
TÉCNICAS DE CONEXIÓN DE NODOS EN MESAS INTERACTIVAS: DISEÑO Y EVALUACIÓN

Anicia Monthzerrat Zorrilla Bustamante



UNIVERSIDAD
POLITECNICA
DE VALENCIA

Trabajo Fin de Máster

Máster en Ingeniería del Software, Métodos Formales y Sistemas de Información
DEPARTAMENTO DE SISTEMAS INFORMÁTICOS Y COMPUTACIÓN

Directores

Dr. Javier Jaén Martínez
Dr. Alejandro Catalá Bolós

Julio de 2013, Valencia

Agradecimientos

*A mi madre, mi hermana y mi hermano, por su apoyo incondicional,
porque ellos son mi fuerza y mi motivación.*

*A mis sobrinos Catalina, Carlos, Alexa y Camila,
que me alentaban en los momentos difíciles*

A Javier por su dirección y colaboración.

*A Nelly y Alejandro por su gran paciencia, sus consejos,
su guía dedicación y esfuerzo.*

*A todos aquellos amigos, que han compartido
y me han acompañado en esta etapa.*

A mis amigos que en la distancia aún están ahí.

ÍNDICE GENERAL

ÍNDICE GENERAL	V
ÍNDICE DE FIGURAS	VII
1 INTRODUCCIÓN	9
1.1 MOTIVACIÓN DEL TRABAJO	9
1.2 OBJETIVOS DEL TRABAJO	12
1.3 ESTRUCTURA DEL DOCUMENTO	13
2 TÉCNICAS PARA LA CONEXIÓN DE NODOS EN SUPERFICIE	15
2.1 CONTEXTO TECNOLÓGICO	15
2.1.1 <i>Mesas Interactivas</i>	15
2.1.2 <i>Microsoft Surface Unit</i>	18
2.2 MARCO DE TRABAJO PARA LA EDICIÓN DE GRAFOS	21
2.3 TÉCNICAS DE INTERACCIÓN	23
2.3.1 <i>Arrastre</i>	23
2.3.2 <i>Pulsación</i>	27
2.3.3 <i>Arrastre con Visor</i>	29
2.3.4 <i>Dirección</i>	34
2.3.5 <i>Proximidad</i>	37
2.3.6 <i>Nodo Magnético</i>	40
2.4 LIMITACIONES Y OTRAS ESTRATEGIAS	45
2.5 CONCLUSIONES	46
3 ESTUDIO EMPÍRICO	47
3.1.1 <i>Participantes</i>	47
3.1.2 <i>Equipamiento y Material</i>	48
3.1.3 <i>Método</i>	48
3.1.4 <i>Tarea</i>	50
3.1.5 <i>Procedimiento</i>	51
3.1.6 <i>Resultados</i>	51
3.1.6.1 <i>Generalidades</i>	51
3.1.6.2 <i>Detalles sobre el efecto de la orientación</i>	53
3.1.6.3 <i>Precisión en técnicas no-asistidas</i>	67
3.1.6.4 <i>Ganancia en técnicas asistidas</i>	71
3.1.6.5 <i>Cuestionarios</i>	74
3.2 <i>DISCUSIÓN</i>	76
4 CONCLUSIONES Y TRABAJOS FUTUROS	79
4.1 <i>CONCLUSIONES</i>	79
4.2 <i>TRABAJOS FUTUROS</i>	81

BIBLIOGRAFÍA 83

ÍNDICE DE FIGURAS

Figura 1. Mesas Interactivas.....	10
Figura 2. Grafos.....	11
Figura 3. Creación de diagramas (Eclipse, VISIO, MagicDraw).....	12
Figura 4. Ejemplo de interfaz WIMP.....	16
Figura 5. URP System (MIT MediaLab).....	17
Figura 6. Iluminación Difusa.....	18
Figura 7. Microsoft Surface Unit (extraída de Microsoft TechNet Library).....	19
Figura 8. Diagrama de clases de los contactos.....	20
Figura 9. Identificación de contactos.....	21
Figura 10. Diagrama de clases del modelo de grafo.....	22
Figura 11. Dispositivos de entrada	23
Figura 12. Trayectoria de la arista.....	24
Figura 13. Arista creada.....	24
Figura 14. Creación de arista con técnica Arrastre (Diagrama de secuencia).....	26
Figura 15. Creación de arista con técnica de Pulsación (Diagrama de secuencia).....	29
Figura 16. Visualización en la técnica de Arrastre con visor.....	30
Figura 17. Diagrama de clases Visor.....	31
Figura 18. Plantilla y modelo por Vogel et al.....	32
Figura 19. Creación de arista con técnica Arrastre con Visor (Diagrama de secuencia).....	33
Figura 20. Creación de arista.....	35
Figura 21. Arista creada.....	35
Figura 22. Vectores Directores ($U, V1, V2 \dots Vn$).....	36
Figura 23. Técnica de Dirección.....	36
Figura 24. Creación de arista con técnica de Dirección (Diagrama de secuencia).....	37
Figura 25. Técnica de Proximidad.....	38
Figura 26. Conexión de nodos.....	38
Figura 27. Arista creada.....	38
Figura 28. Creación de arista con técnica de Proximidad (Diagrama de secuencia).....	40
Figura 29. Activación de nodo.....	41
Figura 30. Movimiento de nodo.....	41
Figura 31. Proceso de conexión.....	42
Figura 32. Arista creada.....	42
Figura 33. Creación de arista con técnica de Nodo magnético (Diagrama de secuencia).....	43
Figura 34. Uso de tangibles.....	48
Figura 35. Instrucciones y área de entrenamiento.....	49
Figura 36. Archivo de configuración para cada usuario.....	50
Figura 37. Tiempo promedio de conexión entre nodos por técnica y modo de interacción.....	52
Figura 38. Valor promedio de los fallos de conexión entre nodos.....	53
Figura 39. Diferentes posiciones donde se ubica el nodo destino.....	53
Figura 40. Tiempo promedio por técnica y modo de conexión entre nodos.....	55
Figura 41. Tiempo medio por técnica y modo de conexión entre nodos.....	57
Figura 42. Tiempo promedio por distancias de conexión entre nodos.....	58
Figura 43. Tiempo promedio por distancias de conexión entre nodos.....	59
Figura 44. Valor promedio de conexiones fallidas por técnica y modo.....	61

<i>Figura 45. Valor medio de conexiones fallidas por técnica y modo.</i>	<i>63</i>
<i>Figura 46. Cantidad media de conexiones fallidas por distancias.</i>	<i>66</i>
<i>Figura 47. Cantidad promedio de intentos fallidos por distancias de conexión entre nodos</i>	<i>67</i>
<i>Figura 48. Error por modo de las técnicas asistidas.</i>	<i>69</i>
<i>Figura 49. Error por distancia de las técnicas asistidas</i>	<i>71</i>
<i>Figura 50. Definición intuitiva de ganancia para las técnicas asistidas.</i>	<i>71</i>
<i>Figura 51. Ganancia de las técnicas asistidas por modo.....</i>	<i>72</i>
<i>Figura 52. Ganancia de las técnicas asistidas por obstáculo.</i>	<i>73</i>
<i>Figura 53. Centroides donde se levantan las líneas para conectar en distancias largas.</i>	<i>74</i>
<i>Figura 54. Centroides donde se levantan las líneas para conectar en distancias cortas.....</i>	<i>74</i>
<i>Figura 55. Cuestionarios basados en el NASA Task Load Index instrument</i>	<i>76</i>

INTRODUCCIÓN

1.1 Motivación del trabajo

La interacción hombre-computadora ha ido evolucionando [Hin2002]. Las interfaces de usuario de productividad mayormente extendidas en la actualidad son aquellas centradas en el paradigma de interfaces WIMP (Windows-Icons-Menues-Pointer). En éste las ventanas, iconos y menús son elementos interactivos, que pueden ser manipulados por medio de dispositivos de entrada de escritura o de señalización, como son el teclado y el ratón respectivamente. Estos elementos juntos constituyen un paradigma muy potente para la construcción de interfaces gráficas de usuario para usuarios individuales en un ordenador personal.

Sin embargo, nuevos dispositivos de entrada y visualización están permitiendo la creación de nuevas interfaces basadas en paradigmas más colaborativos y de múltiples usuarios. Especialmente interesantes son las mesas interactivas (ver Figura 1), que son un tipo particular de Interfaz de Usuario Tangible (ITU). Las mesas integran tanto la visualización como la entrada sobre una misma superficie. De esta forma, los usuarios pueden manipular objetos virtuales que se visualizan directamente con sus dedos, y el sistema puede entonces suministrar la respuesta visual a dichas interacciones.



Figura 1. Mesas Interactivas.

Además, las mesas suelen admitir como métodos de entrada a objetos tangibles, de tal forma que los usuarios podrían utilizar elementos físicos reales con los que están familiarizados. Uno de los trabajos iniciales que ilustran claramente las capacidades potenciales de estas mesas interactivas tangibles es el sistema URP [Ull2001]. *Urban Planning Workbench* (URP) es un simulador, pionero en el uso de interfaces tangibles, donde en tiempo real los usuarios pueden analizar modelos de construcción. Adicionalmente proporciona herramientas interactivas para consultar los parámetros de la simulación. Las mesas interactivas son un enfoque muy prometedor en el diseño, debido a que implican tareas que se pueden realizar de forma colaborativa. En los últimos años han sido estudiadas por investigadores como apoyo en una gran variedad de aplicaciones que se beneficiarían de la dimensión espacial que las mesas facilitan al considerar el uso de una superficie. Es un área de trabajo aumentada, donde objetos tangibles pueden ser manipulados y la información resultante de sus movimientos puede ser almacenada.

Muchas situaciones del mundo real pueden ser descritas mediante un diagrama que esté formado por un conjunto de puntos y líneas que unen ciertos pares de estos puntos, es decir vivimos en una sociedad donde se entrelazan entre sí objetos por medio de redes. Ejemplos de esto puede ser: un organigrama que expresa la estructura jerárquica en una empresa, para obtener el camino más corto en una red de carreteras, para explicar árboles genealógicos, en circuitería electrónica, para organizar la distribución de mercancías para y llevarlas desde el almacén a otros puntos de venta, para organizar la entrega de correspondencia para pasar por todos los caminos posibles sin repetir alguno. En fin, se usa en innumerables situaciones para resolver problemas que aparecen en diversas disciplinas tales como la logística, la robótica, la genética, la sociología, el diseño de redes, el cálculo de rutas óptimas, entre otras. Dicho diagrama es llamado grafo (Figura 2).

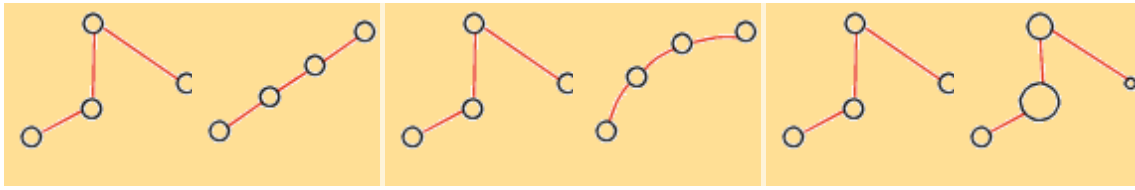


Figura 2. Grafos.

Un grafo es realmente un formalismo matemático que ofrece una representación natural de las redes en las que estamos cada vez mas incluidos. Su consolidación como conjunto de teoría matemática arranca con Leonhard Euler, cuando publicó la solución conocida a los puentes Königsberg. La teoría de grafos es una parte de la matemática discreta, por tanto tiene la estructura y la coherencia inherente a la propia matemática, esto hace que el razonamiento lógico esté siempre presente facilitando el aprendizaje.

Los grafos son diagramas matemáticos de intuitiva representación gráfica a la hora de resolver problemas de muy diferente índole, ya que se pueden expresar visualmente de manera sencilla y efectiva las relaciones que se dan entre elementos. Un grafo simple está formado por dos conjuntos:

- V : Son los puntos llamados vértices o nodos.
- Pares de vértices: llamados aristas o arcos y que indican qué nodos están relacionados.

Matemáticamente un grafo G es un par ordenado $(V(G), E(G))$ que está formado por un conjunto de vértices $V(G)$ y un conjunto $E(G)$ disjunto de $V(G)$ de aristas.

De la larga relación de temas y áreas que hemos enunciado anteriormente que utilizan la teoría de grafos como abstracción matemática esencial, podemos deducir que hay también infinidad de aplicaciones software que funcionan sobre interfaces WIMP para resolver problemas específicos. Muchos de ellos con toda seguridad requieren de manipulación espacial, y posiblemente sería adecuado utilizar una interfaz de gran formato, como son las mesas interactivas, para facilitar el trabajo en grupo. Ejemplos concretos de aplicaciones relacionadas con la Ingeniería Informática e Ingeniería del Software y Sistemas de Información serían las ilustradas en la Figura 3.

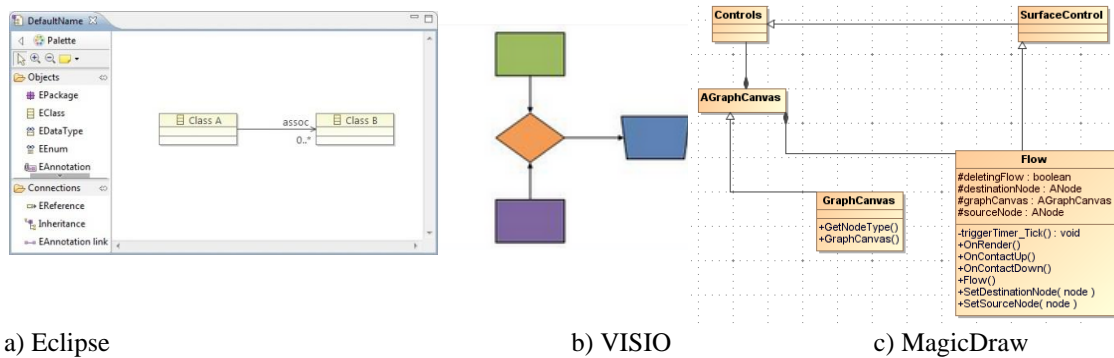


Figura 3. Creación de diagramas (Eclipse, VISIO, MagicDraw).

La utilización de grafos como formalismo subyacente para la resolución de problemas está muy extendida, y hace que se lleve a cabo mucha investigación en torno al uso de grafos. Sin embargo, desde el punto de vista de la interacción hombre-máquina resulta de interés estudiar factores humanos en su representación y manipulación, dado que muchas situaciones de la vida real, para poder representarse, requieren una gran cantidad de datos y que debido a su complejidad, dan como resultado diseños densos y difíciles de leer y sobretodo de representar visualmente. Es por ello que normalmente se estudia las capacidades de nuevas técnicas más efectivas para la visualización y redistribución automática de los nodos y aristas que permitan mayor comprensión por parte de los usuarios. Por ejemplo, algunos trabajos han explorado asuntos de distribución sobre la complejidad en la interpretación de grafos densos [Her2000]; otros sobre la importancia del cruce de aristas y asuntos estéticos [Pur2004], la generación automática de distribuciones visuales mediante aproximaciones algorítmicas [Bat1998],[Pau2006], así como la provisión de técnicas interactivas para la visualización [Won2003][Won2007] y navegación [Mos2009]. Sin embargo, poco hay estudiado en relación al uso de grafos en mesas interactivas pese a ser estas interfaces potencialmente útiles. .

Aunque se ha investigado asuntos sobre cómo incorporar técnicas para la creación de grafos en mesas interactivas [Fri2010], y se ha explorado la manipulación del grafo con propósitos de mejora de la visualización sobre superficies [Sch2010] no se ha explorado qué técnicas pueden ser más efectivas para la creación de aristas sobre mesas, ni se ha discutido elementos de diseño a considerar en la creación de las mismas. Es por ello que se plantea este trabajo final de máster, para empezar a ocupar ese espacio todavía no cubierto.

1.2 Objetivos del trabajo

El objetivo general de este trabajo es:

- Explorar y diseñar técnicas de interacción que permitan la creación de aristas en un grafo sobre mesas interactivas.
- Crear una aplicación en una tecnología multitáctil que permita evaluar un subconjunto de las técnicas de interacción consideradas atendiendo la eficiencia y la eficacia que los usuarios muestran en su uso.
- Discutir consideraciones de diseño importantes para técnicas de interacción para la creación de aristas sobre mesas interactivas.

El objetivo último de los resultados de este trabajo final de máster sería poder incorporar mejoras en un editor de reglas ya disponibles en el grupo de investigación que utiliza flujos de datos como abstracción principal.

1.3 Estructura del documento

Este trabajo se ha organizado en cuatro capítulos. A continuación se describe brevemente el contenido de cada uno de los capítulos restantes:

- Capítulo 2. Técnicas Para la Conexión.

El capítulo 2 introduce el diseño y el funcionamiento de técnicas para la creación de aristas en grafos sobre mesas interactivas. Algunas de ellas son técnicas que son ya comúnmente utilizadas mientras que otras son aproximaciones nuevas que quieren explorarse en este trabajo. El funcionamiento de cada una de ellas se describirá de manera detallada en este capítulo.

- Capítulo 3. Estudio Empírico.

El capítulo 3 presenta el estudio empírico llevado a cabo en el contexto de este trabajo. El objetivo es estudiar la conexión de nodos por parte de cada una de las técnicas implementadas en un entorno controlado. Este capítulo presentará los resultados del estudio y discutirá los aspectos sobre interacción más importantes. Capítulo 4.

- Capítulo 3. Estudio Empírico.

Este último capítulo presenta las conclusiones de este trabajo y los trabajos futuros que pretenden abordarse. Se discutirá las consideraciones de diseño que deben tenerse en cuenta en aplicaciones que requieran la conexión de elementos sobre mesas interactivas.

TÉCNICAS PARA LA CONEXIÓN

Teniendo en cuenta que vamos a estar centrados en la problemática particular de establecer conexiones entre nodos en un grafo sobre una mesa, en este capítulo vamos a introducir las técnicas que hemos diseñado e implementado para su posterior estudio. Previamente se introducirá lo que son las mesas interactivas, que es la tecnología objetivo de nuestro sistema e investigación, y se describirá su funcionamiento para poder entender las descripciones de las técnicas implementadas.

2.1 Contexto Tecnológico

2.1.1 Mesas Interactivas

Las Interfaces Gráficas de Usuario (IGUs) nacen de la necesidad de comunicación del usuario con la máquina. Son las partes del sistema que van desde la pantalla del ordenador a la utilización del ratón o teclado como dispositivo de entrada, y comprende también la forma en que estos dispositivos se utilizan de la manera más efectiva. Las IGUs son por tanto un asunto importante en cualquier sistema porque de su diseño y funcionamiento dependerá la eficiencia y la eficacia con la que los usuarios llevan a cabo sus tareas.

En la Figura 4 podemos ver una interfaz WIMP (Windows-Icons-Menues-Pointer), que es una abreviación de los conceptos de ventanas, iconos, menús y dispositivos de interfaz humano. Las ventanas, los iconos y los menús son elementos interactivos, que pueden ser manipulados por medio de dispositivos de entrada de escritura o de señalización, como son el teclado y el ratón respectivamente. Estos elementos juntos constituyen el paradigma más potente y eficiente alcanzado hasta el momento para interactuar con los ordenadores personales.

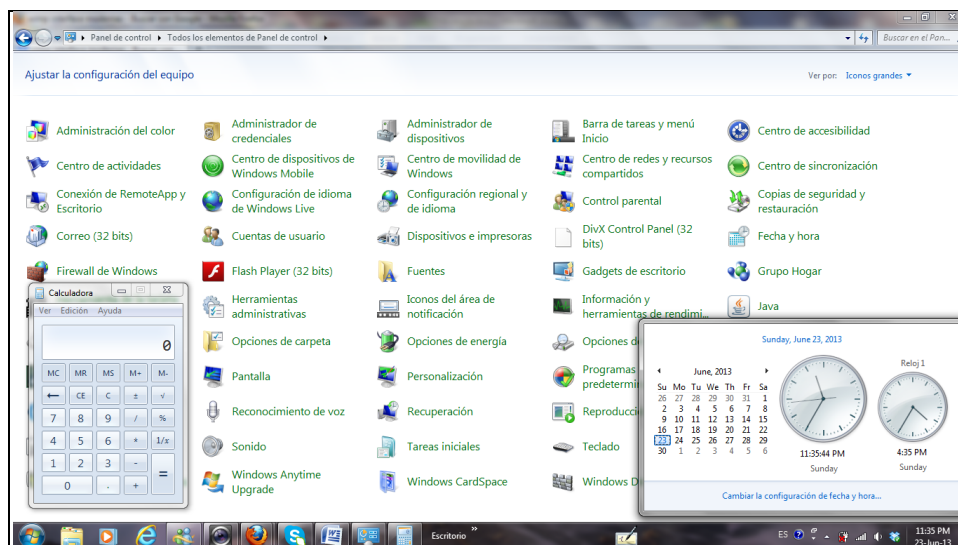


Figura 4. Ejemplo de interfaz WIMP.

Las Interfaces de Usuario Tangibles (ITUs) son una forma de interacción humano-computador que persiguen la combinación de un software de control la representación en un mismo dispositivo físico, de tal forma que unidos, generan una respuesta única para el usuario [Ull2001]. Mientras que las IGUs permiten la manipulación directa al seleccionar representaciones gráficas (iconos, ventanas, etc.) por medio de dispositivos apuntadores (p. ej. ratón), en las ITUs la integración que se persigue es tal que se pretende que el objeto físico o tangible reciba un estímulo de entrada, sea capaz de procesar la información y proporcionar una respuesta generando un cambio físico a la salida (*feedback* tangible).

En el caso de las mesas interactivas, que son consideradas un tipo particular de ITU, los usuarios interactúan con el sistema mediante sus manos y sus dedos, así como por manipulación de objetos físicos configurados (tangibles), y reciben *feedback* visual sobre la propia superficie que están tocando. Estos objetos pueden representar conceptos abstractos o entidades reales que pueden estar relacionados con otros objetos en la superficie de la mesa y que pueden moverse y girar en la superficie de la mesa. Todos estos cambios en sus propiedades espaciales pueden afectar sus propiedades internas y las relaciones con los objetos cercanos.

