad es calculada geoméricamente

$$V_x = Centro_x(t) - Centro_x(t - 1)$$

$$V_y = Centro_y(t) - Centro_y(t - 1)$$

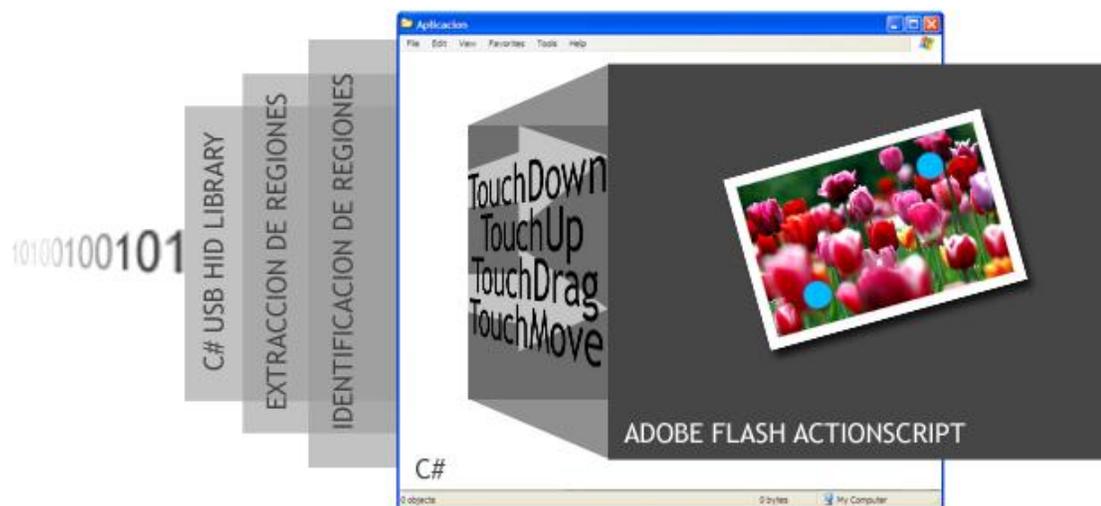
En el algoritmo de identificación se usa el vector de velocidad para darle prioridad a las regiones que estén en el camino previsto por este. Este proceso es mejor explicado con el siguiente pseudocódigo. Se itera sobre cada región actual para compararla con cada región anterior. Las diferencias en velocidad y posición estimadas son calculadas y de acuerdo a esto se selecciona la región con la menor diferencia posible.



8.4 APLICACIÓN

La aplicación final (interfaz grafica) corre sobre la aplicación de procesamiento (lectura del sensor) embebida como un control ocx. De esta forma los eventos multitoque que dispara esta podrán ser interpretados por la aplicación final. De lo contrario las demás aplicaciones y/o sistema operativo no sabrán cómo interpretar la información entregada por el sensor ya que este tipo de interfaz humano máquina no se encuentra estandarizada.

Figura 24. Flujo de información desde el circuito hasta la aplicación



La información se recibe con la librería USB HID, posteriormente se extraen las regiones y se identifican. Con esta información se envían los eventos al archivo swf embebido dentro de la aplicación madre en C#.NET.

Esta aplicación podrá ser programada en Adobe Flash para agilizar el desarrollo y podrá correr dentro de la aplicación madre en C#. La comunicación entre estas se realizara por medio de mensajes en javascript que permiten la comunicación en ambas vías entre los dos lenguajes.

La forma en que C# envía mensajes al archivo SWF es con la función setVariable del modulo ocx. Este método cambia la variable en el archivo flash directamente.

```
miPeliculaFlash.SetVariable("miVariableParaFlash", "miValor");
```

Para enviar mensajes desde flash a C# se hace uso de la función `fscommand`

```
fscommand("miVariableParaCS","miValor");
```

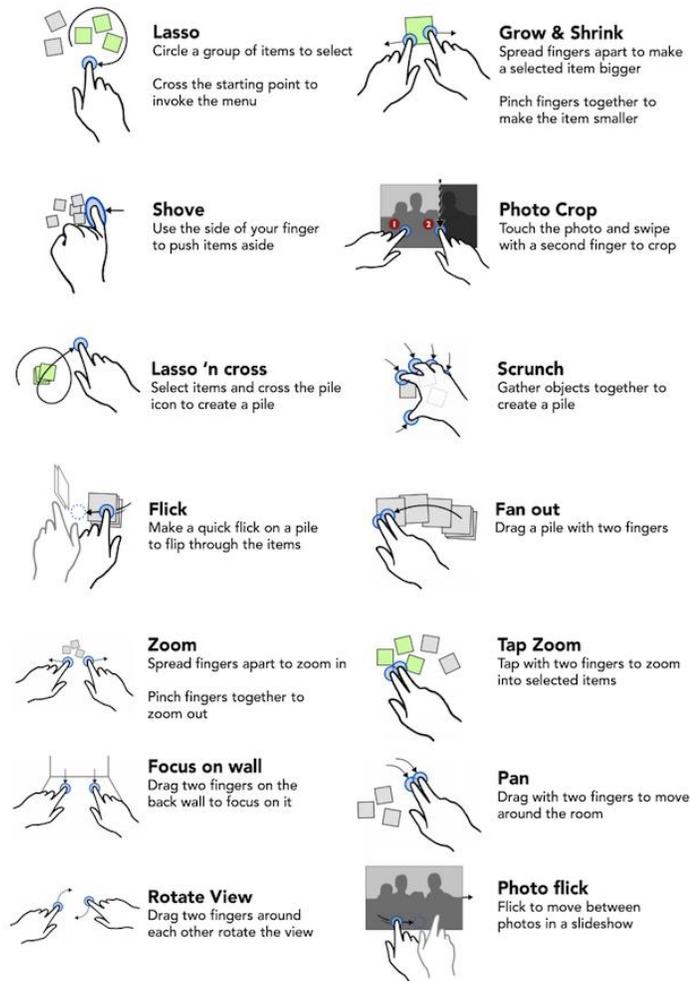
Cada vez que esta función es llamada C# dispara la función handler de esta y procesa la información recibida.

En este caso nos interesa es la comunicación C# -> Flash pues C# enviará toda la información recibida del sensor ya procesada como eventos al archivo swf. Este tendrá las funciones correspondientes a cada lanzamiento del evento. Si en tal caso C# detecta el movimiento de dos puntos de contacto, disparará el evento `TouchMove`.

```
miPeliculaFlash.SetVariable("events",  
    "<touchMove id=21 x=34 y=28><touchMove id=9 x=324 y=658>");
```

Se envía una estructura XML para toda la información de eventos a la variable *events*, Flash procesará esta información disparando los eventos correspondientes a la aplicación generada. Con base en esto, se puede ver que el sistema presenta cierta flexibilidad al momento de interpretar la información obtenida. Para aplicaciones cotidianas se han establecido ciertos patrones para usar este tipo de sensor. Estos son llamados gestos y pueden ser creados por el mismo desarrollador. A continuación se presentan los gestos más utilizados en los sistemas actuales.

Figura 25. Listado de Gestos Básicos



Fuente: Nxpshop. “BumpTopd 3D Desktop gets unique multi-touch gestures” [En línea] <http://www.nxpshop.com/index.php/2009/10/bumptop-3d-desktop-gets-unique-multi-touch-gestures/> , [Enero 2010]

9 RESULTADOS

Los subsistemas planteados presentan un comportamiento esperado. Cada uno de estos existe y su funcionamiento ha sido probado en diversas aplicaciones en la industria mas no trabajando en unisonó para el propósito planteado en este documento. Sin embargo dada la complejidad del sistema total y la gran cantidad de formas de interactuar con el sensor es difícil obtener mediciones simuladas con base en las simulaciones de los subsistemas. Es decir, si se realiza una unión entre los resultados de las simulaciones de cada subsistema no se logra un resultado simulado para el sistema global como se lograría, por ejemplo, con sistemas mecánicos.