Para dar una idea del alcance e impacto que este tipo de interfaces pueden tener, un buen ejemplo es el trabajo seminal de Ben-Joseph *et al.*, URP (ver Figura 5). Es un sistema desarrollado en los 90 por MIT Media Lab [Ben2001], para planificación urbana. Fue uno de los primeros prototipos de aplicación sobre interfaces tangibles, se ejecuta en una mesa aumentada con un proyector y una cámara dirigida hacia la superficie desde arriba. Este sistema permite la detección de cambios en la posición de los objetos físicos

sobre la mesa y también proyecta la información visual relativa a la superficie. Otros elementos pueden ser incluidos, tal como un reloj para controlar el tiempo del día en el sistema. URP detecta los cambios realizados en tiempo real y proyecta las sombras según la hora del día. El interés de este prototipo se encuentra en que la información puede ser manipulada colectivamente, directamente y de manera intuitiva.



Figura 5. URP System (MIT MediaLab)

Si comparamos las interfaces tangibles con las WIMP, las primeras proveen una forma de interactuar más cercana al mundo real. Las interfaces tangibles muestran el potencial de las mesas en tareas espaciales. Precisamente, en el contexto de este trabajo final de máster estamos abordando en cierta medida una tarea en la que las relaciones espaciales son relevantes: el proceso de establecer conexiones entre elementos dispersos en una superficie, basándonos en un modelo de grafos. Es por ello, que explorar esta problemática sobre esta tecnología resulta de interés y novedosa.

La tecnología táctil con la que las mesas interactivas se construyen ha ido evolucionando a lo largo del tiempo [Sch2008]. En sus inicios requería una electrónica relativamente compleja ya que la detección se basa principalmente en el *hardware*. La tecnología táctil se ha popularizado con la entrada en el mercado de dispositivos de menor escala como PDAs, teléfonos móviles y tabletas, e incluso proporcionado algún acercamiento a las grandes pantallas con mesa, como el Touch Diamond [Die2001]. Sin embargo puede decirse que la aportación más significativa sobre la investigación intensiva sobre aplicaciones multitáctiles han sido los enfoques basados en cámaras para la detección de la interacción, que permiten la construcción de superficies de gran envergadura a un bajo costo de forma efectiva incluso en entornos no industriales.

Este enfoque requiere una cámara para capturar imágenes de la superficie, junto con algo de electrónica para disparar el efecto de detección del contacto en la superficie. Los dos principales tipos de superficies con cámaras ópticas se basan en FTIR (Frustrated Total Internal Reflection) y DI (Diffuse

illumination). Por ejemplo, una de estas aproximaciones, la DI utiliza fuentes de luz infrarroja, que se distribuye de manera uniforme sobre la superficie de contacto al disponer dichas fuentes enfocadas hacia la superficie de acrílico. Un difusor debe colocarse por encima o por debajo de la superficie táctil. El difusor se utiliza para difundir la luz, lo que debe ser un poco transparente. Parte de la luz infrarroja pasa a través de la superficie, pero cuando un objeto se coloca en la parte superior de la pantalla, la luz infrarroja es reflejada y la luz puede ser capturada por una cámara de infrarrojos (ver Figura 6).

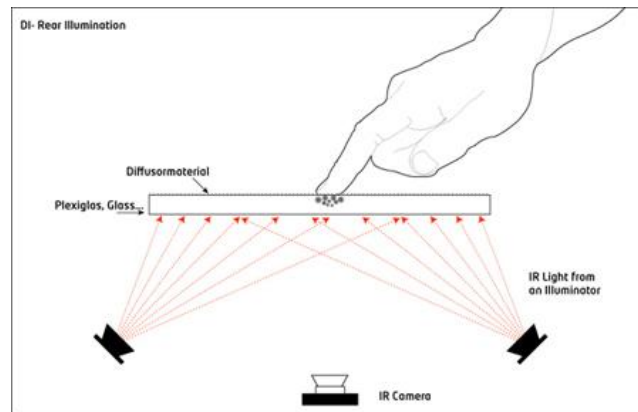


Figura 6. Iluminación Difusa

2.1.2 Microsoft Surface Unit

La Microsoft Surface Unit que actualmente se conoce como Microsoft Pixelsense [Microsoft-Surface], es una mesa con un ordenador, que a nivel de hardware incluye un sistema de visión y pantalla con 5 cámaras de iluminación directa de infrarrojo. Con éstas se captan los objetos, gestos con las manos y el tacto en la pantalla y se procesa esa entrada. La mesa responde a la entrada y proyecta el resultado en una pantalla de 30 pulgadas, con una tarjeta gráfica de 256 MB de memoria y una resolución de 1028 x 768 ver Figura 7.

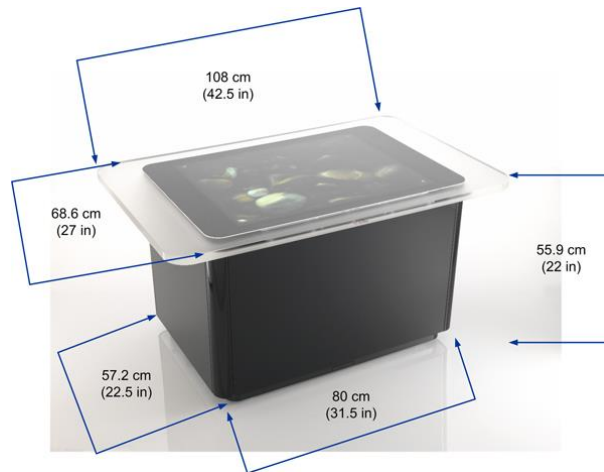


Figura 7. Microsoft Surface Unit (extraída de Microsoft TechNet Library)

La manera más rápida y eficaz para desarrollar aplicaciones para Microsoft *Surface* es la utilización de la capa de presentación basada en WPF suministrada por la Microsoft Surface SDK. Dicha capa extiende la *Windows Presentation Foundation* para ser utilizada en las interfaces de usuario táctiles.

Así como la entrada principal en el desarrollo de interfaces WIMP es el ratón y sus eventos relacionados, la mesa interactiva ofrece varios tipos de interacción como entrada principal, que son los llamados “contactos”. Cada contacto es detectado por el subsistema de visión por computador y es correlacionado a lo largo del tiempo con los demás por medio de un identificador único, proporcionando cierta información básica como la posición, orientación, etc. Para facilitar el uso de objeto tangible, el sistema reconoce etiquetas impresas que pueden ser adheridas a los objetos. Si el contacto es una etiqueta, no un dedo, tendrá información adicional disponible como los datos de la etiqueta. Las etiquetas pueden ser de dos tipos, uno que proporciona un código de 8 bits, y otro que proporciona un esquema de codificación más compleja basada en un par de serie-valor dentro de 128 bits. A continuación se muestra un diagrama de clases que ilustra los tipos de contactos y su información. (ver Figura 8)

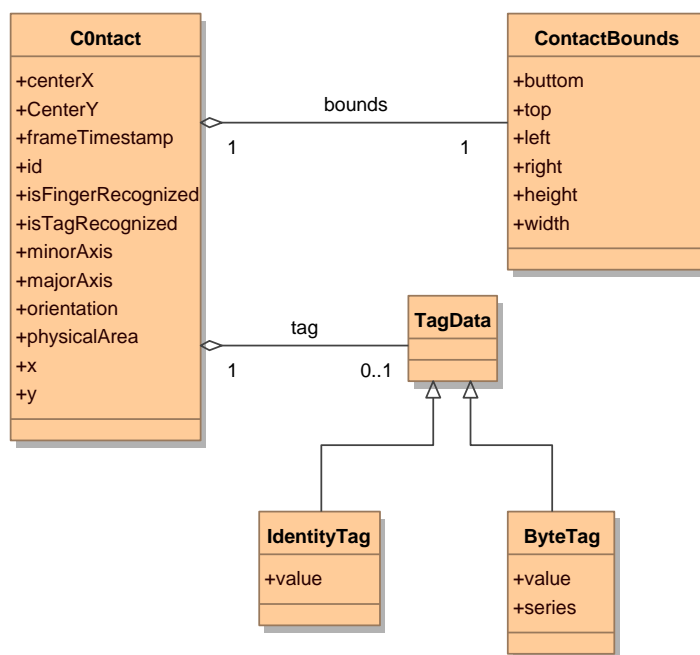


Figura 8. Diagrama de clases de los contactos

Los contactos tienen un ciclo de vida que cabe tener muy claro para entender cómo podemos desarrollar aplicaciones que sean capaces de responder apropiadamente a nuestros dedos. Al momento que un dispositivo de entrada (dedo/etiqueta) se pone sobre la superficie de la mesa, el subsistema de visión crea un identificador único para el contacto hasta que se levante el dedo, y se lanzará una serie de eventos en respuesta a las diferentes manipulaciones que el usuario haga con su dedo, que permitirá al programador conducir su aplicación por eventos.

Los eventos relevantes son *Enter*, *Down*, *Changed*, *Up*, *Leave*, de los cuales los esenciales son *Down*, *Changed* y *Up*, que funcionan de la siguiente manera. Cuando un dedo se pone sobre la mesa, el gestor de la interfaz notifica el evento *Down*, proporcionando la información del contacto (posición, orientación, etc.) como se muestra en la Figura 9. Si el usuario mueve ese dedo sin levantar, el gestor notificará sucesivamente eventos *Changed*, que indicarán la posición del dedo en cada momento. Cuando el usuario levanta el dedo se notifica el evento *Up*.

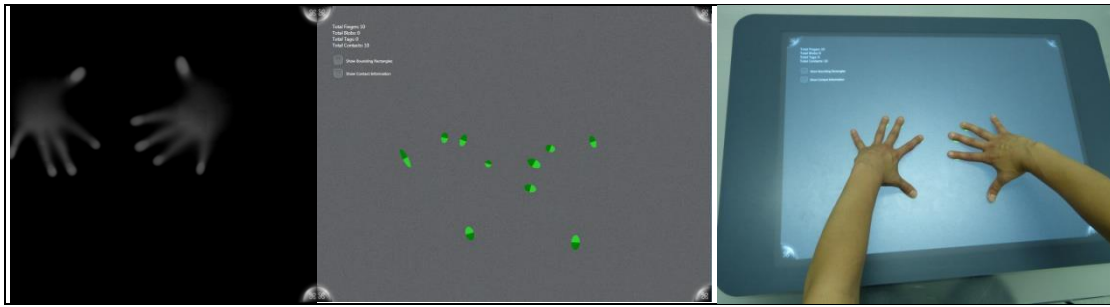


Figura 9. Identificación de contactos.

2.2 Marco de Trabajo para la Edición de Grafos

Dado que el objetivo principal de esta tesis de máster es explorar el diseño y evaluación en el manejo de técnicas para la creación de aristas en la construcción de grafos, se ha desarrollado un marco de trabajo genérico que nos permita representar grafos visualmente de forma muy básica sobre la mesa interactiva. En el modelo propuesto no se está considerando una aplicación específica de grafos, tales como pueden ser aquellas aplicaciones de modelados de diagramas de clase UML o de dibujado de grafos en general. Acto seguido se evaluará el grado de eficacia de los usuarios al manejar dichas técnicas. No obstante, se parte de la idea de explorar técnicas que puedan ser finalmente integradas en un editor de expresiones de *flujos de datos* como los utilizados en TanRule [Pon2013]. Esto nos abre muchas posibilidades para poder adaptar las técnicas diseñadas e implementadas en otras aplicaciones de diagramas sobre mesas interactivas, ya que el marco es lo suficientemente genérico.

El diagrama de clases que ilustra la Figura 10 muestra los principales conceptos que hemos considerado en nuestro modelo conceptual para la creación de grafos. La clase principal es el *AGraphCanvas*, que es la clase que permitirá definir, visualizar, e interaccionar con el grafo. Éste hereda de la clase *Canvas*, que es un control de interfaz de usuario que permite contener otros controles, y que permite fácilmente manejar la entrada de contactos sobre la mesa. Un grafo en el *AGraphCanvas* se compone de nodos y de aristas. En nuestro caso particular, los nodos vienen representados por la clase *ANode*, mientras que las aristas o arcos están representados por la clase *Edge*. Es por ello que el grafo (*AGraphCanvas*) tiene las relaciones *nodes* y *edges* que contiene los nodos y aristas respectivamente.

Otro elemento importante es la clase *ANode*. Los nodos forman las aristas (clase *Edge*). Es por ello que la relación *edges* permite obtener todas las aristas en las que participa mientras que las asociaciones *sourceNode* y *destinationNode* de la clase *Edge* establecen los roles de los nodos que forman parte de la arista.

Las clases *AGraphCanvas* y *ANode* son clases abstractas que concentran también la implementación genérica las funciones necesarias. Cada técnica requerirá extender e implementar estas clases siguiendo el patrón de diseño creacional *Abstract*, con el propósito de poder particularizar la forma en que las conexiones se realizan pero manteniendo genérica su gestión por parte de la aplicación.

El resto del diagrama de la Figura 10 representa las clases y las relaciones más importantes que serán necesarias para llevar a cabo *testing* y gestionar el registro de información experimental. Podemos encontrar la clase *EventLogger*, que gestiona la creación de ficheros de *logs* y que contiene los métodos específicos de escritura de entradas al log por evento. En el sistema hay una única instancia creada siguiendo el patrón de diseño *singleton*, que es mantenida por la clase *AGraphCanvas*. Así mismo, la clase *ExperimentManager* gestiona el progreso de las tareas en el entorno de experimentación y muestra las instrucciones de cada tarea por medio de los *InstructionCanvas*.

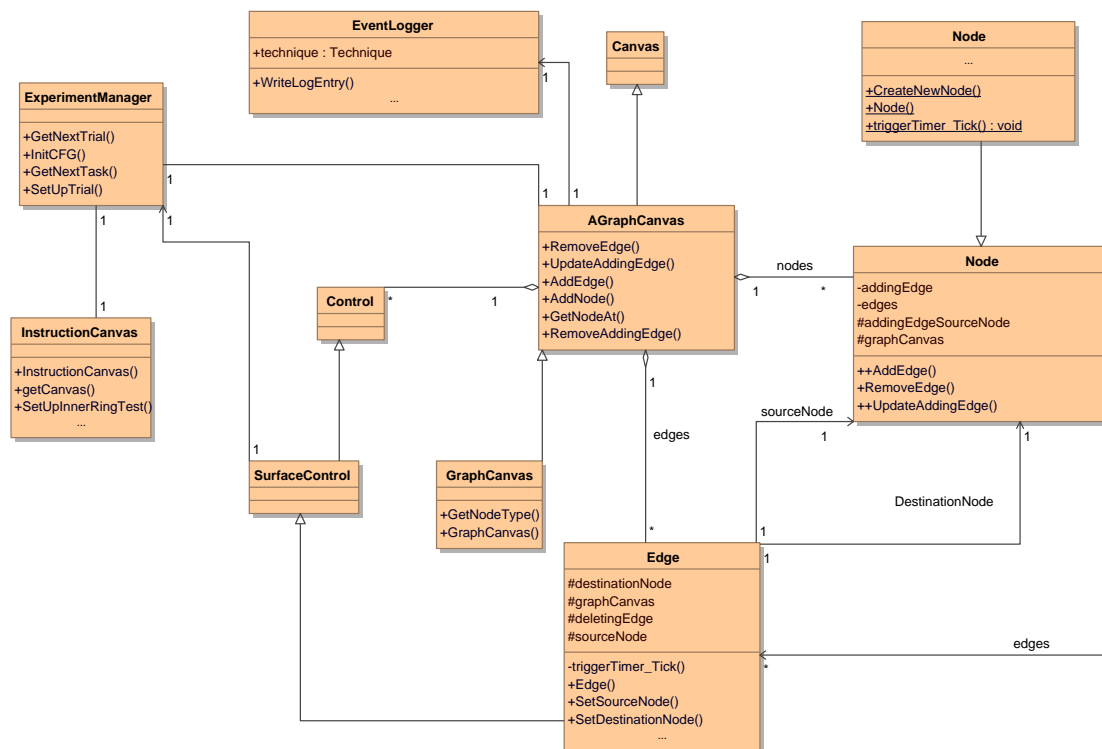


Figura 10. Diagrama de clases del modelo de grafo.

2.3 Técnicas de Interacción

A continuación se describen las técnicas de interacción diseñadas e implementadas para su uso en la mesa interactiva en el contexto de este trabajo.

2.3.1 Arrastre

Una de las aproximaciones más utilizadas, sería aquella que consiste en la unión de nodos describiendo íntegramente la trayectoria que la arista tiene que tomar por medio del dispositivo de entrada, ya sea ratón, *stylus* o dedos. Esta técnica la hemos denominado “Arrastre”, porque en esencia, se requiere arrastrar el dispositivo de entrada a lo largo de la trayectoria determinada desde el nodo origen hasta el destino, teniendo como resultado la unión de ambos nodos.

Por ejemplo, diversos paquetes de software comerciales basados en WIMP utilizan dicha aproximación en la creación de diagramas genéricos como Microsoft Visio (Figura 3b) o herramientas CASE de soporte de UML tales como MagicDraw (Figura 3c) o editores de diagramas de clase en Eclipse_(- Véase la Figura 3a).

Esta técnica está muy extendida y es comúnmente utilizada. También se ha podido comprobar que forma parte de las técnicas que ya se pueden encontrar adaptadas a superficies multi-táctil. En particular, en el trabajo de Frisch et al [Fri2010], se integra en la edición de diagramas en superficies que disponen de *stylus* como método de entrada.

En la Figura 11 se muestran los dispositivos de entrada implementados en esta técnica.

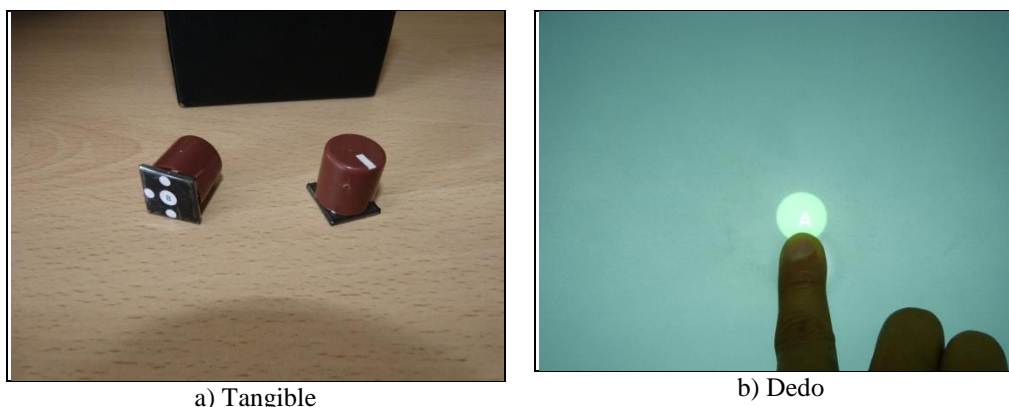


Figura 11. Dispositivos de entrada

Los nodos pueden moverse de forma directa, de tal forma que si tocamos un nodo e inmediatamente los arrastramos sobre la superficie, la posición del

nodo irá cambiando siguiendo la estela de nuestro dedo o tangible, tal y como cabría esperar que ocurriera en una aplicación real que requiriera la edición de grafos. Al soportar esta funcionalidad de movimiento de nodos a voluntad, y como la principal interacción es el “arrastrar”, es necesario introducir un mecanismo de activación que permita discriminar las situaciones de “mover” de las de “conectar”.

Llamamos *activación* a la interacción de adquisición que permite indicar a un nodo concreto que debe trazar una arista en respuesta al movimiento de nuestro dedo o tangible. Aunque habría muchas posibilidades diferentes para llevar a cabo dicha activación (p. ej. teniendo un botón que haga transitar o conmutar el modo de movimiento/conexión), hemos optado finalmente por tratar de integrar una técnica que sea local a cada nodo, que permita activación independiente para que soporte la realización de conexiones simultáneas y paralelas en el grafo por parte de varios usuarios, y que sea coherente con todas las técnicas descritas en este trabajo. En particular, la activación del nodo para que pueda trazar una arista consiste en la pulsación continuada prolongada durante un lapso de tiempo determinado configurable en la aplicación, por defecto establecido a 250ms.

Indistintamente del dispositivo de entrada que se elija para crear la arista, se pulsa sobre el nodo origen hasta que se active, esto es fácil de observar pues cambia ligeramente el tamaño y el color, proporcionando *feedback* visual sutilmente. Una vez activado, cuando se mantiene pulsado el nodo y se inicia la trayectoria de arrastre sobre la superficie de la mesa hasta el nodo destino, se observa el dibujado progresivo de la línea mientras se arrastra en la Figura 12. Trayectoria de la arista. Figura 12. En este momento, una vez sobre el nodo destino, puede levantarse el dispositivo de entrada y los nodos quedarán unidos (Figura 13). Si la trayectoria de arrastre no llega hasta el nodo destino y se levanta el dispositivo de entrada, la creación de la arista se cancela.

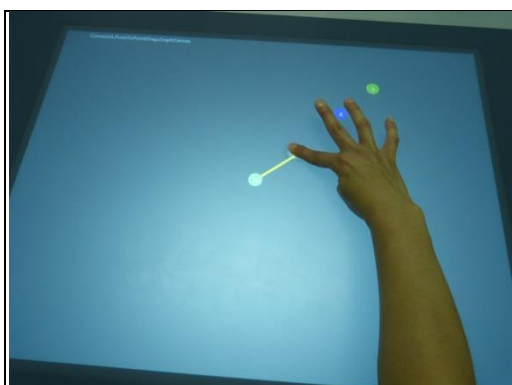


Figura 12. Trayectoria de la arista.

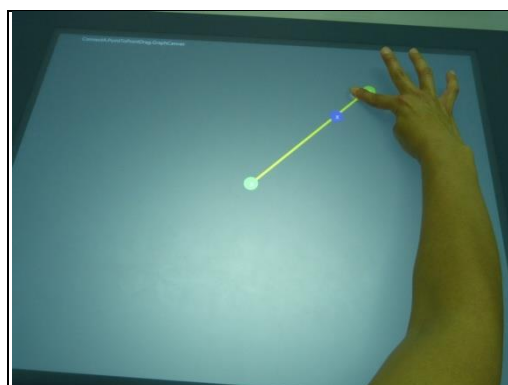


Figura 13. Arista creada.

A continuación se describe la interacción principal para llevar a cabo la conexión de dos nodos en esta técnica mediante un diagrama de secuencia

(ver Figura 14). Para facilitar la legibilidad del diagrama y no complicarlo innecesariamente, no se incluyen en el propio diagrama alternativas en el progreso de la interacción (e.g. cancelación o fallo en la conexión).

Debemos distinguir en el diagrama tres acciones relevantes que el usuario debe realizar para llevar a cabo la conexión: poner el dedo sobre el nodo origen y activarlo; sin alzar su dedo, moverlo de manera continuada trazando la arista; levantar el dedo una vez alzando el nodo destino. Estas acciones, tal y como hemos visto en la sección 2.1, producen eventos *OnContactDown*, *OnContactChanged*, y *OnContactUp* convenientemente sobre los controles que son gestionados e informados por la IU.

Cuando el usuario pone el dedo sobre el nodo origen se lanza un temporizador, que activará el nodo tal y como se ha descrito anteriormente si pasa bastante tiempo (unos 250 ms) sin mover el dedo que se ha puesto sobre el nodo. Será entonces cuando se provea *feedback* visual para indicar la activación del nodo, representado en el diagrama con el mensaje *activatedNode*.

Hemos considerado que una arista (*edge*) es una conexión consolidada entre dos nodos, que se implementa con el control *Edge*. Sin embargo, en el proceso de creación de la conexión, es importante proporcionar *feedback* visual sobre la creación, de tal forma que se vaya mostrando la conexión que se está trazando a medida que se desplaza el dedo o tangible. Para poder realizarla, el canvas (*AGraphCanvas*), que es el contenedor de todos los controles, va a encargarse de visualizar aquellos trazos que se correspondan a aristas en progreso. Para ello, el canvas mantiene internamente una tabla con la información necesaria para poder representar estos trazos, que hemos llamado *addingEdges*. En esencia, es necesario mantener para cada trazo en proceso de creación (i.e. para cada *addingEdge*), el nodo origen que está activado y el punto de contacto del dedo/tangible que en cada momento es el extremo final del trazo.

Será tras la activación del nodo cuando se informe al *canvas* del inicio del proceso de añadir una *edge*. De esta forma, el *canvas* inicializa una entrada en su tabla interna de *addingEdges* para que pueda dibujar el trazo parcialmente realizado.

Si el usuario mantiene el dedo sobre la superficie e inicia la trayectoria de arrastre del nodo origen hacia un posible nodo destino, se producirán eventos *OnContactChanged* de manera sucesiva. Será entonces cuando se actualiza la información en la tabla interna *addingEdges* (por medio del método *UpdateAddingEdge*) para que el canvas sea capaz de mostrar el trazo en progreso totalmente consistente con la nueva posición. Esto producirá que usuario vea que hay un trazo que sale del nodo activado y que va siguiendo su dedo/tangible.

Al momento que el usuario levanta el dedo, se verifica si el último punto de contacto actual sobre la mesa antes de levantarlo ha sido sobre un nodo. Si es este el caso se crea una instancia de *Edge* y se establece la arista entre ambos nodos, creando la arista consolidada que une el nodo origen y el destino. Si se levanta el dedo sobre cualquier otra posición en el que no haya nodos, se cancela la creación de la arista. En cualquiera de los casos, se vuelve a los valores por defecto que permitan iniciar la creación de nuevos trazos partiendo de ese nodo.

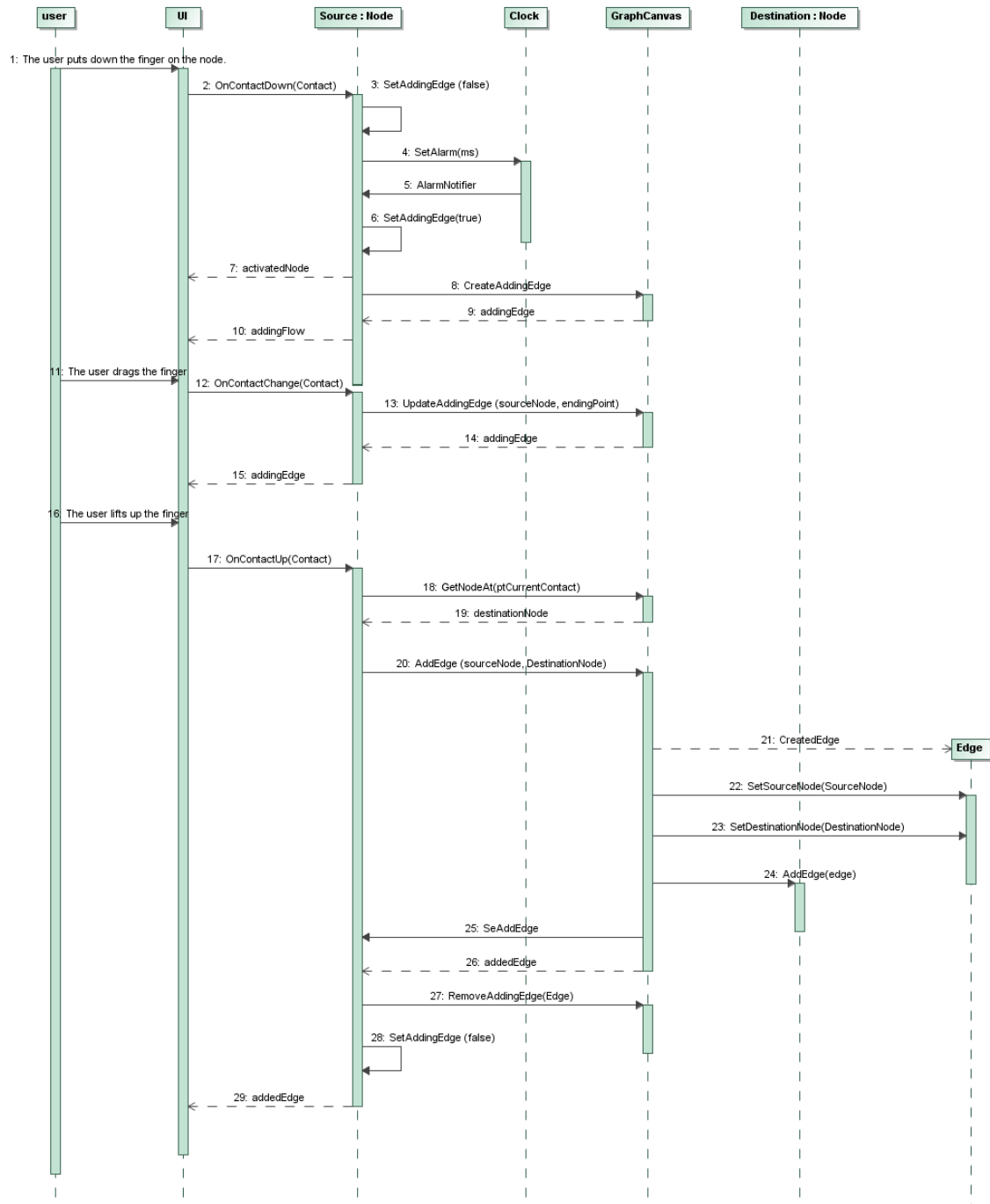


Figura 14. Creación de arista con técnica Arrastre (Diagrama de secuencia).

Pese a ser el “arrastré” una técnica muy extendida en interfaces WIMP, el hecho de adaptarla y utilizarla en una mesa interactiva puede tener ciertas implicaciones debido a que su superficie suele ser de tamaño considerable y requiere el movimiento del brazo a lo largo de la superficie. Por tanto, a priori puede considerarse a la fatiga y la precisión como una desventaja potencial, si la distancia entre los nodos es considerable o se tienen que construir muchas aristas en los grafos, porque será necesario arrastrar toda la distancia.

Otra cuestión muy interesante a mencionar es que, a diferencia de las interfaces WIMP mediadas con ratón (i.e. método de entrada indirecto), el uso del dedo produce oclusión, de tal forma que el brazo, la mano y el propio dedo pueden dificultar la visión del área de interés. La oclusión parece un factor importante al momento de conectar con el nodo destino ya que dificulta conocer con exactitud en el momento en el que se realiza esa conexión y es un problema que puede afectar seriamente el rendimiento y queda de manifiesto en trabajos de investigación, tales como en los trabajos [Han2004], [Ink2006]. En diversas técnicas de interacción ha quedado de manifiesto este problema [Ze2006]. Algunas técnicas intentan suavizar este problema usando un control deslizante como en [Ram2003], [Api2005] e incluso se consideran ubicar los controles en un área que no lo esté ocluida, [Vog2012].

2.3.2 Pulsación

La técnica de “Pulsación” es probablemente una de las primeras formas en que el usuario esperaría e intentaría hacer la conexión de dos nodos. En primer lugar se determina un nodo como origen, presionando sobre él, a continuación se precisa el nodo destino y se pulsa sobre él. Es en este momento cuando los nodos quedan unidos automáticamente, generando una arista entre ellos.

Es bastante trivial, de fácil comprensión y manejo para el usuario, dado que es muy intuitiva. Solo con haber pulsado de forma secuencial los nodos implicados se crea la conexión, y además ha sido también utilizada en trabajos de investigación relacionados con edición de diagramas en superficies [Fri2010].

En el diagrama de secuencia (ver Figura 15), se destacan las acciones importantes que el usuario ejecuta para que se realice la conexión entre nodos: poner el dedo sobre el nodo que sea considerado como origen, levantar el dedo tras haber sido activado el nodo y tocar el nodo que se quiera como destino. Esta secuencia de actividades genera los eventos *OnContactDown* y

OnContactUp sobre los controles que representan los nodos en la interface de usuario.

De igual forma que ocurría en la técnica de arrastre, es necesario activar el nodo para empezar a realizar la conexión. Así, cuando el usuario pone el dedo sobre el nodo origen y lo mantiene durante el lapso de tiempo necesario sin moverlo el nodo se activa. Una vez activado el usuario puede levantar su dedo.

Cuando el usuario pulsa sobre el nodo destino se producirá nuevamente el evento *OnContactDown*. Es entonces cuando el *canvas* proporciona la referencia del nodo origen guardada con anterioridad. Se crea una instancia de *Edge* y se crea la arista, uniendo el nodo origen y el destino de forma automática.

Cabe destacar que tras la activación, se informa al *canvas* de qué nodo ha sido activado para controlar el proceso de creación de la unión de nodos, de tal forma que no habrá activado más de un nodo al mismo tiempo, y la secuencialidad de pulsaciones determinará el establecimiento de conexiones. Esto tiene varias implicaciones en las posibilidades de interacción que una aplicación que utilice esta técnica para la conexión de nodos. Por ejemplo, parece evidente que no se soportaría el establecimiento en paralelo de múltiple conexiones, dado que solo puede haber un nodo activado al mismo tiempo. No obstante, dicha limitación podría ser mitigada con la incorporación de técnicas que marquen territorios de manipulación [Gar2013] o que discriminen qué usuario está realizando la pulsación [Zha2012], de tal forma que se puedan disponer de un nodo activado por cada área o usuario respectivamente.

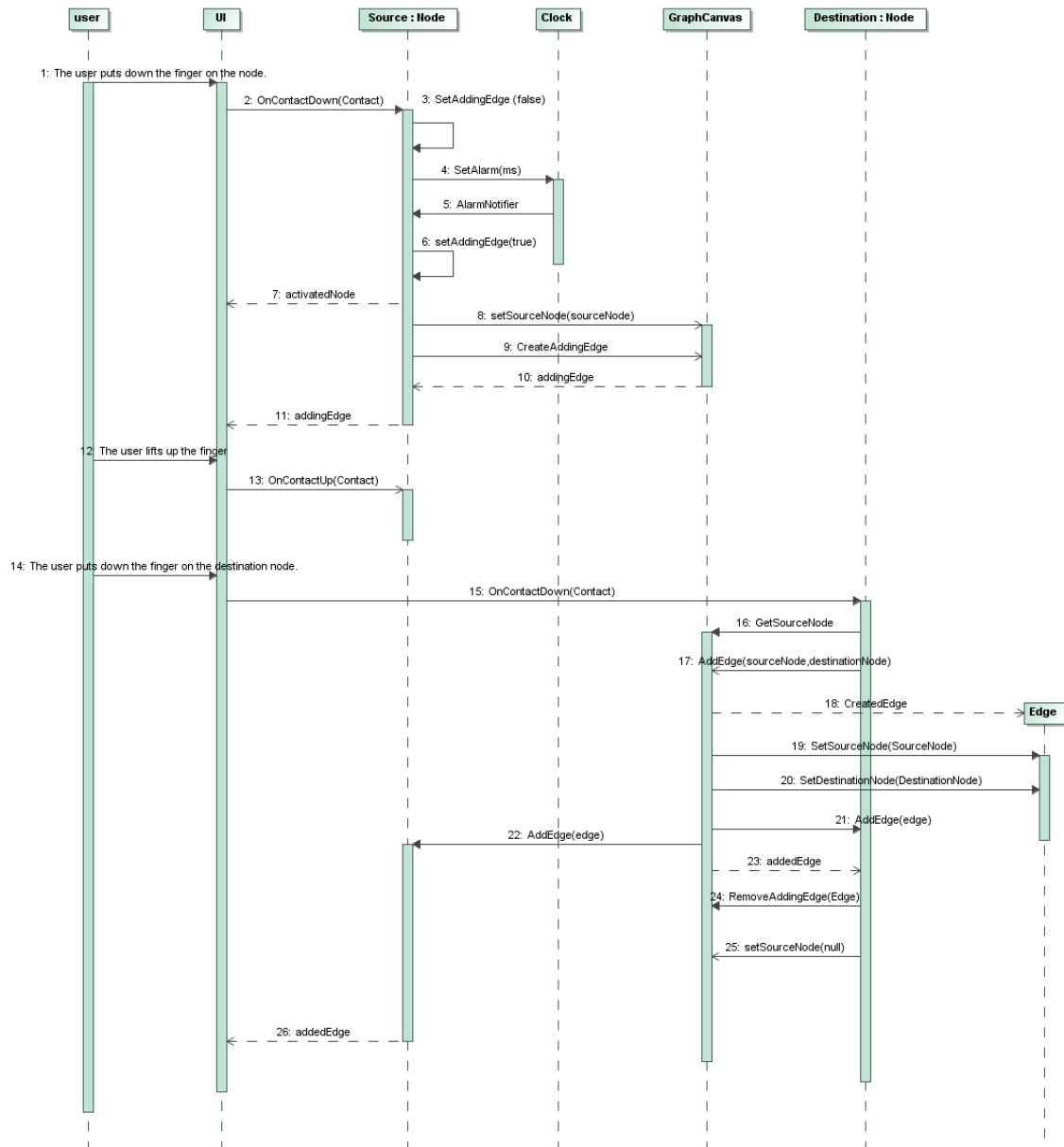


Figura 15. Creación de arista con técnica de Pulsación (Diagrama de secuencia).

Puesto que solo se debe seleccionar los nodos origen y destino, cuando el área donde está el nodo destino es muy poblada, esta técnica puede suponer una reducción de error al conectar con el nodo deseado directamente. Por otro lado puede suponer una reducción tanto en el esfuerzo del usuario como en el tiempo requerido para la conexión de nodos porque solo se actúa sobre cada uno de los nodos, y el brazo tan solo se tiene que mover de un punto a otro sin realizar ningún tipo de interacción con la superficie.

2.3.3 Arrastre con Visor

Podríamos decir que esta técnica es equivalente a la denominada “Arrastre” (sección 2.3.1), salvo que tiene el valor añadido de proveer una lente que

resulta interesante desde el punto de vista de visualización pues viene a solucionar la problemática que se da cuando sucede oclusión por parte del dedo/brazo del usuario. Esta técnica se basa en la adaptación de trabajos previos sobre interacción con dispositivos táctiles que tratan de mitigar el problema de la oclusión. En particular, la técnica se basa en *Shift* [Vog2007], que crea y desplaza una copia del área de la pantalla ocluida y la coloca en una posición no ocluida.

En este caso le hemos llamado “Visor”. Al activar el nodo origen aparece esta lente en un área cercana a la mano pero donde se espera que no se tenga problemas de oclusión. El visor está delimitado por un rectángulo que se mueve solidariamente al movimiento de arrastre del dedo y que muestra la región que se encuentra en cada momento debajo del dedo en cuestión. Si el usuario utiliza el visor, podría superar cierta incomodidad producida por la oclusión, y podría mejorar también la fiabilidad en la realización de las conexiones porque sería capaz de ver si realmente su dedo se encuentra sobre el nodo destino y por tanto no levantar el dedo pensando que estaba sobre el destino cuando realmente no lo está. En el momento en el que se produce la conexión, el visor desaparece.

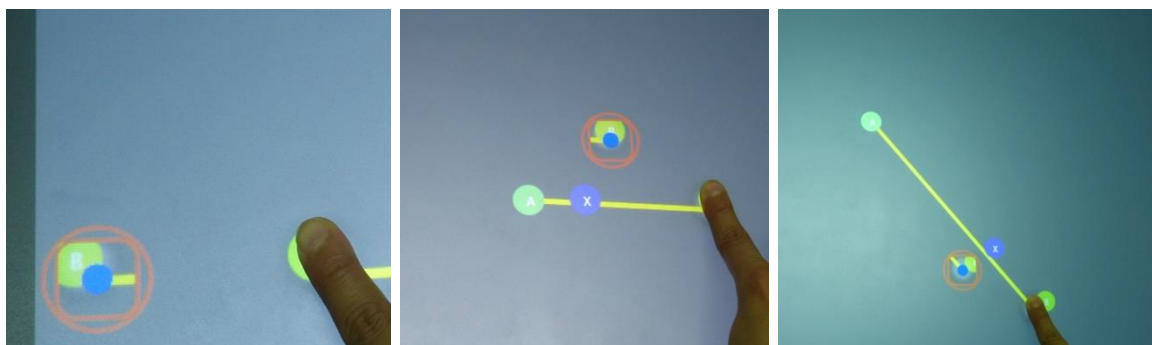


Figura 16. Visualización en la técnica de Arrastre con visor.

En el modelo de clases para la técnica de Arrastre con Visor hay una clase adicional llamada *RegionViewer* (ver Figura 17), que representa una región con el área ocluida por el dedo/tangible. Cada objeto nodo tiene una *RegionViewer*, y en esta clase se encuentra el método para desplazar la región, se le pasa como parámetro la nueva posición. Esta región permite que el visor se visualice por encima del resto de los controles, de esta manera el *canvas* contiene la instancia del *RegionViewer*.

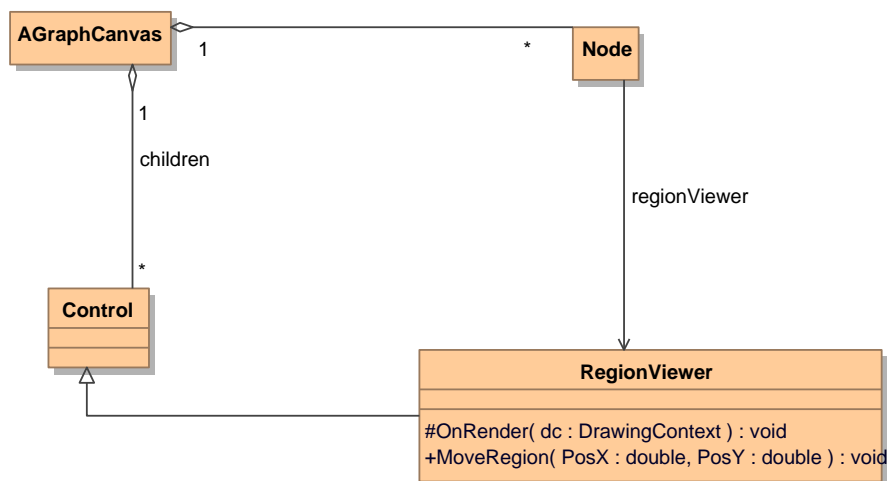


Figura 17. Diagrama de clases Visor.

Tal y como muestra el siguiente diagrama de secuencia (ver Figura 19), el usuario necesita realizar tres acciones importantes para llevar a cabo la conexión entre nodos: Poner el dedo sobre el nodo origen y activarlo en el evento *OnContactDown*; sin alzar su dedo, moverlo de manera continuada trazando la arista produciendo el evento *OnContactChanged* y levantar el dedo una vez alzando el nodo destino en el evento *OnContactUp*.

Cuando el usuario pone el dedo sobre el nodo origen se lanza el temporizador hasta activarse el nodo, representado en el diagrama con el mensaje *activatedNode*.

Cabe destacar que el evento *OnContactChanged* es igual que el de la técnica de “Arrastre”, solo que actualiza el contenido de una cola con los últimos n contactos. Esta cola (*lastContactQueue*) permite obtener una posición así como y una orientación suavizada (promediada) que permitirá ubicar cómodamente el visor sin que salte de forma abrupta entre cada mínimo cambio del dedo. A cada cambio del dedo, también se captura la región que se encuentra debajo del dedo y se actualiza inmediatamente la imagen que debe mostrar el visor. Dado que esto ocurre de manera sucesiva se va actualizando la información de la región capturada y su posición de forma continuada.

Al momento que el usuario levanta el dedo, después de haber creado la arista que une el nodo origen y el destino, las variables vuelven a su valor por defecto. Es preciso mencionar que esta técnica funciona con varios nodos a la vez, es decir, cada nodo al que se le está añadiendo una arista le informa al *canvas* que le proporcione la región para que el nodo la almacene y al final se muestran las regiones que poseen los nodos implicados.

La localización del visor se sitúa siempre sobre la parte superior izquierda en relación a la posición y orientación del punto de contacto. Este no es un

hecho al azar y se realiza de esta forma en virtud de los resultados del estudio conducido por Daniel Vogel [Vog2012]. En dicho estudio se trata el problema de la oclusión del antebrazo sobre mesas interactivas (Figura 18). Para ello examina la forma de la mano y la posición relativa del brazo, y han dado con un modelo geométrico de oclusión resumido una serie de plantillas. Estos resultados sugieren que esa posición sería la más idónea para colocar la información activa para manipulaciones realizadas por usuarios diestros.

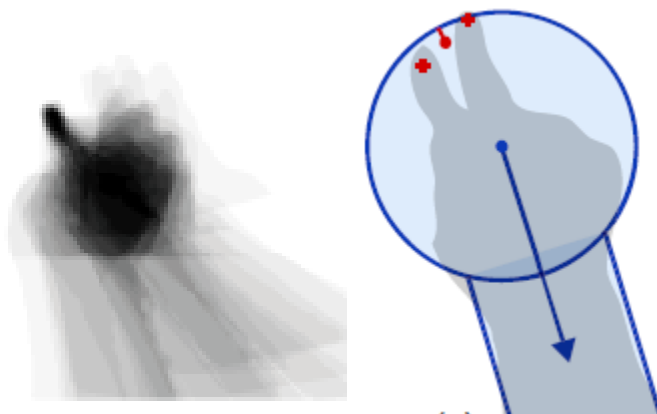


Figura 18. Plantilla y modelo por Vogel et al.