Se propone realizar una simulación basada en el objetivo final del sistema. Se plantea la tarea de referencia sobre la cual se realizó el benchmark donde el rotar, escalar y trasladar una imagen en un PC tardaba en el mejor de los casos 7 segundos. En la forma en que es diseñado el sensor, se busca tener una interacción más intuitiva y natural con el PC utilizando gestos y movimientos naturales para una persona. Según el diseño propuesto, para realizar la tarea de referencia con este sistema basta con posicionar dos dedos sobre la superficie. El gesto de aumentar la distancia entre ambos dedos genera un proceso de escalado. El gesto de posicionar un dedo sobre el centro de rotación y desplazar el otro genera un proceso de rotación. El gesto de desplazar ambos dedos en la misma dirección genera un proceso de traslación. Es fácil ver que estos tres gestos pueden ser realizados a la vez por lo cual es seguro decir que para trasladar, rotar y escalar una imagen con este sistema, es necesario el tiempo requerido para mover dos dedos sobre una superficie sin fricción ni restricciones.

Para esto se le pidió al mismo grupo de usuarios que establecieron el benchmark inicial que desplazaran y rotaran (La naturaleza solida del papel impide realizar el proceso de escalado en el mundo físico) una fotografía real, empezando desde posiciones y rotaciones aleatorias hasta una posición central y una rotación de 45 grados, solo con la yema de dos dedos. Los resultados fueron los siguientes:

Tabla 10 - Tiempos de Rotación y traslación para usuarios BASICOS

| Tiempo(s) | Prueba # | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | Promedio Usuario(s) |
|-----------|-------------------------|-----------|-----|----------------------|-----|-----|-----|--------------------------|-----|---------------------|
| | | Usuario 1 | 2.5 | 4.1 | 4.2 | 2.6 | 3.7 | 4.6 | 3.2 | 2.7 |
| | Usuario 2 | 4.1 | 2.0 | 4.7 | 4.3 | 3.4 | 2.4 | 4.2 | 3.8 | 3.6 |
| | Usuario 3 | 3.0 | 4.2 | 2.1 | 2.3 | 4.6 | 2.2 | 4.5 | 4.5 | 3.4 |
| | Usuario 4 | 2.4 | 2.1 | 4.6 | 3.8 | 4.7 | 3.3 | 3.4 | 3.7 | 3.5 |
| | Usuario 5 | 3.7 | 3.6 | 2.9 | 4.2 | 2.6 | 3.9 | 2.9 | 3.1 | 3.4 |
| | Usuario 6 | 3.7 | 4.1 | 2.5 | 2.6 | 2.6 | 3.0 | 2.2 | 3.6 | 3.0 |
| | Usuario 7 | 4.6 | 2.9 | 4.1 | 1.9 | 4.6 | 4.1 | 2.3 | 3.2 | 3.5 |
| | Usuario 8 | 4.2 | 4.0 | 2.5 | 4.6 | 4.7 | 2.1 | 3.0 | 3.1 | 3.5 |
| | Tiempo Minimo(s) | | 1.9 | Tiempo Max(s) | | 4.7 | | Promedio Total(s) | | 3.4 |

Tabla 11 - Tiempos de Rotación y traslación para usuarios PROMEDIO

| Tiempo(s) | Prueba # | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | Promedio Usuario(s) |
|-----------|-------------------------|-----------|-----|----------------------|-----|-----|-----|--------------------------|-----|---------------------|
| | | Usuario 1 | 4.6 | 4.1 | 2.8 | 2.2 | 3.6 | 2.6 | 2.7 | 5.1 |
| | Usuario 2 | 3.0 | 3.8 | 3.5 | 2.8 | 2.3 | 1.9 | 1.9 | 3.2 | 2.8 |
| | Usuario 3 | 2.0 | 4.0 | 2.6 | 2.9 | 2.7 | 3.6 | 2.1 | 2.1 | 2.8 |
| | Usuario 4 | 2.7 | 2.8 | 4.9 | 3.4 | 3.1 | 2.5 | 4.4 | 1.8 | 3.2 |
| | Usuario 5 | 3.8 | 5.1 | 4.0 | 4.2 | 1.8 | 4.7 | 2.9 | 2.5 | 3.6 |
| | Usuario 6 | 4.9 | 1.8 | 4.7 | 5.0 | 4.0 | 3.4 | 4.7 | 2.6 | 3.9 |
| | Usuario 7 | 2.7 | 3.0 | 5.1 | 2.1 | 2.2 | 2.8 | 4.4 | 3.8 | 3.3 |
| | Usuario 8 | 3.6 | 2.4 | 3.6 | 3.8 | 4.4 | 4.4 | 1.8 | 2.6 | 3.3 |
| | Tiempo Minimo(s) | | 1.8 | Tiempo Max(s) | | 5.1 | | Promedio Total(s) | | 3.3 |

Tabla 12 - Tiempos de Rotación y traslación para usuarios AVANZADOS

| Tiempo(s) | Prueba # | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | Promedio Usuario(s) |
|-----------|-------------------------|-----------|-----|----------------------|-----|-----|-----|--------------------------|-----|---------------------|
| | | Usuario 1 | 2.8 | 2.4 | 3.3 | 3.3 | 1.9 | 3.4 | 2.1 | 4.1 |
| | Usuario 2 | 4.5 | 3.5 | 4.2 | 3.9 | 3.5 | 1.8 | 2.0 | 2.0 | 3.2 |
| | Usuario 3 | 2.9 | 2.5 | 4.1 | 1.9 | 2.7 | 2.9 | 1.9 | 4.7 | 3.0 |
| | Usuario 4 | 4.5 | 3.3 | 3.7 | 4.7 | 2.5 | 1.8 | 2.3 | 3.4 | 3.3 |
| | Usuario 5 | 2.4 | 3.7 | 2.0 | 2.0 | 3.2 | 3.7 | 1.8 | 3.1 | 2.7 |
| | Usuario 6 | 4.0 | 4.6 | 4.2 | 1.9 | 2.7 | 3.1 | 2.6 | 3.0 | 3.3 |
| | Usuario 7 | 1.9 | 4.5 | 2.1 | 2.9 | 2.8 | 4.2 | 3.3 | 3.0 | 3.1 |
| | Usuario 8 | 2.9 | 3.6 | 4.2 | 4.4 | 2.5 | 4.1 | 3.1 | 2.2 | 3.4 |
| | Tiempo Minimo(s) | | 1.8 | Tiempo Max(s) | | 4.7 | | Promedio Total(s) | | 3.1 |

Los resultados de la simulación de interacción presentan unos tiempos drásticamente menores respecto al benchmark inicial. Adicionalmente se obtiene un tiempo promedio de ejecución muy similar para usuarios básicos, promedios y avanzados lo cual hace que cualquier persona independiente de su formación tecnológica pueda interactuar con un PC de forma más intuitiva, natural y eficiente.

10 APLICACIONES

Este tipo de sensor permite entregarle al PC más cantidad de información e instrucciones que con cualquier otro método actual de interacción humano-máquina. Inicialmente se plantea el uso de este como interfaz de control en el cual el usuario por medio de toques maneja su PC. Sin embargo es fácil ver que los beneficios de este sensor pueden ser beneficiosos para otro tipo de aplicaciones.

9.1 APLICACIONES MUSICALES

Ya que este sensor tiene una tasa de muestreo superior a los 500 hz se hace posible emular instrumentos musicales. Puede ser usado como la superficie de un instrumento de percusión digital, donde el sonido sintetizado por el PC es resultado de la presión ejercida por la baqueta, la posición donde golpea y la rapidez con que se genera el impacto. Este tipo de información es casi imposible de adquirir con los métodos tradicionales de interacción multitoque pues presentan tasas de muestreo mucho más bajas y no miden la presión sobre el punto de contacto.

9.2 APLICACIONES MÉDICAS

Se puede usar este sensor para medir los puntos de presión en la pisada de atletas o pacientes con problemas óseos en miembros inferiores.

9.3 APLICACIONES COMERCIALES

El uso de este tipo de sensor en aplicaciones comerciales abre las puertas a nuevos tipos de interacción, desde simples interfaces táctiles hasta pisos y muebles sensibles al entorno. Mesas de restaurantes pueden ser cubiertas con este sensor y no solo permitirle al cliente hacer pedidos desde la superficie de su mesa sino también detectar si el vaso de su bebida se encuentra lleno o vacío.