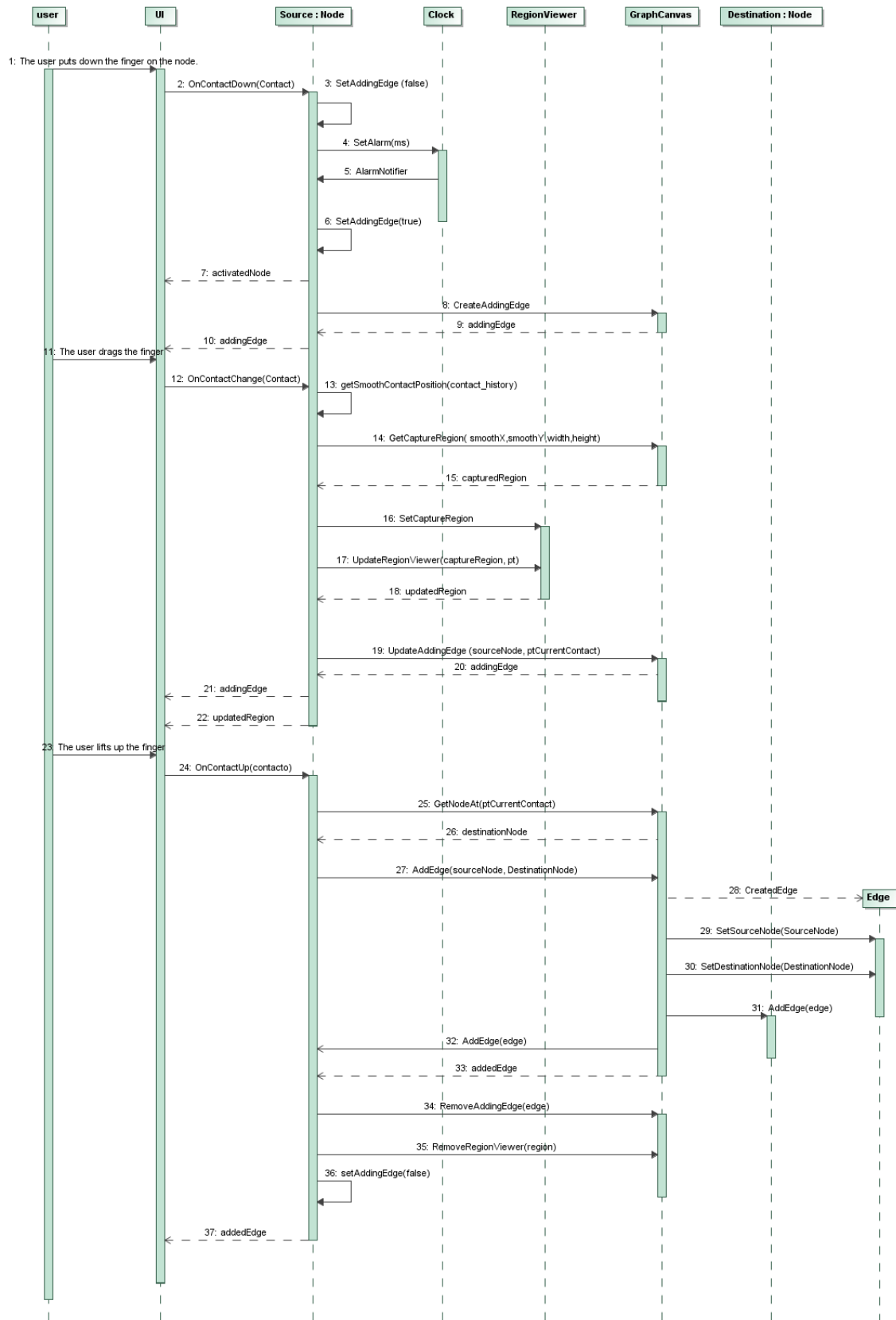


Figura 19. Creación de arista con técnica Arrastre con Visor (Diagrama de secuencia).

A priori se ha propuesto la técnica de Arrastre con Visor como una alternativa para mejorar el problema de oclusión, implementando este visor en un punto estratégico para los usuarios. En la parte de experimentación evaluaremos el rendimiento.

2.3.4 Dirección

Esta técnica la hemos llamado “*Dirección*”. Es una alternativa que permite hacer la conexión entre los nodos de una manera muy sencilla cuando estos se encuentran alineados y sin obstáculos entre ellos. La técnica funciona como se describe a continuación (véase la Figura 20).

Cuando el usuario activa el nodo origen (A) y arrastra en dirección a un posible nodo destino; si se ha recorrido una distancia mínima determinada (d), la técnica sugiere la conexión con el nodo que se encuentre en la dirección del rayo trazado en esa dirección. Con el propósito de no exigir precisión absoluta al usuario en la determinación de la dirección del trazo, la técnica utiliza un ángulo θ que determina con cierta holgura todos los nodos que están dentro de su abertura como posibles candidatos para la conexión, escogiendo finalmente el más cercano en la dirección del rayo trazado. Del punto de contacto se emite una línea de unión (vector director) al nodo más cercano teniendo en cuenta los criterios de holgura anteriormente descritos. Una vez el vector haya coincidido con el nodo destino deseado, al levantar el dedo o el tangible consolidará la conexión (Figura 21). Es importante mencionar que si al generarse el ángulo y dentro de la abertura del mismo no hay nodos, en este caso no se producirá línea de unión (vector director) alguna, será necesario girar el punto de contacto en busca de un nuevo ángulo que pueda encontrar nodos que sean captados como posibles conexiones.

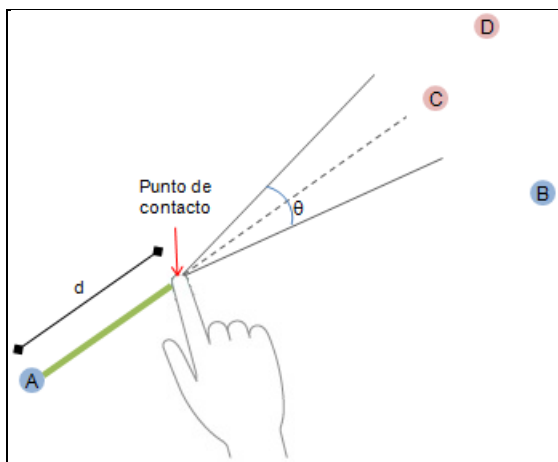


Figura 20. Creación de arista.

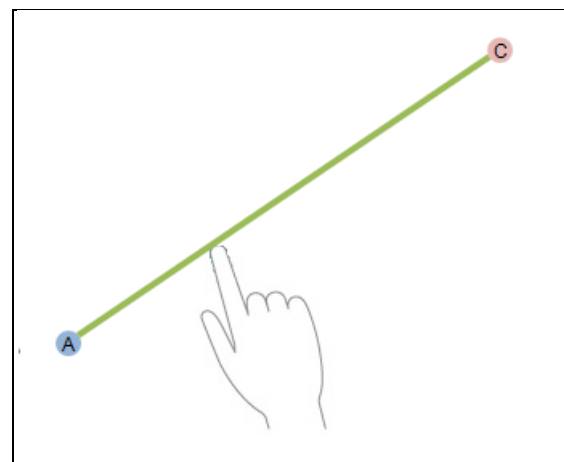


Figura 21. Arista creada.

A excepción de la técnica de pulsación, la implementación de esta técnica es similar al resto. El cambio significativo radica en la forma en la cual se determina el nodo destino candidato, en base al criterio del rayo que traza el vector director entre el nodo origen y el dedo. A continuación se muestra el diagrama de secuencia donde se visualizan las principales interacciones entre los objetos (ver Figura 24).

Al producirse el evento *OnContactChanged* cuando el usuario mantiene el dedo sobre la superficie e inicia la trayectoria de arrastre del nodo origen hacia un posible nodo destino, el canvas muestra el trazo en progreso con la nueva posición, cuando ese punto de contacto llega a una distancia determinada, además también en el método *GetNodeAtDirectionRay(sourceNode, ptCurrentContact, Angle)* se le envía, el nodo origen, el punto de contacto actual y el ángulo(θ).

Se obtiene los vectores directores de la recta que pasa por el nodo origen y el punto de contacto denominado (U). Para cada una de las rectas que van desde el punto de contacto hasta los posibles nodos destinos se determina también un vector director ($V_1, V_2 \dots V_n$) (Figura 22). Se calcula el ángulo (α) formado por las dos rectas conociendo sus vectores directores (U y V).

$$\cos \alpha = \frac{|U_x - V_x + U_y - V_y|}{\sqrt{U_x^2 + U_y^2} * \sqrt{V_x^2 + V_y^2}}$$

Se obtienen todas las distancias a los nodos destinos candidatos siempre y cuando el ángulo α sea menor que el ángulo de referencia θ . Es entonces cuando se obtiene la menor de ellas y es designada como el nodo destino y notificada por el *canvas*, y finalmente cuando se actualiza la información en la tabla interna *addingEdges (UpdateAddingEdge)* para que el *canvas* actualice consistentemente la nueva posición. Esto producirá que el usuario vea un trazo que sale del nodo activado de su dedo/tangible al nodo más próximo al trazo del vector director.

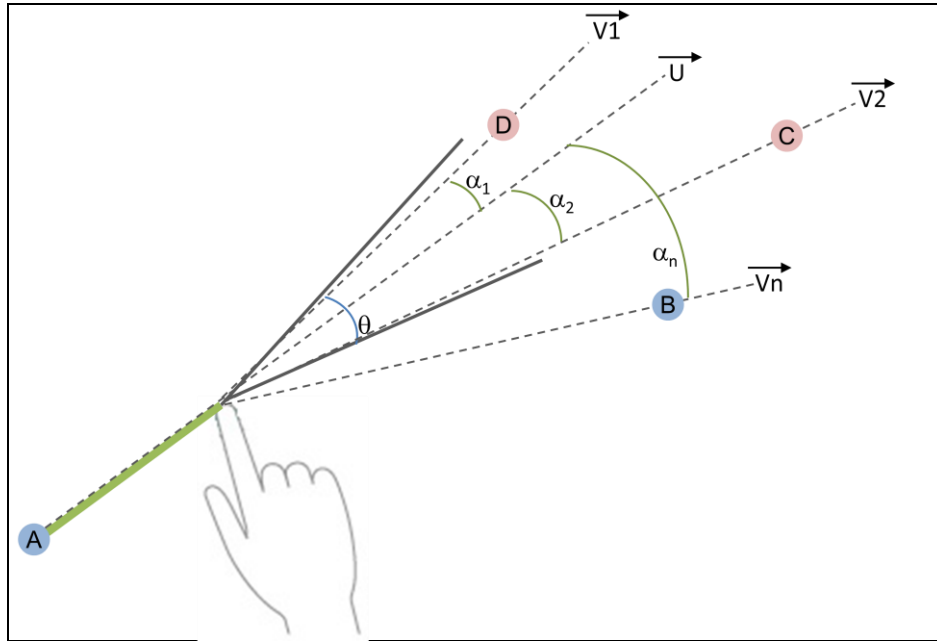


Figura 22. Vectores Directores ($U, V_1, V_2 \dots V_n$)

Al momento que el usuario levanta el dedo, se verifica si el último punto de contacto actual sobre la mesa antes de levantar el dedo ha sido sobre un nodo. Si es este el caso se crea una instancia de Edge y establece la arista entre ambos nodos, creando la arista consolidada que une el nodo origen y el destino (ver Figura 23). Si se levanta el dedo sobre cualquier otra posición en el que no haya nodos, se cancela la creación de la arista. En cualquiera de los casos, se vuelve a los valores por defecto que permitan iniciar la creación de nuevos trazos partiendo de ese nodo.

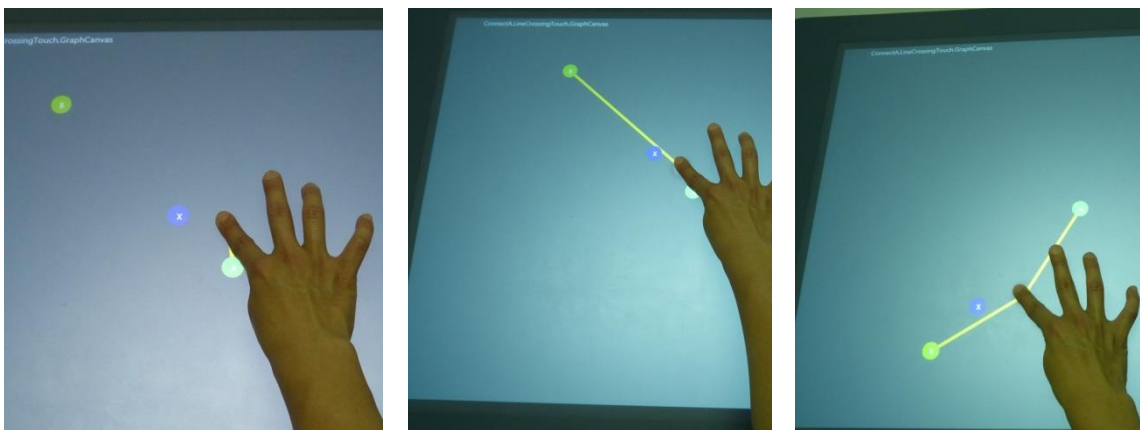


Figura 23. Técnica de Dirección

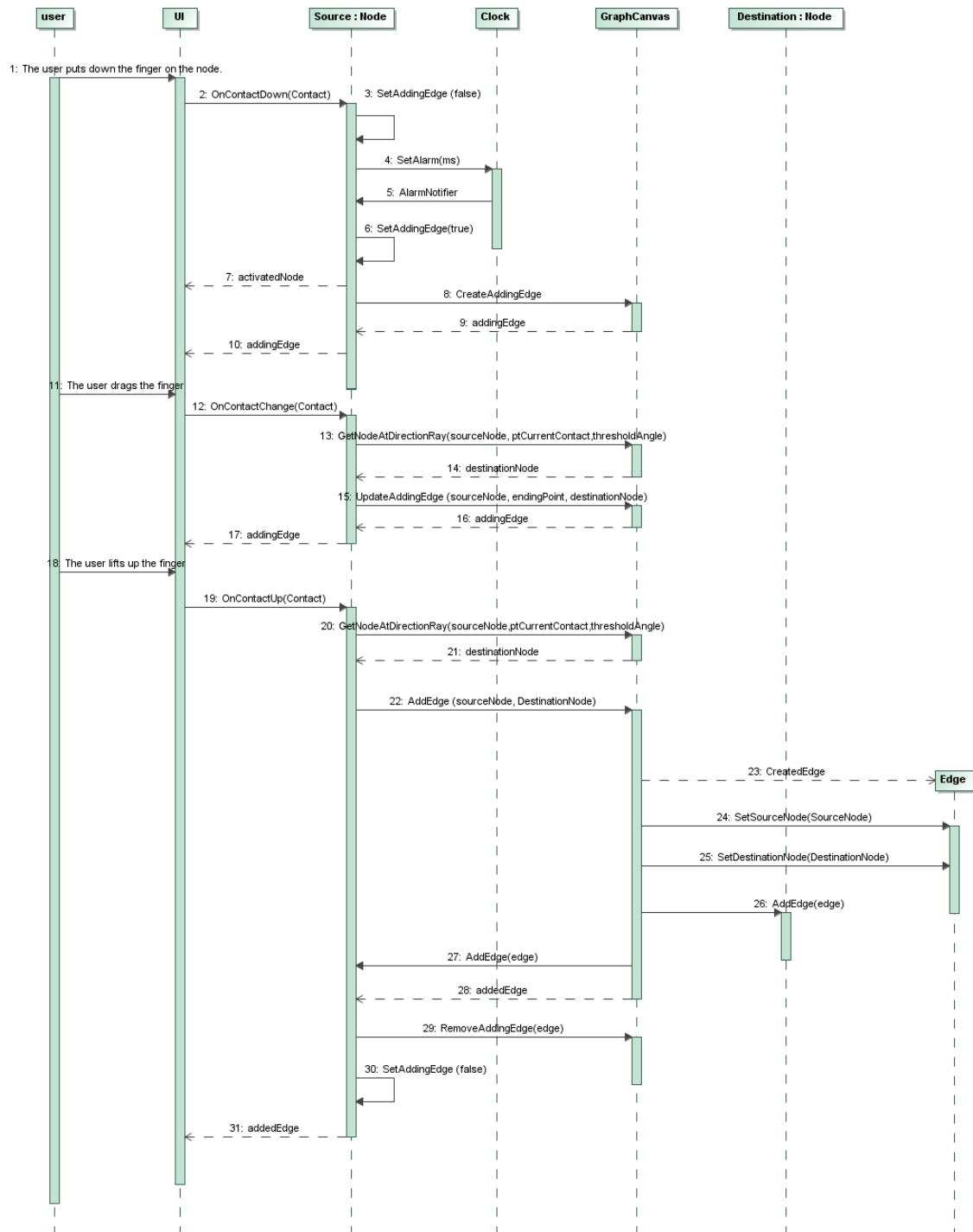


Figura 24. Creación de arista con técnica de Dirección (Diagrama de secuencia).

2.3.5 Proximidad

Con la técnica tradicional de arrastre, en ocasiones se cometen errores como consecuencia de la oclusión. Esto es así porque se levanta el dedo pensando que el nodo destino estaba ubicado debajo, cuando en realidad estaba en las inmediaciones (i.e. próximo pero no exactamente encima). Por otra parte, la

idea de poder conectar nodos a largas distancias, tal y como propone la técnica Dirección, es altamente interesante, no requiriendo llegar hasta el nodo destino pero exigiendo que éste esté dispuesto en la dirección del trazo realizado. Así pues, una alternativa a ambas, sería tener en cuenta la proximidad no direccional del dedo conector al nodo destino. Esa es la técnica que se presenta en sección, y que se denomina “Proximidad” (Figura 25).

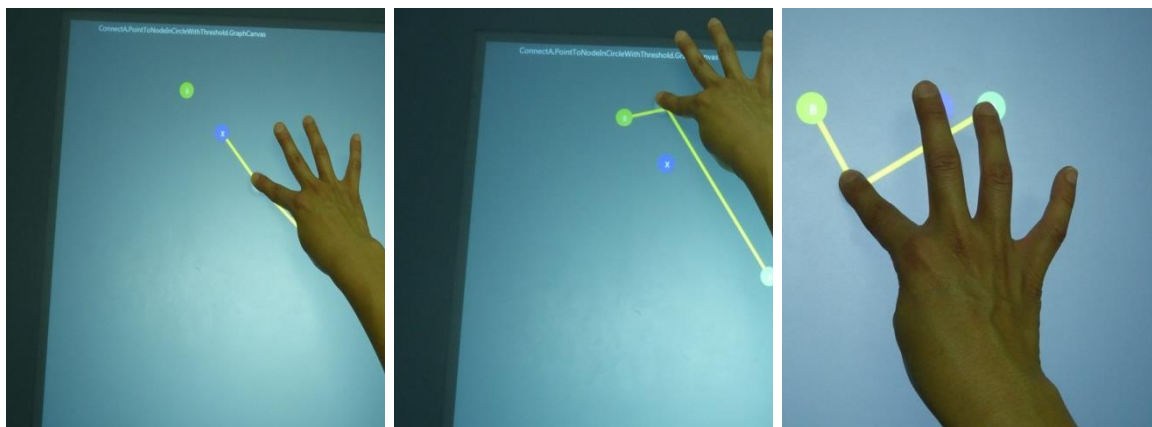


Figura 25. Técnica de Proximidad.

Cuando el usuario activa el nodo origen y arrastra en dirección al nodo destino, se genera un área circular no visible, cuyo radio es un parámetro de la técnica predefinido en la implementación. Esa área tiene su origen en el punto actual de contacto, y determina aquellos nodos candidatos para establecer la conexión. La técnica sugiere el nodo destino candidato más cercano sin que sea necesario arrastrar hasta él (Figura 26). En ese momento puede o bien levantar el dedo o el tangible, quedando los nodos unidos automáticamente (Figura 27).

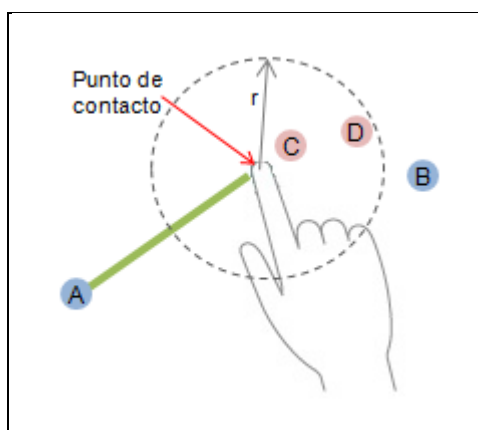


Figura 26. Conexión de nodos.

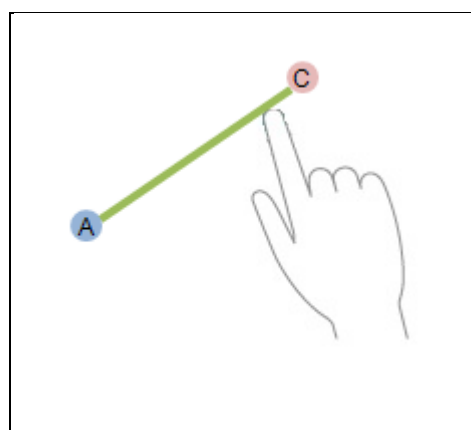


Figura 27. Arista creada.

Esta técnica se ha implementado de manera similar a las técnicas basadas en Arrastre. Tiene una particularidad importante al momento de

determinar el nodo destino, en seguida se muestra el diagrama de secuencia, con las principales interacciones entre sus objetos (ver Figura 28).

Cuando el usuario activa el nodo origen e inicia la trayectoria de arrastre, se produce el evento *OnContactChanged*, se añade la información necesaria para que el *canvas* dibuje la arista en proceso de creación, se obtiene el nodo destino a través del método *GetCloserNodeAt* y se notifica al *canvas* la línea. El método *GetCloserNodeAt* (*sourceNode*, *ptCurrentContact*, *Radius*) tiene como parámetros de entrada el nodo origen, el punto de contacto actual y el radio(*r*). Para todos los posibles nodos destino este método calcula la distancia entre el punto de contacto actual y estos, mediante la ecuación:

$$d = \sqrt{(X_{\text{nodo destino}} - X_{\text{punto de contacto}})^2 + (Y_{\text{nodo destino}} - Y_{\text{punto de contacto}})^2}$$

Una vez obtenidas las distancias, se descartan las que son mayores al radio establecido, y de éstas se obtiene la menor, que resulta el nodo destino. Será entonces cuando se actualiza la información en la tabla interna *addingEdges* mediante el método *UpdateAddingEdge*, para que el *canvas* sea consistente con la nueva posición. Esto producirá que usuario vea un trazo que sale del nodo activado de su dedo/tangible al nodo más próximo. Cuando se produce el evento (*OnContactUp*), es decir al momento que el usuario levanta el dedo, después de haber creado la arista que une el nodo origen y el destino, las propiedades vuelven a su valor por defecto.

Al ser la técnica quien sugiere los posibles nodos destino, se podría reducir la distancia de arrastre al no ser necesario llegar a contactar con el nodo destino. Esta reducción está condicionada al valor del radio asignado en la implementación de la aplicación. Si existen muchos nodos en el área donde hay que seleccionar el nodo destino y/o los nodos están muy próximos entre sí, puede dar lugar a error al conectar con el nodo destino no deseado.

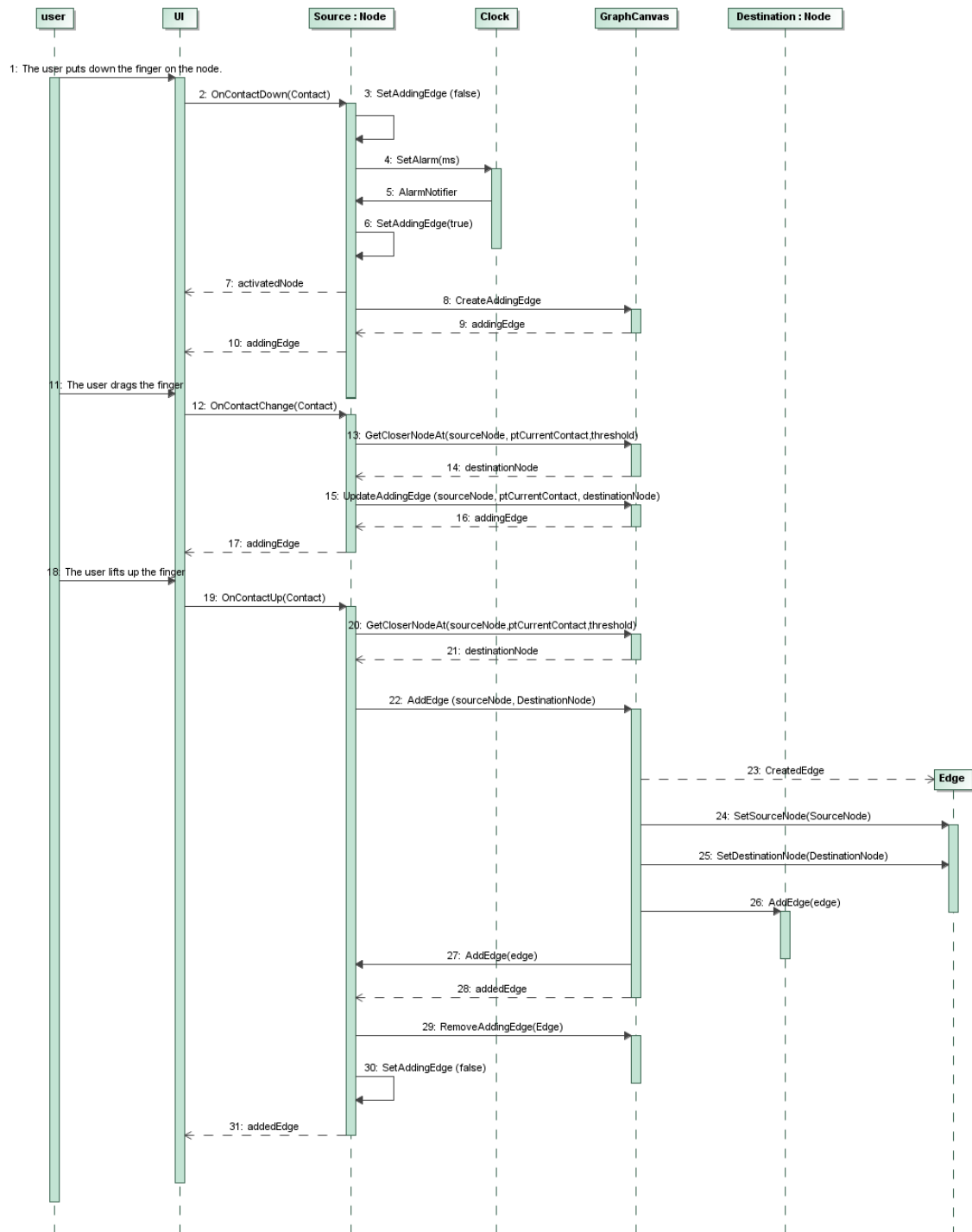


Figura 28. Creación de arista con técnica de Proximidad (Diagrama de secuencia).

2.3.6 Nodo Magnético

Esta técnica está inspirada en las interacciones utilizadas en la aplicación BATS [Hor2012], que fue diseñada para ser usada por diversos museos de historia natural en EE.UU. para establecer relaciones entre elementos. En particular, se presenta el diseño de un juego sobre una mesa interactiva, que

permite la creación de árboles que muestran la evolución de los seres vivos, para ayudar a los visitantes del museo en el aprendizaje de dicha evolución. La forma en que se conectan dos elementos en dichos árboles se basan en aproximar los elementos, actuando como imanes aquellos elementos que se mueven.

Esta técnica es similar a la denominada *proximidad*, salvo que en esta ocasión el nodo se mueve solidariamente al punto de contacto.

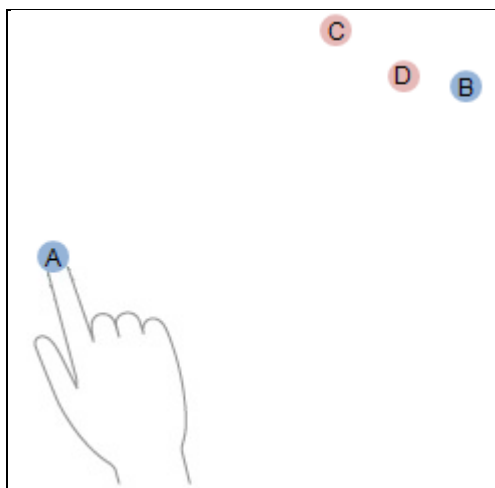


Figura 29. Activación de nodo.

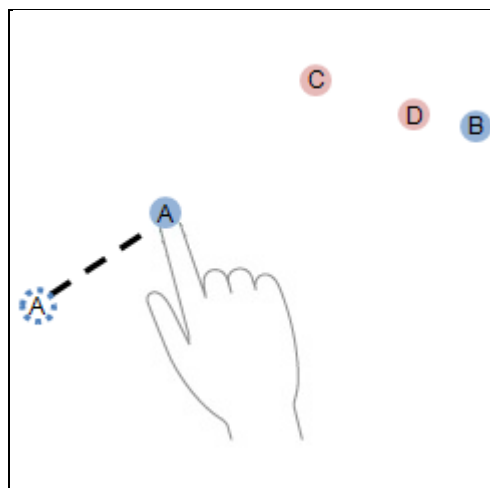


Figura 30. Movimiento de nodo.

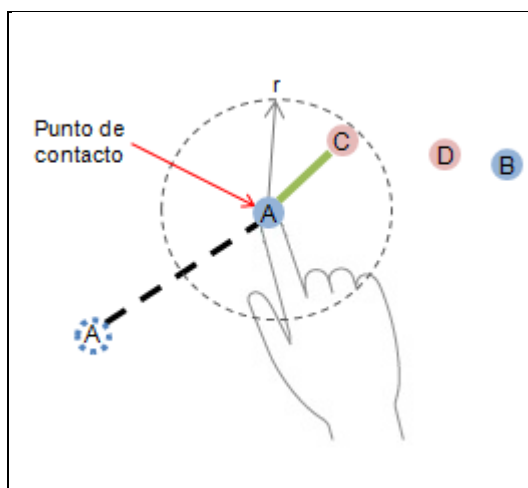


Figura 31. Proceso de conexión.

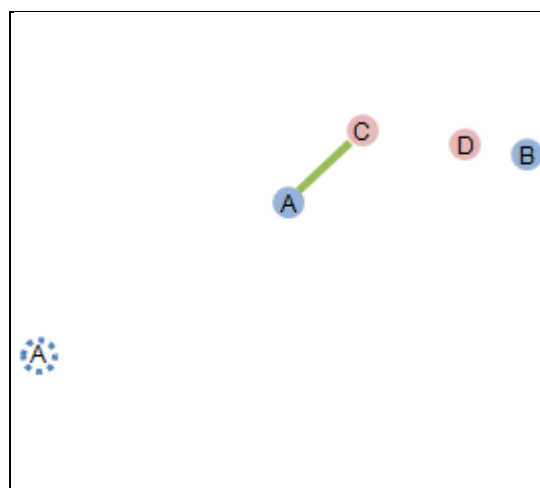


Figura 32. Arista creada.

A continuación se muestra la interacción entre los objetos de esta técnica a través de un diagrama de secuencia (Figura 33)

Cuando el usuario activa el nodo origen (Figura 29) e inicia la trayectoria de arrastre como se puede ver en la Figura 30, se produce el evento *OnContactChanged*, se añade la información necesaria para que el *canvas* dibuje la arista en proceso de creación, se obtiene el nodo destino a través del método *GetCloserNodeAt* y se notifica al *canvas* la línea. El método *GetCloserNodeAt* (*sourceNode*, *ptCurrentContact*, *Radius*) tiene como parámetros de entrada el nodo origen, el punto de contacto actual y el radio (*r*) (Ver Figura 31).

Una vez obtenidas las distancias, se descartan las que son mayores al radio establecido, y de éstas se obtiene la menor, obteniéndose de este modo el nodo destino (ver Figura 32). A continuación se actualiza la información en la tabla de *addingEdges* para que el *canvas* sea consistente con la nueva posición. Esto producirá que el usuario vea un trazo que sale del nodo activado de su dedo/tangible al nodo más próximo. Cuando el usuario levanta el dedo, se produce el evento (*OnContactUp*), después de haber creado la arista que une el nodo origen y el destino, las propiedades vuelven a su valor por defecto.

Lo más importante a resaltar en esta técnica, es que se pierde la referencia del punto de partida del nodo origen. En la figura se ha mantenido la ubicación del nodo origen A (contorno punteado) para una mejor

comprensión de esta técnica.

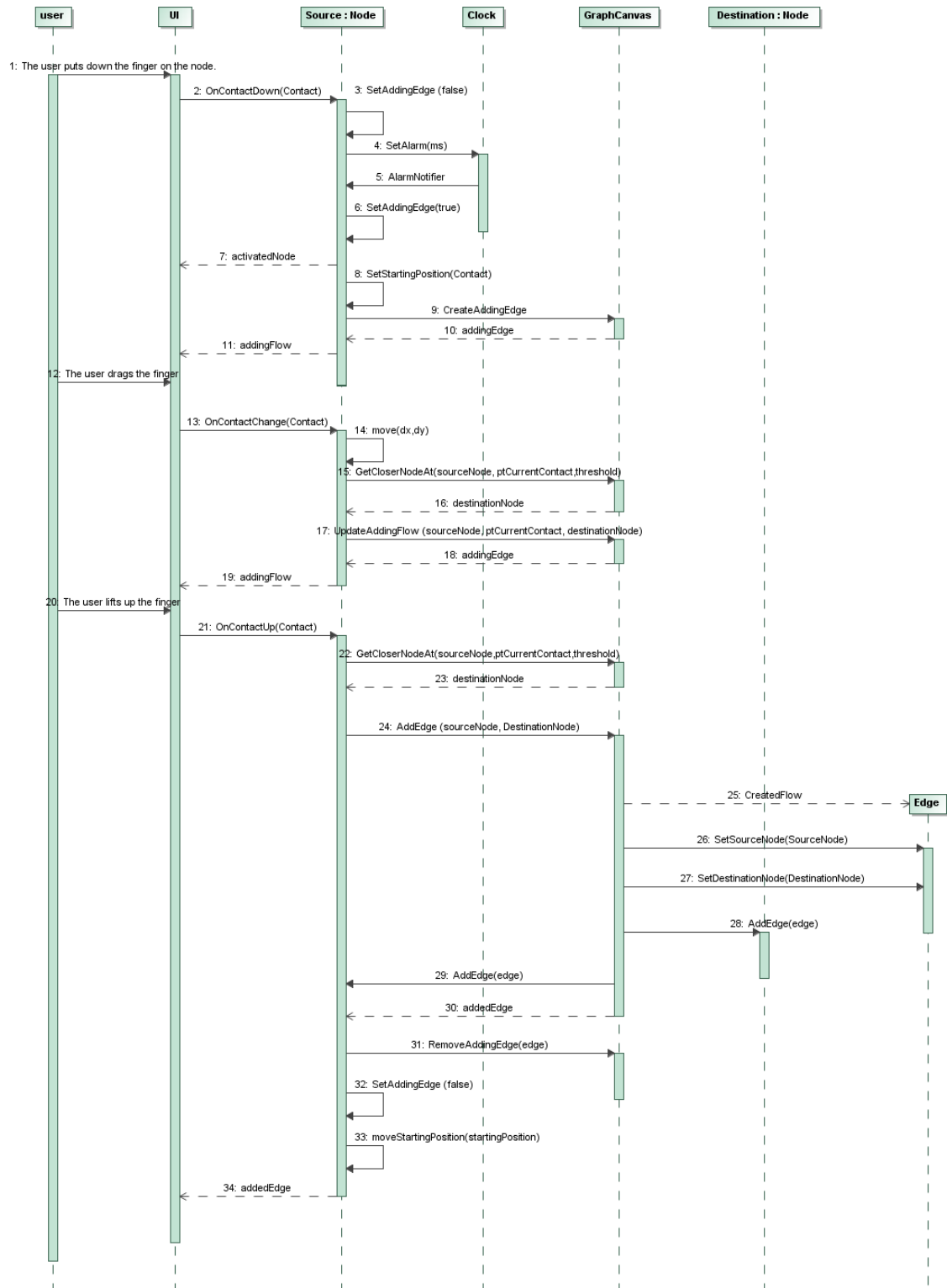


Figura 33. Creación de arista con técnica de Nodo magnético (Diagrama de secuencia).

2.4 Limitaciones y Otras Estrategias

Los diagramas de conexión de nodos (grafos), son usados en muchos ámbitos de la vida real para describir y explorar relaciones entre datos. Por ejemplo, en redes sociales, de transporte o para resolver problemas de optimización de caminos, entre otros.

Este trabajo incluye el diseño y evaluación de 6 técnicas (Arrastre, pulsación, Arrastre con Visor, Dirección, Proximidad y Nodo Magnético). De las cuáles se ha podido comprobar, que dos de ellas ya se pueden encontrar adaptadas a superficies multi-táctil. En este caso estamos hablando de Arrastre y Pulsación.

En principio un posible inconveniente que puede tener la técnica de arrastre, es la distancia, pues es necesario hacer el arrastre desde la selección del nodo origen, hasta la del nodo destino. Si esta distancia es muy grande puede ser un factor de fatiga para el usuario. Por otro lado la técnica de Pulsación suele ser bastante sencilla, debido a su funcionamiento resulta ser muy intuitiva y de fácil manejo por parte del usuario.

La técnica de Arrastre con Visor, tiene como base la implementación de la técnica de arrastre, pero tiene un añadido. Este añadido es un visor que permite en todo momento visualizar la parte ocluida por el dedo y/o brazo del usuario. Esto puede ser una ventaja para el usuario al momento de la conexión de los nodos.

Después tenemos dos técnicas asistidas (Dirección y Proximidad), que al momento de conectar con el nodo destino, sugieren al usuario el nodo más cercano dependiendo de la implementación de cada una de ellas. Esto puede ser una ventaja para el usuario, pues no tendrá que arrastrar desde el nodo inicial hasta el final, como en el caso de la Técnica de Arrastre. Por este caso el usuario puede verse beneficiado en este sentido.

Por último hemos diseñado la técnica de nodo magnético, que es muy similar a la técnica de proximidad, salvo que en este caso es el nodo inicial el que se puede mover y aproximar a el nodo destino, como se menciono anteriormente puede ser una ventaja porque no tiene que ser llevada desde el nodo inicial hasta el final. Pero si se quiere dejar el nodo inicial en su sitio, esta no es la técnica más adecuada, pues al moverlo pierde su posición original.

En general, para todas las técnicas, cuando hay gran densidad de nodos en el área donde se va a seleccionar el nodo destino, puede dar lugar a conexiones fallidas o no deseadas. El caso particular de la técnica de

pulsación, podría considerarse como una excepción a lo anterior, ya que en este caso, no requiere el arrastre del nodo.

Una posible mejora a las técnicas antes presentadas, está relacionado con la escala, si el grafo llega a ser muy grande y/o complejo, será difícil el poder visualizarlo y explorarlo a simple vista, sería necesario implementar un zoom-in o zoo-out, según sea el caso. Otra posible alternativa sería fraccionar el grafo, de manera que se pueda hacer la implementación de ventanas, que permita visualizar el grafo por secciones y que el usuario de esta manera, pueda moverlo y manipularlo a su conveniencia.

2.5 Conclusiones

En este trabajo se han explorado e implementado un conjunto de técnicas de interacción. Este conjunto de técnicas fue diseñado para ser evaluado en un marco de trabajo genérico, que nos permita representar grafos visualmente de forma muy básica sobre una mesa interactiva

La técnica de Arrastre y Pulsación, están desarrolladas a partir de técnicas ya existentes en esta tecnología. En el caso de la técnica de Nodo magnético, es una técnica adaptada que sugiere al usuario, el nodo final para realizar la conexión.

Una aportación importante a este trabajo es el diseño de técnicas asistidas como es el caso de Dirección y Proximidad, las cuáles son totalmente nuevas. En esta técnica se sugiere al usuario la conexión al nodo destino, sin que él tenga que hacer el desplazamiento completo hasta dicho nodo. Puede decirse que son técnicas novedosas, no desarrolladas hasta el momento y que persiguen dotar de cierta asistencia al proceso de establecimiento de aristas.

El objetivo de este trabajo es evaluar el rendimiento de técnicas manuales (Arrastre, Pulsación, Arrastre con Visor) y asistidas (Dirección, Proximidad) en tareas diferentes. Aunque se persigue la idea que puedan ser finalmente integradas en otras aplicaciones, tal es el caso de un editor de expresiones de flujos de datos. Esto es factible ya que el marco es lo suficientemente genérico. Además cabe destacar que con la mayoría de las técnicas, salvo la de Pulsación se pueden llevar a cabo tareas colaborativas, que es una de las aportaciones más significativas de la tecnología multitáctil.

ESTUDIO EMPÍRICO

En este trabajo se han diseñado un conjunto de técnicas de interacción para la conexión de nodos en una mesa interactiva. En este capítulo se describe la evaluación empírica a las que se han sometido estas técnicas, involucrando a usuarios sin notable experiencia con las mesas interactivas, y se describen los resultados en el experimento exploratorio realizado.

Mediante este experimento controlado se desea estudiar el tiempo y el esfuerzo que los usuarios requieren para establecer conexiones en la mesa con cada una de las técnicas diseñadas. También se va a estudiar cuán dificultosa es cada técnica, asumiendo que será menos problemática en tanto en cuanto el usuario cometa menos errores en el uso de las técnicas. Además de la técnica, otro aspecto importante a estudiar es el efecto del dispositivo de entrada dedo o tangible, para conocer su influencia en el rendimiento de cada técnica. Además, se va a analizar el provecho que los usuarios hacen de las técnicas asistidas. Como veremos más adelante, hay dos aspectos fundamentales que se han tomado en cuenta en en la tarea experimental: construir una arista entre dos nodos en el menor tiempo posible, y tratando de no cometer fallos al momento de dicha conexión.

3.1.1 Participantes

Han participado en el estudio exploratorio 8 voluntarios: 6 hombres y 2 mujeres, todos diestros, edades entre 27 y 38 años. Media =35,975, Desviación estándar=0,860. Dos de los participantes usan lentillas/gafas graduadas para corregir la visión a normal. Todos ellos utilizan ordenador personal todos los días, así como dispositivos táctiles (smartphones, tablets). Tres de ellos rara vez usan mesas interactivas (surfaces) en alguna exhibición y los otros 5 nunca han tenido experiencia alguna con mesas interactivas. Por diseño experimental, sólo se consideraron participantes que fueran diestros, debido al interés en estudiar las regiones que podrían ser de mayor/menor dificultad. De haber incluido participantes zurdos hubiera complicado innecesariamente el correspondiente análisis ya que hubiera tenido que repetirse tanto para zurdos como para diestros.

3.1.2 Equipamiento y Material

La aplicación para nuestro proyecto fue implementada en C Sharp, usando Microsoft Surface SDK versión 1.0. con las librerías específicas de Microsoft WPF. Dos mesas interactivas Microsoft Surface Unit han sido utilizadas para conducir el estudio.

Uno de los dispositivos de entrada que usamos además del dedo es un objeto etiquetado que llamaremos a lo largo del experimento como tangible, los tangibles fueron marcados con un conjunto *tagbyte* estándar compatible con Microsoft. La Figura 34 muestra uno de los tangibles creados específicamente para este estudio.