9.4 APLICACIONES EN INGENIERÍA

El uso de este sensor como piel para un robot es muy factible pues al implementarlo en un grip se podrá determinar la presión sobre el objeto manipulado y sobre las colisiones con el entorno, para así tomar decisiones de control. Al implementar el sensor como un tapete es posible determinar por medio de análisis de pasos cuantas personas transitan por un determinado lugar. Aprovechando la presión ejercida por cada cuerpo se puede determinar el peso de

la persona, infiriendo así la edad y/o contextura para usos estadísticos. De lograr una mejora en la precisión de las tintas resistivas para asegurar una medición no solo exacta sino precisa, es posible implementar el sensor para medir presiones sobre una superficie expuesta a flujo hidráulico. Por ejemplo analizar las presiones sobre la superficie del alerón de un fórmula 1.

9.5 APLICACIONES EN DISEÑO INDUSTRIAL

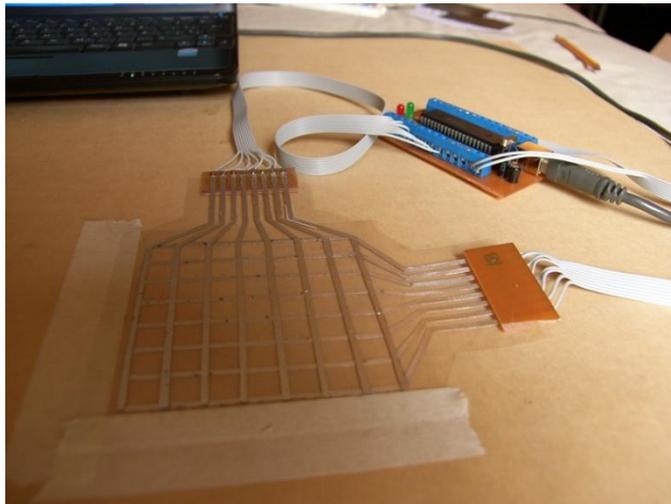
Al implementar este sensor, aprovechando su flexibilidad, sobre mangos, manijas, espaldares, etc. El diseñador industrial obtendrá una visualización clara sobre los puntos de presión que perjudican la ergonomía de su diseño.

11 CONCLUSIONES

Las simulaciones de los circuitos planteados para el sensor presentan el comportamiento esperado. A su vez el desarrollo de software para la interfaz y etapa de procesamiento es trivial. Esto permite asegurar que la integración de diversas tecnologías de una forma creativa e innovadora da como fruto este tipo de sensor.

Se prueba la factibilidad de un sensor multidimensional de baja complejidad, fácil manufactura y operación confiable. Este podrá ser mejorado al mejorar sus componentes resistivos en cuanto a linealidad y homogeneidad. Los valores de las resistencias utilizadas en el sensor dependen solo de la potencia consumida por este y no afectan la precisión del sistema. Un uso creativo de la información en el software permite obtener un sistema capaz de auto calibrarse constantemente, permitiendo así que el sensor sea inmune a un gran número de defectos de construcción que harían que cualquier otro sensor quedara inútil. Esta prueba de concepto de funcionamiento establece que es posible construir un sensor capaz de reconocer una cantidad de grados de libertad equivalente a tres veces la cantidad de celdas de toque que componen el sensor. Ya que cada celda tiene capacidad de detectar la posición de un toque y su correspondiente presión, un sensor de 8x8 celdas tendría máximo 192 grados de libertad.

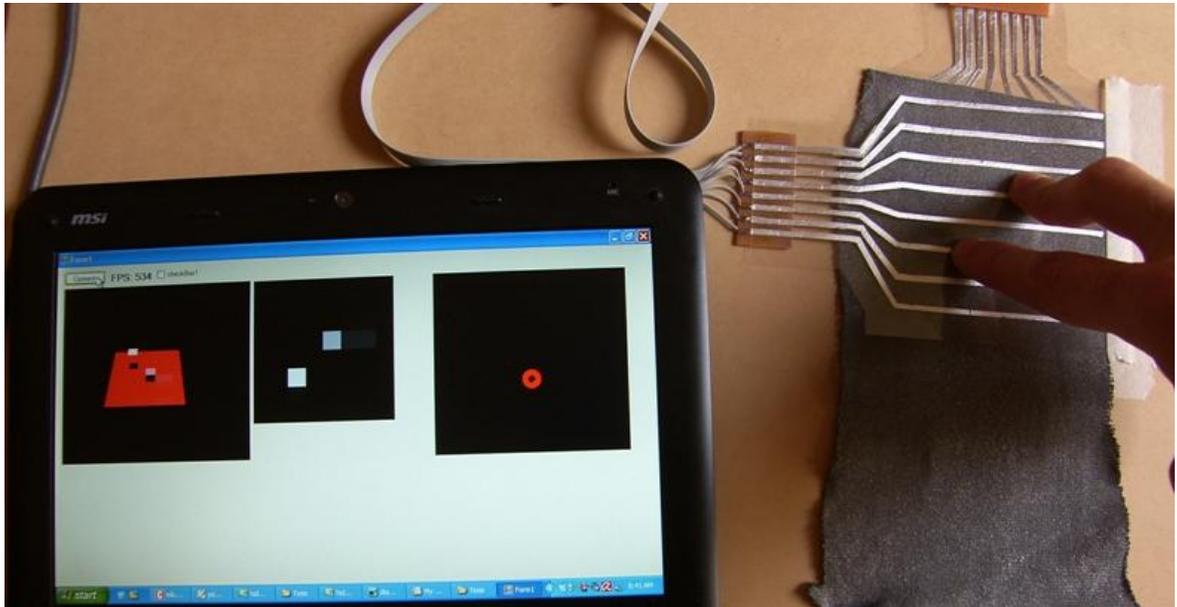
Figura 26. Prototipo para prueba de concepto



Para probar el concepto básico de funcionamiento del sensor se construyó un prototipo que permitiera observar el comportamiento de los componentes fundamentales de éste. El prototipo abarca desde la comunicación al pc vía USB 2.0 , recolectando información de una matriz de 8x8 celdas, hasta la visualización

de la información recolectada sin procesar. Con este prototipo se obtuvieron los siguientes resultados.

Figura 27. Resultados obtenidos con el prototipo



Prototipo para prueba demostrando el comportamiento esperado del sensor. Se aplican 2 toques sobre el sensor y la información sin procesar es graficada en pantalla. A la izquierda una visualización en 3D. En el centro una visualización en 2D donde el brillo representa la presión con la cual se está tocando. A la derecha está la visualización del resultado de la interpolación en la celda 0,0. Se puede apreciar que en la visualización central los puntos más intensos representan los puntos de contacto de los dedos sobre el sensor.

De acuerdo a estos resultados es fácil ver que este tipo de sistema efectivamente reduce el tiempo requerido para ejecutar tareas en un PC gracias a la mayor cantidad de información sobre la intención del usuario que puede recolectar. Esto se logra haciendo que una tarea que normalmente se logra por una secuencia de instrucciones como:

Llevar el mouse al menú -> Seleccionar opción rotar -> llevar mouse a punto de control -> hacer click sostenido -> llevar punto de control al ángulo deseado -> hacer click en "Aceptar Cambios".

Se logre ahora con un gesto más intuitivo, que no requiere de capacitación alguna pues se hace en el día a día con nuestra interacción con objetos físicos. Este gesto es un movimiento básico de girar dos dedos sobre la superficie. Para lograr esto la interfaz funciona a velocidades de muestreo superiores a la velocidad de

movimiento promedio de la mano humana por lo cual el tiempo de respuesta es inmediato para el ojo humano.