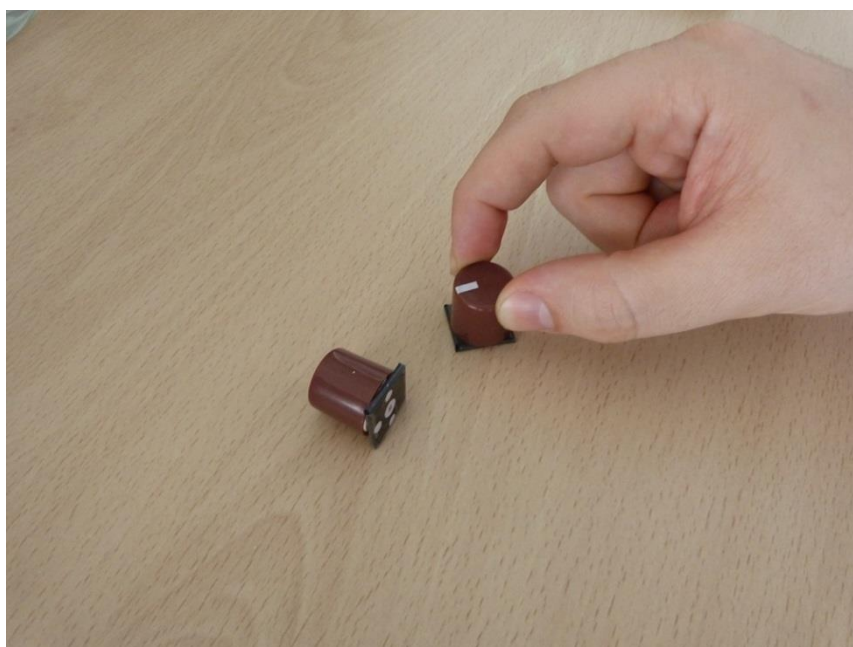


Figura 34. Uso de tangibles.

3.1.3 Método

En una sesión experimental, cada participante recibía una charla introductoria sobre la tecnología de mesas interactivas, dispositivos de entrada, y una descripción general del objetivo del experimento. La sesión se compone de un conjunto de tareas que diferían en la técnica y el modo de entrada utilizado en la misma. Dado que se ha seguido un diseño factorial, el número de tareas que debe completar cada participante es N° de técnicas $(5) \times (2) = 10$. Las técnicas que se han considerado son las de Arrastre, Arrastre con Visor, Pulsación, Dirección y Proximidad.

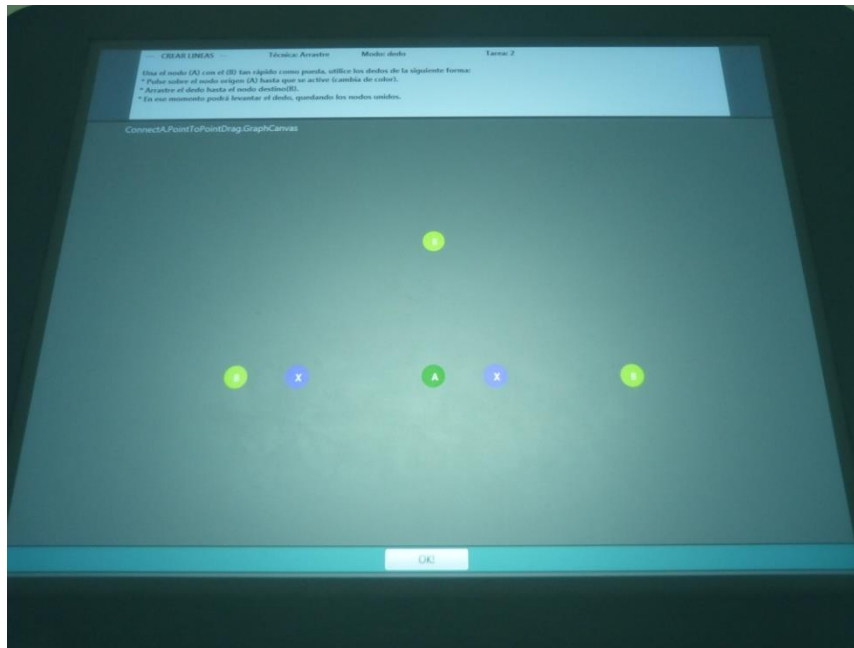


Figura 35. Instrucciones y área de entrenamiento.

Antes de cada tarea a realizar, hay un área de entrenamiento en la aplicación acompañada de una breve descripción de la técnica a utilizar (ver Figura 35). En ese entorno, el usuario se familiariza con los conceptos que están en el contexto de la tarea. Realizan las interacciones típicas de la aplicación de forma libre, pero supervisada hasta que se sientan lo suficientemente cómodos con los mecanismos de interacción de esa técnica y modo de entrada. Llegado el momento realizan la tarea de evaluación correspondiente.

Se exhorta a los participantes a que realicen las uniones tan rápido como les sea posible y con la mayor precisión. El entorno software de experimentación registra toda la información referente a los contactos (instante, posición, etc.) así como aquellos sucesos relevantes que son necesarios para íntegramente reconstruir todas las acciones del usuario. De esta información registrada se podrá extraer información relativa el tiempo empleado para llevar a cabo cada conexión, las trayectorias seguidas, así como las interacciones fallidas.

Al finalizar cada tarea el participante debe responder una serie de cuestionarios basados en el NASA Task Load Index instrument [Har2006]. Tras su compleción, el participante dispone del área de entrenamiento y las instrucciones de la siguiente tarea.

3.1.4 Tarea

Como ya se ha anticipado, la tarea principal consiste en establecer la conexión entre el nodo identificado como A con el nodo identificado como B lo más rápido posible y tratando de no fallar en las interacciones.

Existe una configuración, que va cambiando para cada participante (ver Figura 36) que determina la actividad, técnica, dispositivo de entrada y las pruebas (*trials*) a realizar. La primera línea indica la actividad, para nuestro estudio solo existe la tarea de conectar. Enseguida aparece la descripción general de las tareas, donde se indica la técnica a usar, el modo del dispositivo de entrada (dedo, tangible) y la cantidad de conexiones a realizar, esto es transparente para el usuario. Después están las pruebas a realizar en el orden allí establecido, que la componen tres elementos: 1) orientación de los nodos (Norte, Sur, Este, Oeste, Noreste, Sureste, Suroeste y Noroeste); 2) el siguiente parámetro indica la distancia respecto al nodo origen. Se han considerado dos radios, de 364 y 182 respectivamente que formarán dos anillos concéntricos al nodo origen (A). Finalmente hay un tercer parámetro que hace referencia a un nodo obstáculo dispuesto en la trayectoria entre el nodo A y el nodo B. Dicho nodo obstáculo se puede encontrar a una de las dos distancias diferentes, llamadas en este caso como N (la más cercana) y F (la más alejada), que se sitúan exactamente en el punto correspondiente al 30% y 70% de la longitud del segmento AB respectivamente. La razón para incluir un obstáculo viene motivada por que se pretende dificultar o entorpecer el funcionamiento de las técnicas asistidas, que son realmente aquellas sobre las que queremos conocer más detalles.

```

test3.cfg
Connect
Technique  MagneticConnector  F  32
SW  L  N
S   S  F
SW  S  N
SE  L  N
E   S  N
E   L  N
N   S  N
N   S  F
NE  S  N
NW  L  N
N   L  F
SE  S  F
SW  S  F
S   L  N
W   S  N
NE  L  N
SE  L  F
NE  L  F
S   S  N
NW  S  F
NE  S  F
NW  L  F

```

Figura 36. Archivo de configuración para cada usuario.

El participante une el nodo (A), desde el centro de la superficie táctil, con otro nodo (B) situado en la posición que ha sido extraída del fichero de configuración. Cada vez que el usuario realiza con éxito una conexión entre el nodo (A) y (B) en ese orden respectivamente, otra posición es leída del archivo de configuración.

El nodo (B) puede aparecer en 16 posiciones diferentes no repetidas mientras que el obstáculo puede aparecer en dos localizaciones relativas. Por tanto, en una tarea dada, el participante tiene que realizar una serie de 32 conexiones.

3.1.5 Procedimiento

Este experimento fue realizado por cada participante, usando 2 modos diferentes como dispositivo de entrada: dedo y tangible. El orden en que se administraron las tareas (i.e. las técnicas y los modos de entrada) se establecieron de manera aleatoria con el propósito de bloquear un posible factor de aprendizaje.

3.1.6 Resultados

3.1.6.1 Generalidades

Las variables que fueron medidas en este experimento, son el tiempo de conexión entre los nodos y el número de interacciones necesarias para lograr el objetivo en cada técnica. Los dispositivos de entrada usados son el dedo y tangible dependiendo del archivo de configuración.

La Figura 37 muestra los tiempos promedio de conexión entre dos nodos, usando cada una de las respectivas técnicas y modo de interacción. Como se puede observar, los participantes obtuvieron mejores tiempos de conexión en casi todas las técnicas, cuando fueron realizadas con el dedo (no tangible) que con el modo tangible, a excepción de la técnica de pulsación. En lo que se refiere a las técnicas no asistidas (arrastre, arrastre con visor y pulsación), observamos claramente que la técnica de arrastre con visor fueron las que necesitaron un mayor tiempo (dedo=44.95; tangible=48.67 seg). Esto puede ser debido a que la técnica requería una mayor habilidad para manejar adecuadamente el visor, y especialmente por el hecho que los usuarios no sacaron realmente partido al uso del visor. Pese a haber tenido un breve entrenamiento en el que los usuarios entendieron el uso del visor y lo pusieron en práctica, se observó que los usuarios en general dudaban en utilizar el visor o si seguir manejando como si fuera una técnica rasa de arrastre, provocando una mayor lentitud en sus interacciones. Todo apunta a que como consecuencia de que los usuarios no adquirieron realmente el uso efectivo del visor, en combinación con el hecho de que realmente la tarea

ESTUDIO EMPÍRICO

podía llevarse a cabo sin hacer caso al visor, éste resultó ser más una distracción que una herramienta útil. Mayor entrenamiento con la técnica seguramente podría producir mejores resultados para esta técnica del visor.

En términos generales, la técnica que presenta mejores resultados en términos de tiempo requerido es la técnica de Pulsación (dedo=24.12 seg; tangible=22.22 seg). Esto puede ser debido a esta técnica es mucho más intuitiva y de fácil manejo para el participante.

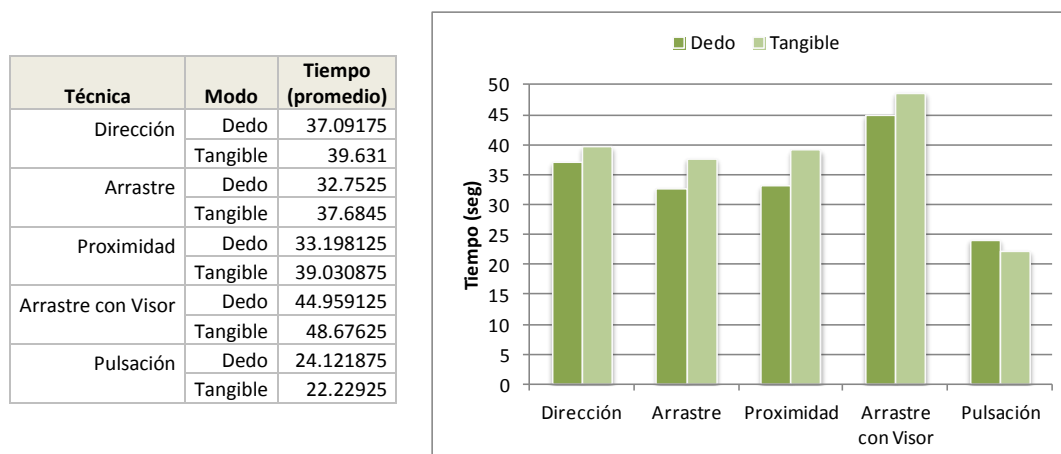


Figura 37. Tiempo promedio de conexión entre nodos por técnica y modo de interacción..

La Figura 38 muestra el valor promedio del número de de conexiones fallidas realizadas con su respectiva técnica y modo de interacción. Es claramente visible que la técnica con menor número de conexiones fallidas es la de pulsación (dedo=0.25; tangible=0.875). Se observa también que todas las técnicas son menos propensas a fallos cuando éstas son realizadas con el dedo que con un modo tangible. En las técnicas asistidas se observó una diferencia, mucho más significativa entre los modos de interacción de dirección y proximidad (aproximadamente una proporción de 3,5 del modo dedo respecto al modo tangible).

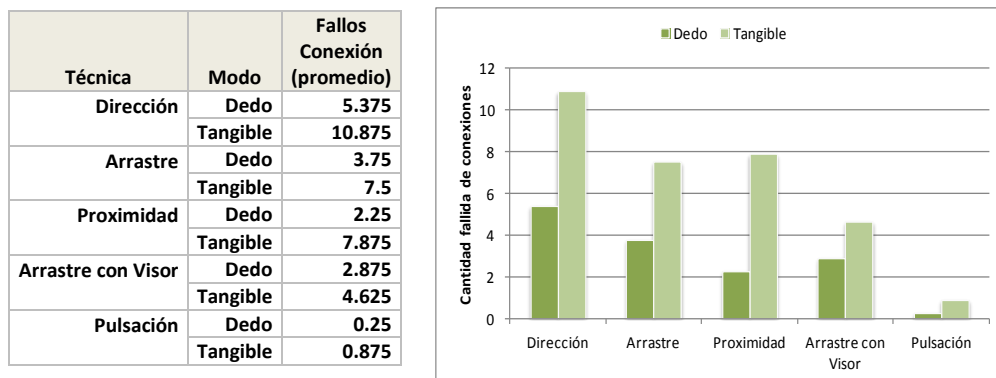


Figura 38. Valor promedio de los fallos de conexión entre nodos.

3.1.6.2 Detalles sobre el efecto de la orientación

Con el fin de investigar si la localización del nodo destino en la mesa interactiva influye en los tiempos de conexión, se ha analizado cada una de las 8 posibles localizaciones (norte (N), noreste (NE), este (E), sureste (SE), sur (S), suroeste (SO), oeste (O) y noreste (NO)), tal como se presenta en la Figura 39.

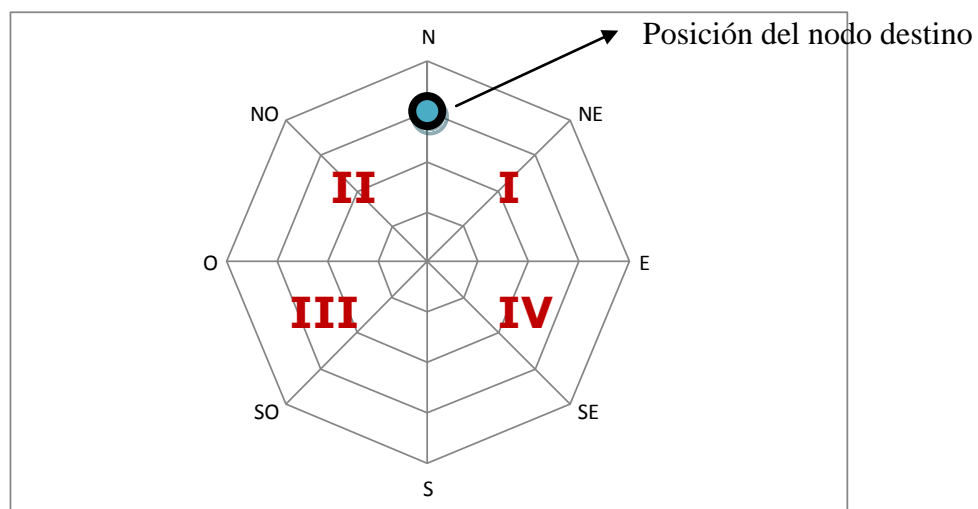


Figura 39. Diferentes posiciones donde se ubica el nodo destino.

En la Figura 40 se puede observar que en modo tangible, los participantes obtuvieron los mejores tiempos cuando realizaban las conexiones con el nodo destino en el tercer cuadrante: suroeste y oeste. Sin embargo, para el modo dedo no hay un claro patrón. Por ejemplo, comparando los dos mejores tiempos para las técnicas de arrastre y arrastre con visor, se observa que los participantes conectaron los nodos, con cierta mayor facilidad cuando el nodo destino estaba localizado únicamente en la orientación sur.

En cambio la orientación suroeste resultó ser relativamente mejor para las técnicas de arrastre con visor y pulsación. Observamos también que para la técnica arrastre, los participantes obtuvieron un mejor tiempo en la

ESTUDIO EMPÍRICO

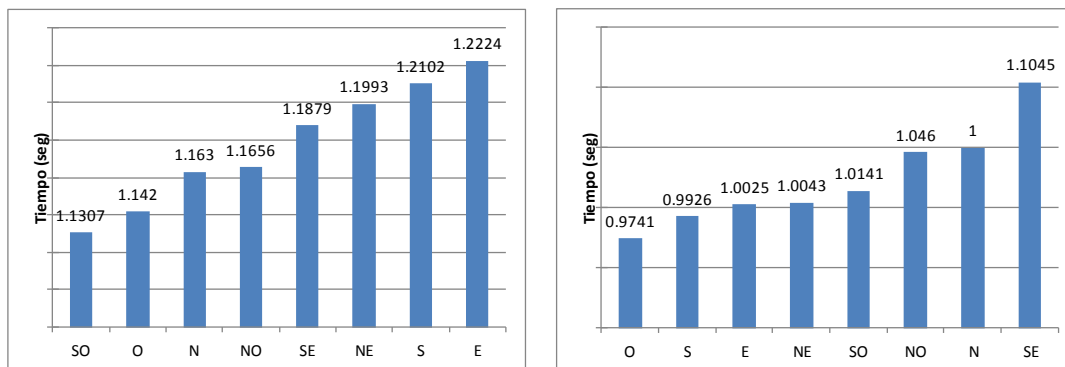
orientación oeste, mientras que para las otras dos técnicas no asistidas esta orientación no figura entre los dos mejores tiempos.

Es importante remarcar que para la técnica arrastre en modo tangible, el tiempo mayor (tiempo = 1.2102 seg) se ha obtenido cuando el nodo destino está ubicado en el sur, mientras que para el modo dedo ocurre lo contrario (tiempo= 0.9926 seg). Probablemente sea porque el participante al hacer este desplazamiento al nodo destino en la orientación sur, sea de alguna manera un movimiento habitual o cotidiano, que ha sido entrenado inconscientemente.

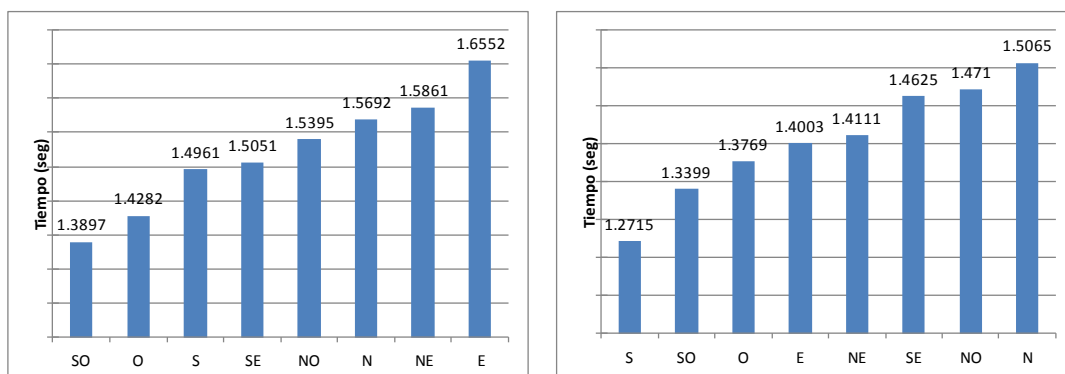
En las técnicas de arrastre y arrastre con visor, en el modo tangible, se observó un mayor tiempo cuando la posición del nodo destino estaba en el este. En las técnicas de arrastre con visor y pulsación, en modo dedo el tiempo mayor se obtuvo en la orientación norte.

Tiempo promedio por técnica y modo de conexión entre nodos.	
Técnicas no asistidas	
Tangible	Dedo

Arrastre



Arrastre con visor



Pulsación

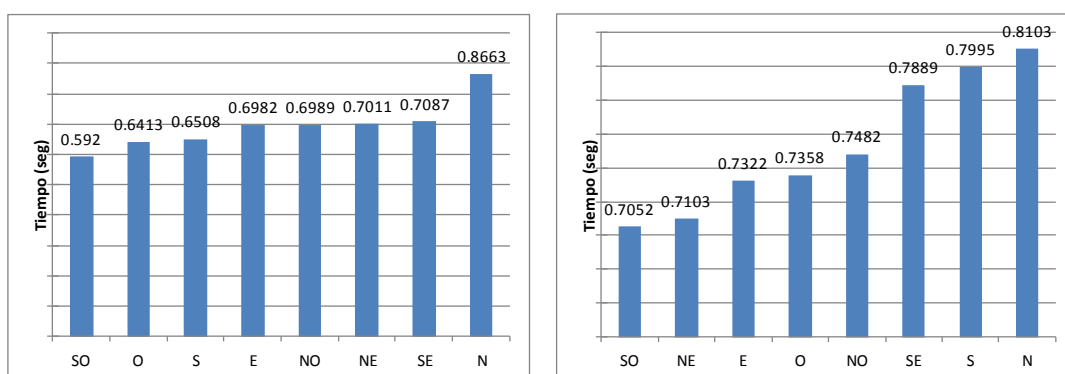


Figura 40. Tiempo promedio por técnica y modo de conexión entre nodos.

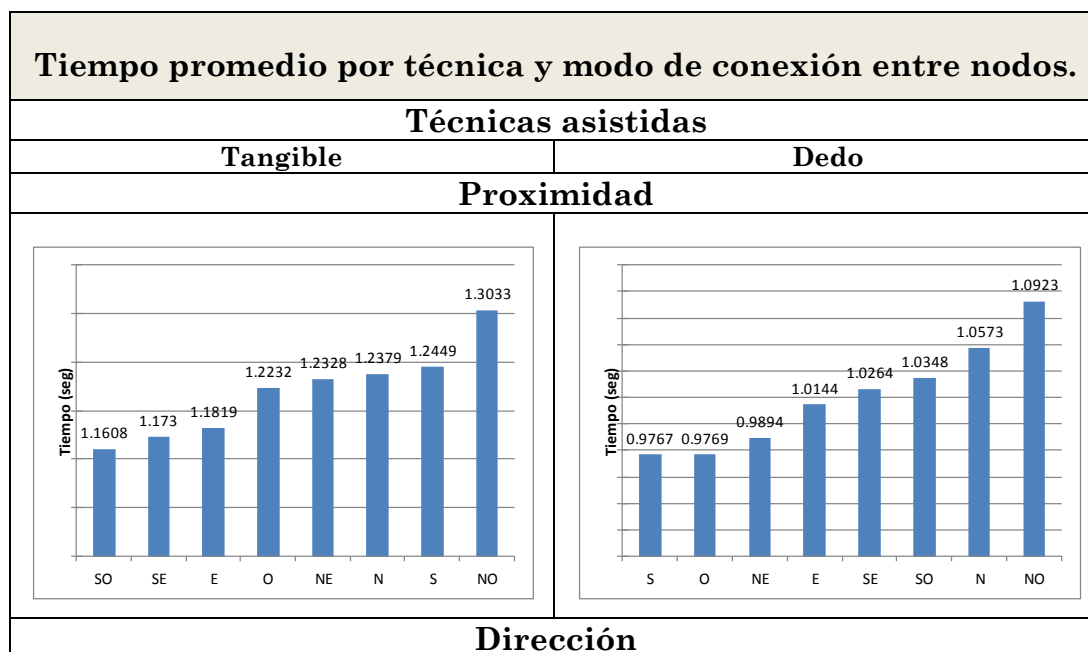
Por lo general, para las técnicas no asistidas podríamos decir que las áreas que requieren menor tiempo para conectar nodos son las relativas al S, SO, O. Esto se puede entender así por el hecho de requerir movimiento más cercanos al cuerpo (i.e. a la posición en la que está sentado el usuario) en combinación al hecho de que esa región no presenta tanta oclusión como en

ESTUDIO EMPÍRICO

las regiones que tienen relación con el este. Las regiones del norte se observaron que podían llegar a ser problemáticas por el hecho que el usuario tenía que estira más el brazo (por ser puntos más alejados de la posición en la que sentaba).

En la Figura 41 para las técnicas asistidas, se puede observar que los mejores tiempos obtenidos en modo tangible, no muestran un claro patrón. Por ejemplo, comparando los dos mejores tiempos para éstas, se observa que los participantes conectaron los nodos, con cierta mayor facilidad cuando el nodo destino estaba localizado en la orientación sureste. Para el modo dedo comparando también los dos mejores tiempos, se observa que las conexiones realizadas con cierta mayor facilidad cuando el nodo destino está orientado al oeste. Observamos también que para la técnica proximidad, los participantes requirieron un mayor tiempo en la orientación noroeste, tanto para tangible (tiempo= 1.3033 seg) como para dedo (tiempo= 1.0923 seg).

Es importante remarcar que para la técnica asistida dirección en modo dedo, el tiempo mayor (tiempo = 1.2431 seg) se ha obtenido cuando el nodo destino está ubicado en el sureste, mientras que para el modo tangible ocurre lo contrario (tiempo= 1.1535 seg). En la técnica de dirección, en ambos modos, comparando los dos mejores tiempos, se observa un menor tiempo cuando la posición del nodo destino estaba en el noroeste.



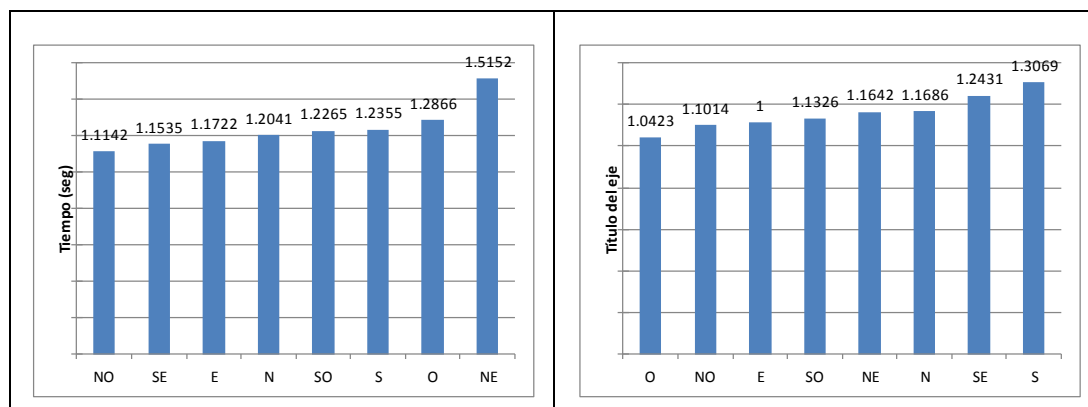


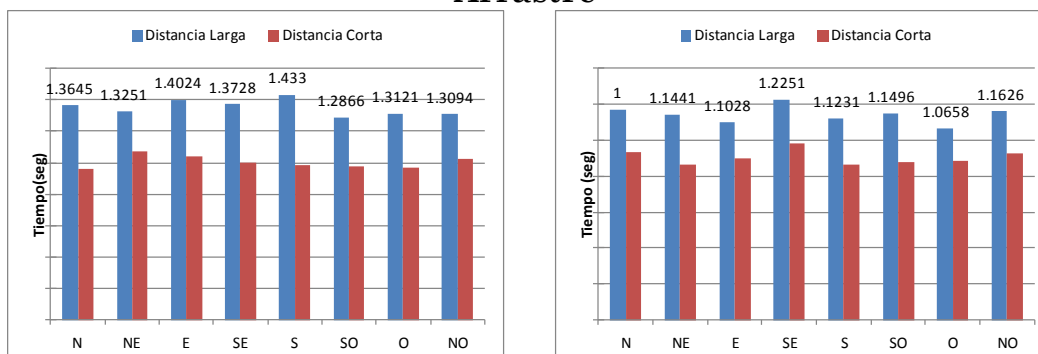
Figura 41. Tiempo medio por técnica y modo de conexión entre nodos.

Comparando las técnicas asistidas vs las no asistidas, en general se observó que los tiempos menores estaban en las orientaciones suroeste, este y oeste. En la Figura 42, se puede observar que para las técnicas no asistidas (arrastre, arrastre con visor, pulsación), en ambos modos (dedo, tangible) el tiempo obtenido en las distancias largas siempre es mayor respecto del tiempo obtenido en las distancias cortas, como era de esperar. Observando los cuatro mejores tiempos en modo tangible para la distancia larga, los participantes obtuvieron los tiempos mínimos, cuando realizaban las conexiones con el nodo destino en el tercer cuadrante (suroeste, oeste). Sin embargo, para el modo dedo no hay un claro patrón. Por ejemplo, comparando los 4 mejores, se observa que los participantes conectaron los nodos, con cierta mayor facilidad cuando el nodo destino estaba localizado en las orientaciones: este, noreste, suroeste y sureste. Es importante remarcar que para las 3 técnicas no asistidas (arrastre, arrastre con visor, pulsación) en modo tangible, con una distancia larga, el tiempo mínimo se ha obtenido cuando el nodo destino está ubicado en el suroeste.

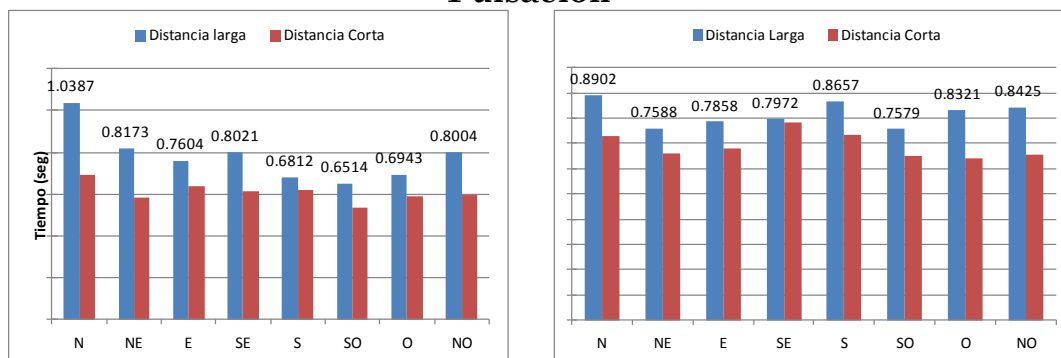
Es importante remarcar que para distancias cortas en ambos modos (dedo, tangible), comparando los 4 mejores, el tiempo menor se ha obtenido cuando el nodo destino está ubicado en el suroeste y oeste. En estos resultados podemos observar que los tiempos mínimos en las técnicas no asistidas se obtienen en la orientación suroeste y oeste, sin importar la distancia y el modo.

Tiempo medio por distancias de conexión entre nodos	
Técnicas no asistidas	
Tangible	Dedo

Arrastre



Pulsación



Arrastre con visor

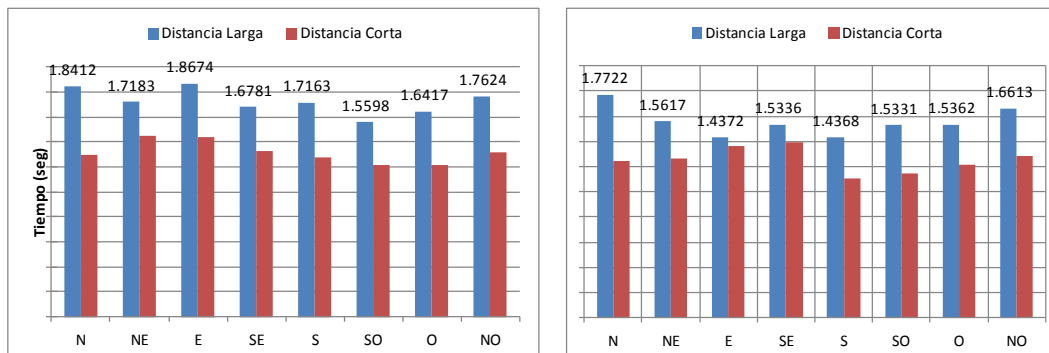


Figura 42. Tiempo promedio por distancias de conexión entre nodos .

En la Figura 43, se puede observar que para la técnica asistida proximidad, en ambos modos (dedo, tangible) el tiempo obtenido en las distancias largas siempre es mayor respecto del tiempo obtenido en las distancias cortas, como era de esperar. Sin embargo esto no ocurre en la técnica de dirección, en ambos modos el tiempo menor obtenido es en las distancias largas. Eso puede ser como consecuencia de la dificultad que tenían los usuarios para conseguir que la técnica sugiriera rápidamente la conexión correcta debido a la proximidad del nodo X. Cabe recordar que este nodo X está puesto en la trayectoria directa entre los nodos A y B, lo que significa que este sería un escenario muy desfavorable para esta técnica de dirección.

Observando los cuatro mejores tiempos para la técnica proximidad en ambos modo para la distancia larga, los participantes obtuvieron los tiempos mínimos, cuando realizaban las conexiones con el nodo destino en el noreste y en el este. En referencia a los tiempos máximos se obtuvieron en las orientaciones norte y noroeste. Respecto a la distancia corta se puede decir que en esta técnica asistida (proximidad), la orientación que les resultó más compleja a los usuarios fue la noroeste, pues en ambos modos se obtuvo un tiempo mayor.

Para la técnica de dirección en modo tangible y para una distancia larga observamos que los mejores tiempos se obtienen cuando el nodo destino está ubicado hacia la posición noroeste y suroeste.

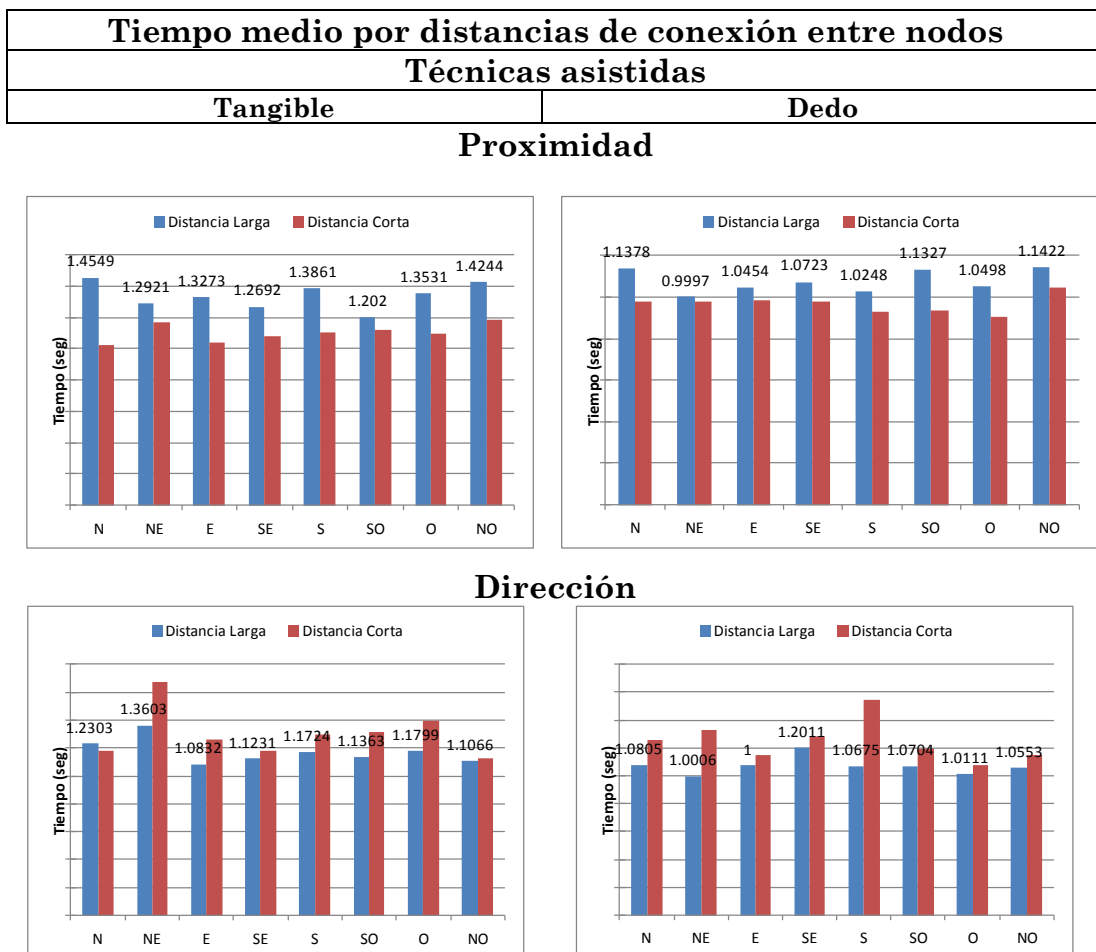


Figura 43. Tiempo promedio por distancias de conexión entre nodos .

En esta sección evaluaremos el valor promedio de conexiones fallidas por modo para las técnicas asistidas y no asistidas. Entendemos por conexión fallida, cuando el usuario al finalizar la conexión en el nodo destino por algún motivo no lo ha seleccionado, es decir ha seleccionado un área diferente o un nodo diferente al nodo destino.

ESTUDIO EMPÍRICO

En la Figura 45 se puede observar que los participantes obtuvieron los mejores tiempos en modo tangible, cuando realizaban las conexiones con el nodo destino en el tercer cuadrante: suroeste y oeste. Sin embargo, para el modo dedo, comparando las cuatro orientaciones con menores fallos de conexión, se observa que los mejores resultados se obtuvieron en la orientación sur y suroeste.

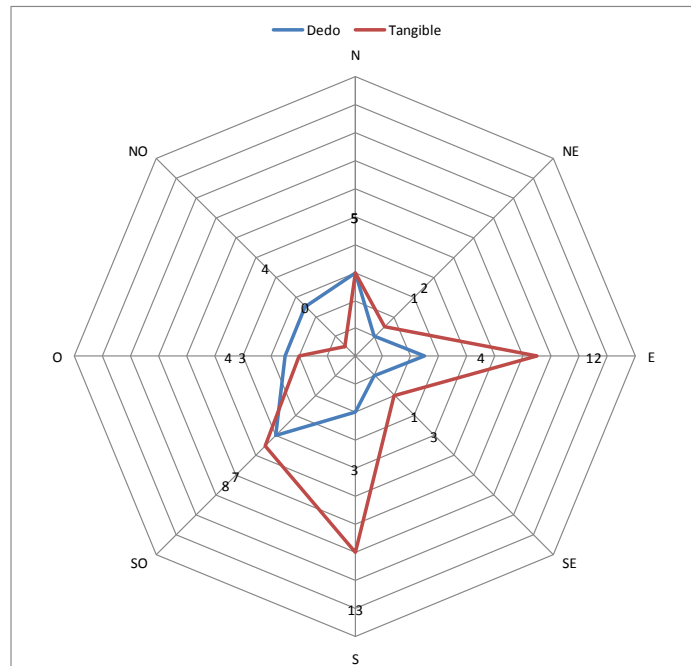
En cambio las orientaciones sur, este y norte para el modo tangible, resultaron con mayor número de conexiones fallidas, mientras que en el modo dedo en el oeste es donde se contabilizaron el mayor número de conexiones fallidas.

Valor medio de conexiones fallidas por técnica y modo Técnicas no asistidas

a) Arrastre

Localización	Dedo
N	5
NE	1
E	4
SE	1
S	3
SO	7
O	4
NO	4

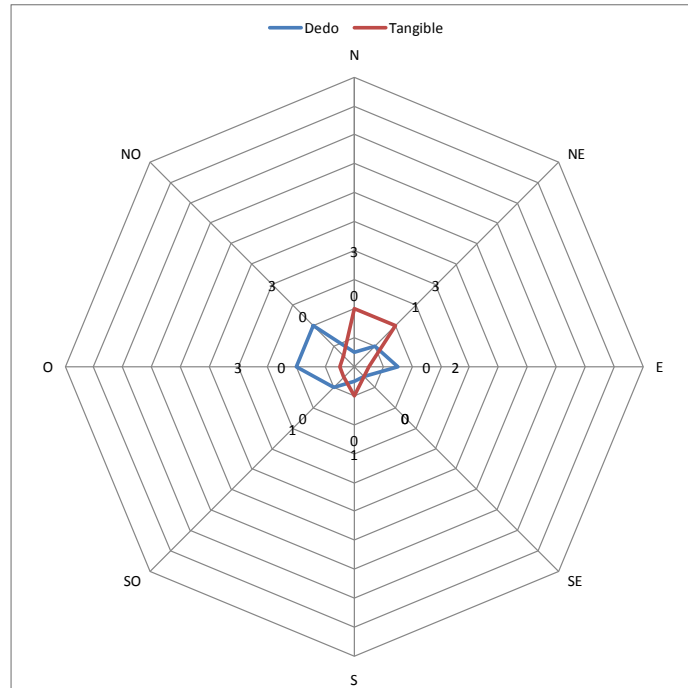
Localización	Tangible
N	5
NE	2
E	12
SE	3
S	13
SO	8
O	3
NO	0



b) Pulsación

Localización	Dedo
N	0
NE	1
E	2
SE	0
S	0
SO	1
O	3
NO	3

Localización	Tangible
N	3
NE	3
E	0
SE	0
S	1
SO	0
O	0
NO	0



c) Arrastre con Visor

Localización	Dedo
N	3
NE	3
E	0
SE	1
S	0
SO	3
O	6
NO	7

Localización	Tangible
N	9
NE	8
E	7
SE	0
S	5
SO	3
O	3
NO	2

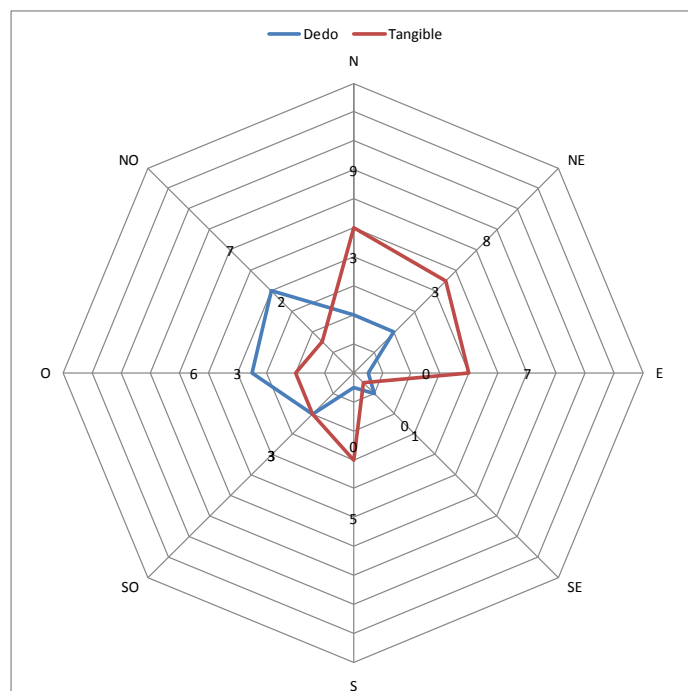


Figura 44. Valor promedio de conexiones fallidas por técnica y modo.

En la Figura 45 se puede observar que los participantes obtuvieron los mejores resultados en modo tangible, cuando realizaban las conexiones con el nodo destino en la orientación noroeste y sureste. Sin embargo, para el modo dedo, comparando las cuatro orientaciones con menores fallos de conexión, se observa que los mejores resultados se obtuvieron en la orientación oeste, norte y noroeste.

ESTUDIO EMPÍRICO

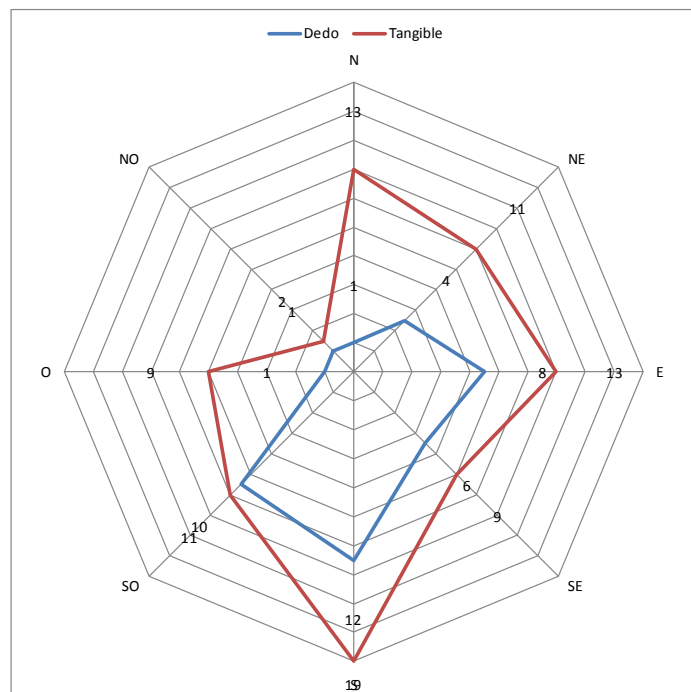
Comparando las cuatro orientaciones con mayores fallos de conexión, se observa que para el modo tangible, las orientaciones con mayor número de fallos de conexión son sur, este, norte y noreste, mientras que en el modo dedo en el sur, suroeste y este es donde se contabilizaron el mayor número de conexiones fallidas.