Las grandes marcas de desarrolladores están adoptando el uso de tecnologías multitoque, sin embargo aun no se ha definido un estándar de comunicación e interacción. Por ahora cada aplicación debe ser construida en su totalidad según el protocolo que mejor se adapte al requerimiento. La implementación de este sensor en diversos campos ofrece beneficios no disponibles por alternativas de tan bajo costo. Su bajo costo y alta dimensionalidad son factores que prometen infinidad de aplicaciones que algún día harán de la experiencia de interactuar con una máquina, algo tan familiar como un humano.

BIBLIOGRAFIA

Fairchild Semiconductor. "4051 Multiplexor Datasheet" [En línea]
<http://www.jameco.com/Jameco/Products/ProdDS/1218327.pdf> [Octubre 2009]

Synaptics Technologies. "Capacitive Position Sensing". [En línea]
<<http://www.synaptics.com/technology/cps.cfm>>, [Marzo 21, 2009]

Mitsubishi. DiamondTouch. [En línea]
<<http://www.merl.com/projects/DiamondTouch/>>, [Marzo 23, 2009]

MALONEY, Timothy. Electrónica Industrial Moderna. 5ª ed. México, Pearson Education, 2006. 972 p.

PEREZ GARCIA, Miguel A y ALVAREZ ANTON, Juan C. Instrumentación Electrónica. 2ª ed. Madrid, Thompson Editores, 2005. 862 p.

Han, Jeff. "Low-Cost Multi-Touch Sensing through Frustrated Total Internal Reflection" [En línea]
<<http://www.tedtalks.com>>. [Febrero 2009].

Buxton, B. "Multi-Touch Systems That I Have Known And Loved". [En línea]
<<http://billbuxton.com/multitouchOverview.html>>, [Marzo 20, 2009]

Microchip. "PIC 18f4550 Datasheet" [En línea]
<http://ww1.microchip.com/downloads/en/devicedoc/39632c.pdf>, [Octubre 2009]

Wilson, A. "PlayAnywhere: A Compact Interactive Tabletop Projection-Vision System". [En línea] <www.microsoftresearch.com>, [Febrero 2009]

RUPEREZ, José Luis. Tarjeta USB de Adquisición de Datos. En: Elektor, Noviembre, 2007, No. 329, p. 16 – 21.

Craig Peacock. "USB in a Nutshell" [En línea]
<http://www.beyondlogic.org/usbnutshell/> [Octubre 2009]

M. DE LA CRUZ GARCIA, Jesús y PAJARES, Gonzalo. Vision por Computador: Imágenes digitales y aplicaciones. 2ª ed. Madrid, RA-MA Editorial, 2008. 740 p.

Anexo A. – Código Implementado en C# para la identificación de regiones en el algoritmo de sincronización entre cuadros.

```
//regionActual es un objeto de tipo Region

//regionAnterior es un objeto de tipo Region

foreach(regionActual){
    diferencia = 10000;
    id = NULL;
    foreach(regionAnterior){
        posicionEstimada =regionAnterior.centro +
regionAnterior.V;

        diferenciaVelocidades = regionActual.V -
regionAnterior.V;
        diferenciaPosiciones = posicionEstimada -
regionActual.centro;

        diferenciaTotal = diferenciaVelocidades +
diferenciaPosiciones;

        if (diferenciaTotal < diferencia){
            diferencia = diferenciaTotal;
            id = regionAnterior.id;
        }
    }
    if (id!=NULL){
        regionActual.id = id;
        remove(regionAnterior.id);
    }else{
        throw.event(new_touch);
    }
}
```

Anexo B. – Código implementado en C# para la calibración del sensor

```
For (int x=0;x< data.GetLength(0);x++){
    For (int y=0;y< data.GetLength(0);y++){

        If (data[x,y] > dataRef[x,y]){
            //data recibida es mayor, ref debe crecer
            dataRef += (data[x,y]-dataRef[x,y])*0.01f;
        }else if (data[x,y] < dataRef[x,y]){
            //data recibida es mayor, ref debe disminuir lentamente
            dataRef += (data[x,y]-dataRef[x,y])*0.0001f;
        }

        If (data[x,y] > dataRef[x,y]){
            data[x,y] -= dataRef[x,y];
        }else if (data[x,y] < dataRef[x,y]){
            data[x,y] = 0;
        }

    }

}
```