Es importante remarcar que para la técnica asistida dirección en modo tangible siempre es mayor el número de fallos de conexión, en todas las orientaciones respecto al modo dedo (ver Figura 45 a).

Valor promedio de conexiones fallidas por técnica y modo Técnicas asistidas a) Dirección

Localización	Dedo
N	1
NE	4
E	8
SE	6
S	12
SO	10
O	1
NO	1

Localización	Tangible
N	13
NE	11
E	13
SE	9
S	19
SO	11
O	9
NO	2



b) Proximidad

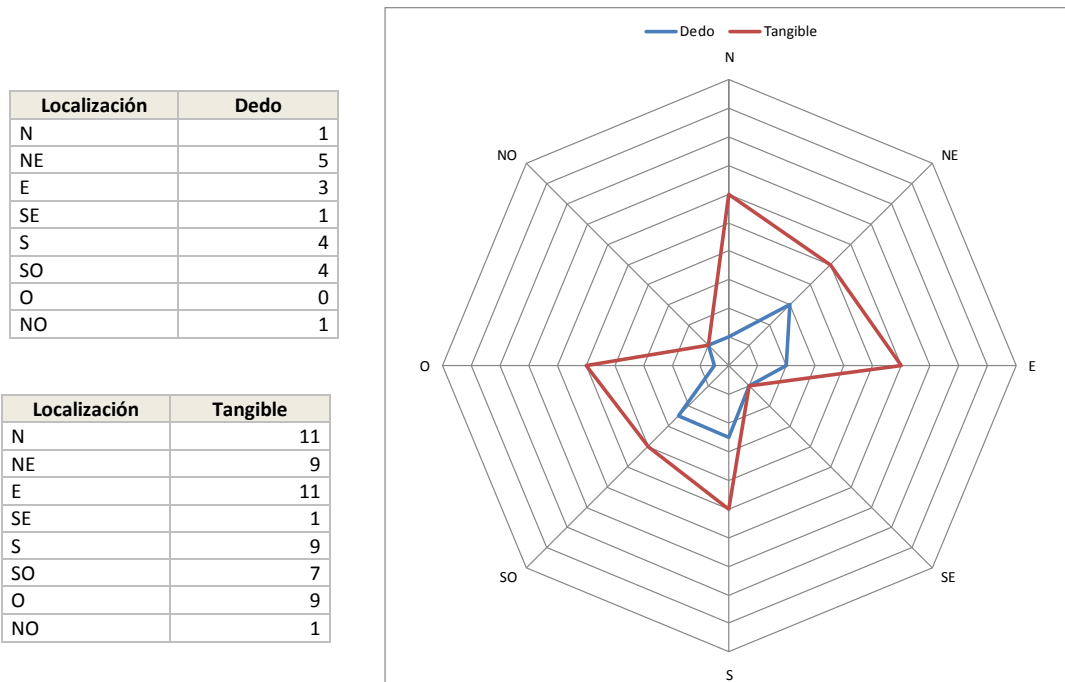


Figura 45. Valor medio de conexiones fallidas por técnica y modo.

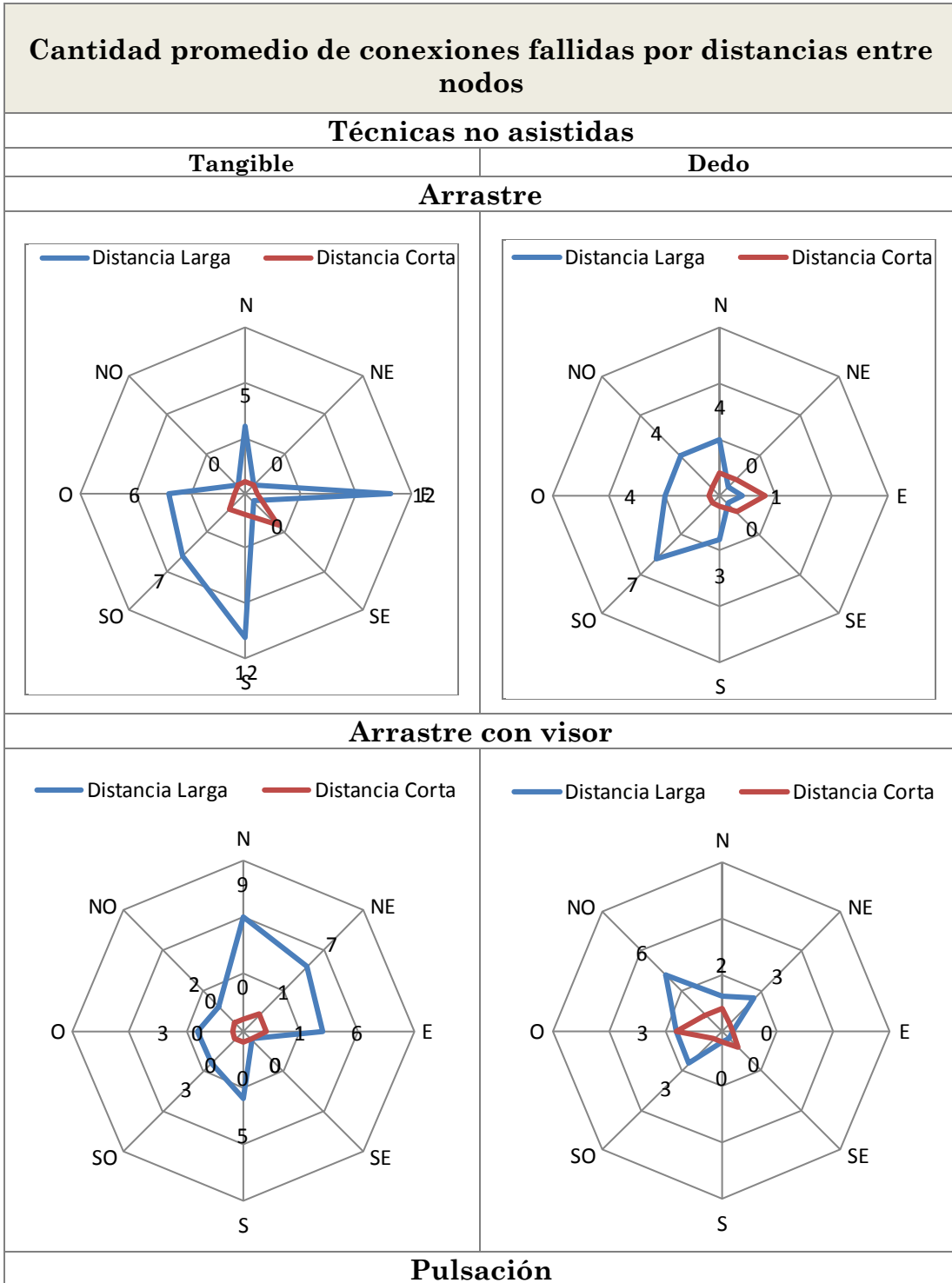
En la Figura 46 se puede observar para las técnicas no asistidas, independientemente del modo (dedo, tangible), los participantes obtuvieron menor cantidad de conexiones fallidas en la distancia corta, cuando realizaban las conexiones con el nodo, como era de esperarse.

Comparando las cuatro orientaciones con menores fallos de conexión, se observa que para la distancia larga y el modo tangible, las orientaciones con menor número de fallos de conexión es sureste, mientras que en el modo dedo en el sur y sureste es donde se contabilizaron el menor número de conexiones fallidas. Para una distancia corta y modo tangible la orientación con menor número de fallos de conexión entre nodos es el oeste, mientras que para el modo dedo es el sur.

En cualquier caso, tal y como ya habíamos adelantado en la sección de generalidades, el uso de tangibles introdujo un mayor número de fallos en términos generales. Durante la realización de los experimentos, se observó que el tangible presentaba un problema a la hora de realizar movimientos rápidos. El sistema de seguimiento del tangible requiere identificar el tag, y puede trabajar a cierta velocidad (frecuencia). Cuando se desplaza un tangible rápidamente ocurre que el sistema de visión emblanquece el área de la etiqueta del tangible, perdiendo en consecuencia el seguimiento del objeto. Esto inevitablemente hizo que los usuarios tuvieran que ir algo más lentos en el uso del tangible, y cuando se excedía la rapidez con la que el sistema puede trabajar, se producían fallos. También es razonable ver que a mayor distancia, mayor probabilidad para perder el seguimiento del tangible, porque hay que recorrer mayor distancia, tal y como muestran los resultados de las gráficas. Para tratar de mitigar este efecto, se requeriría

ESTUDIO EMPÍRICO

de mayor esfuerzo en la programación de las técnicas con tangible para permitir una pérdida momentánea del seguimiento de la etiqueta.



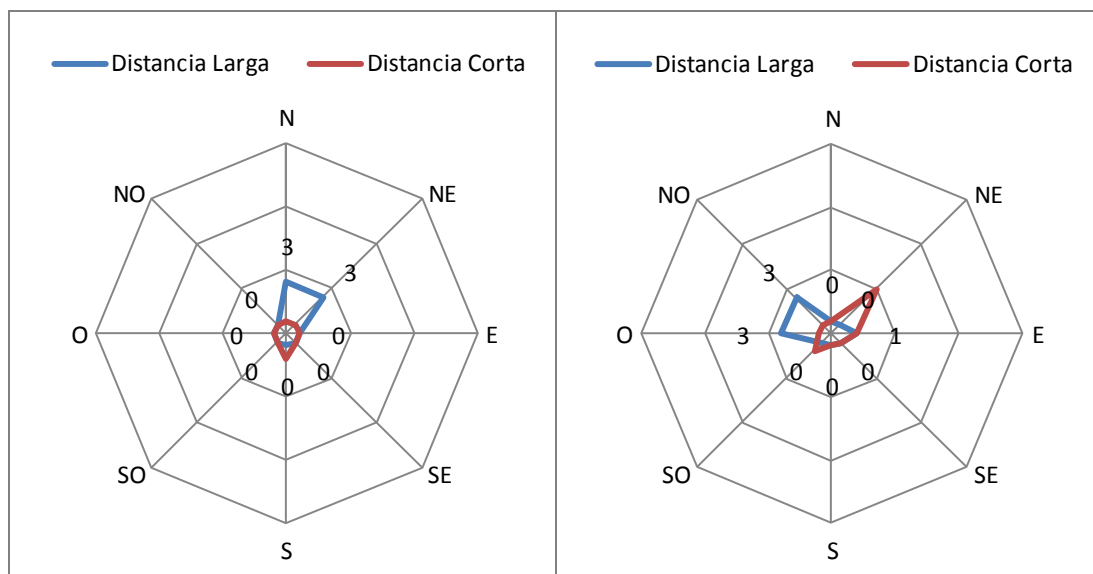


Figura 46. Cantidad media de conexiones fallidas por distancias.

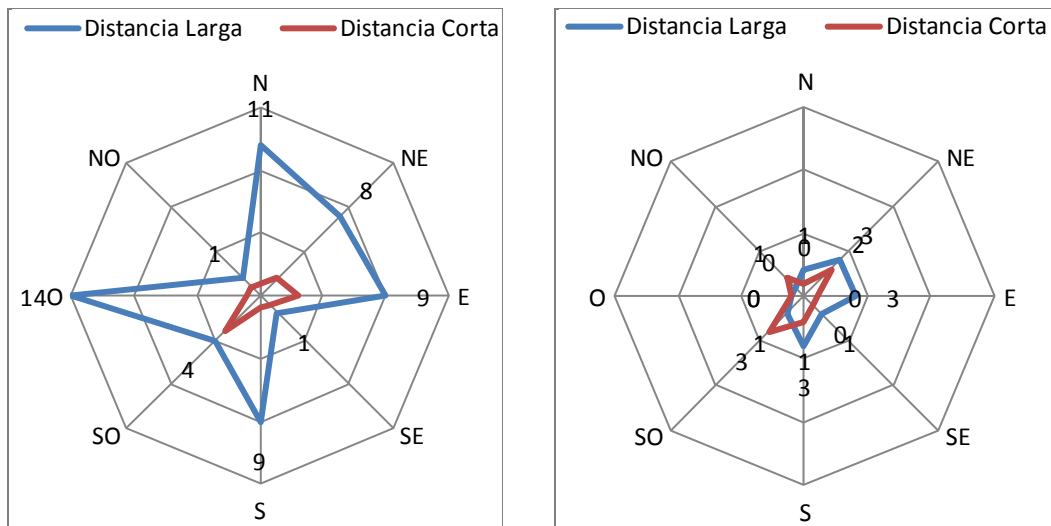
En la Figura 47 se puede observar para las técnicas asistidas (proximidad, dirección), el valor medio de la cantidad de conexiones fallidas por distancia y modo.

Comparando las cuatro orientaciones con menores fallos de conexión, se observa que para la distancia larga y el modo tangible, las orientaciones con menor número de fallos de conexión es sureste y noroeste, mientras que en el modo dedo en el norte, oeste y noroeste es donde se contabilizaron el menor número de conexiones fallidas. Para una distancia corta y modo tangible la orientación con menor número de fallos de conexión entre nodos es el noroeste, norte y sureste, mientras que para el modo dedo es el norte y el oeste.

Es importante remarcar que para la técnica asistida dirección en modo dedo, siempre es mayor el número de fallos de conexión, en todas las orientaciones en la distancia corta. Sin embargo en la técnica de proximidad ocurre lo contrario, siempre es mayor el número de fallos de conexión, en todas las orientaciones en la distancia larga.

Cantidad promedio de conexiones fallidas por distancias entre nodos	
Técnicas asistidas	
Tangible	Dedo

Proximidad



Dirección

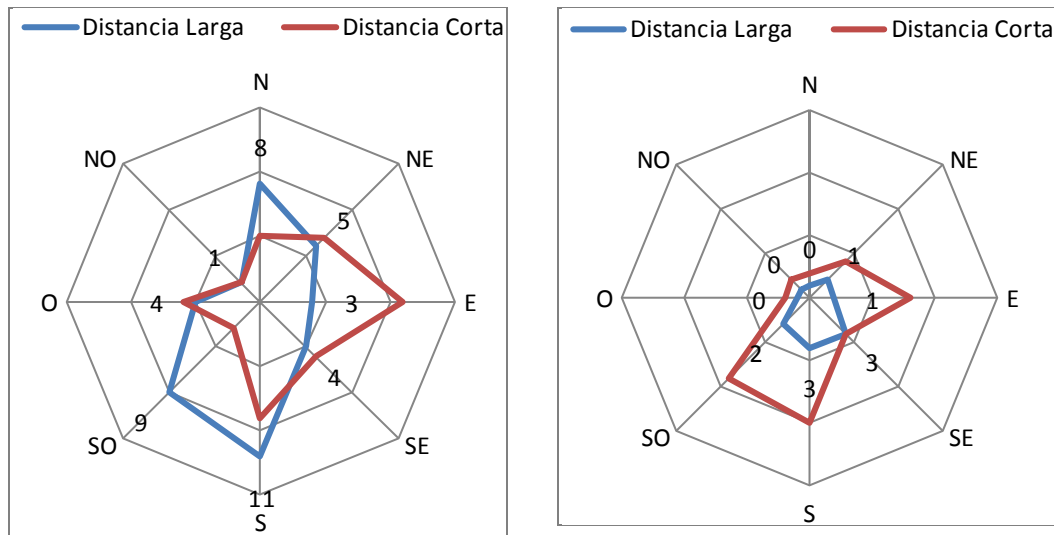


Figura 47. Cantidad promedio de intentos fallidos por distancias de conexión entre nodos .

3.1.6.3 Precisión en técnicas no-asistidas

Con la finalidad de saber cuál de las técnicas y modos son más precisas, hemos decidido medir el error en términos de distancia entre la posición ideal del nodo destino B y la posición en que realmente el usuario levantó el dedo/tangible para consolidar una conexión. Dicho de otro modo, se pretende

ESTUDIO EMPÍRICO

ver cuán precisos los usuarios fueron en adquirir el nodo destino, pese a que no fue un aspecto en el que se le informara el usuario durante la sesión experimental.

La Figura 48 muestra error para las técnicas no asistidas, que son las que realmente requieren alcanzar y adquirir el nodo destino. La Figura 49 muestra el error, pero también distinguiendo por la distancia del nodo destino. Este error mide la distancia en píxeles entre el punto de levantamiento del contacto y la posición del nodo destino. El hallazgo más importante observable es que el tangible está proporcionando mayor precisión (i.e. menor error). Esto significa que el usuario tendió a levantar el tangible cuando estaba más cercano al centro real del nodo que cuando se hacía lo propio con el dedo. Esto es por tanto un aspecto a tener en cuenta en el diseño de técnicas de conexión de elementos. En particular, si se requiere una mayor precisión, manteniendo objetos de tamaños similares a los del dedo/tangible utilizado, el uso del tangible estaría mejor indicado. Por supuesto si se consiguiera mitigar de alguna forma la problemática de la pérdida de seguimiento de tangibles cuando se interacciona con rapidez.

Error por modo de las técnicas asistidas.

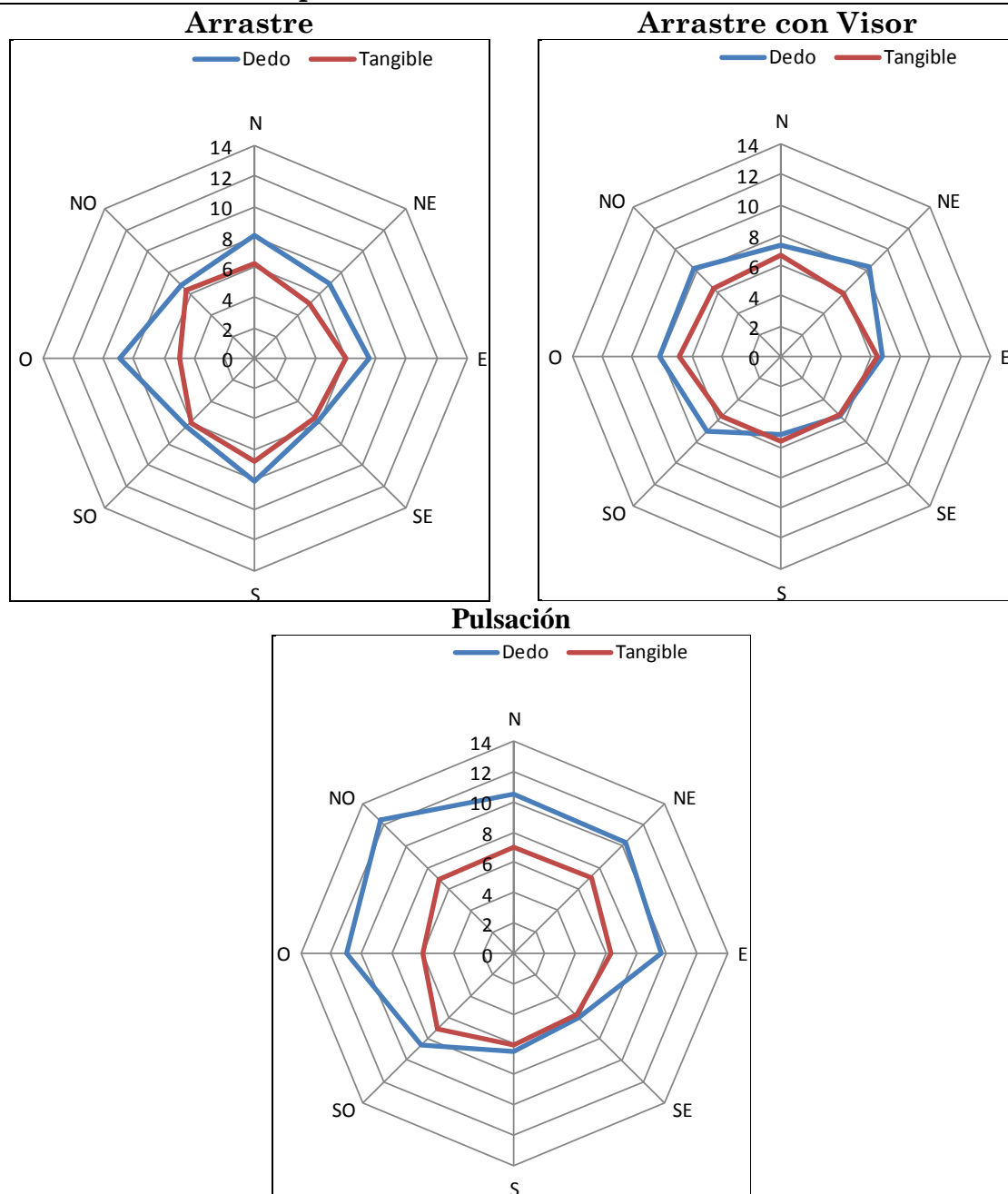
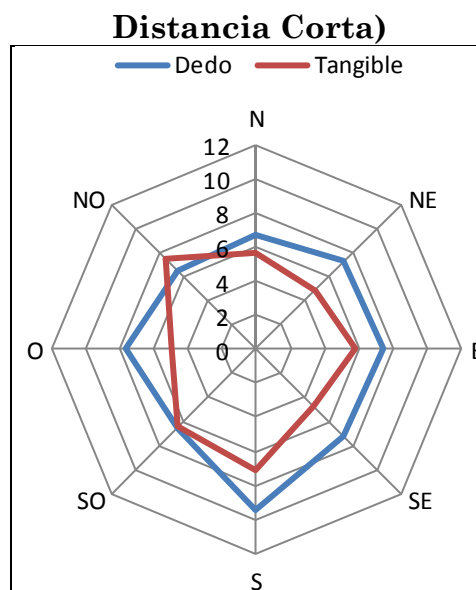
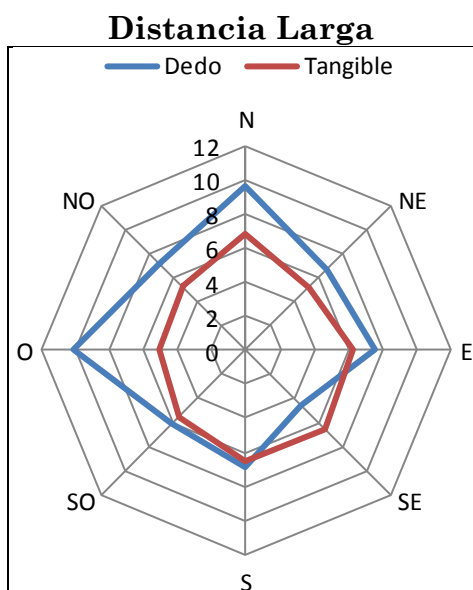
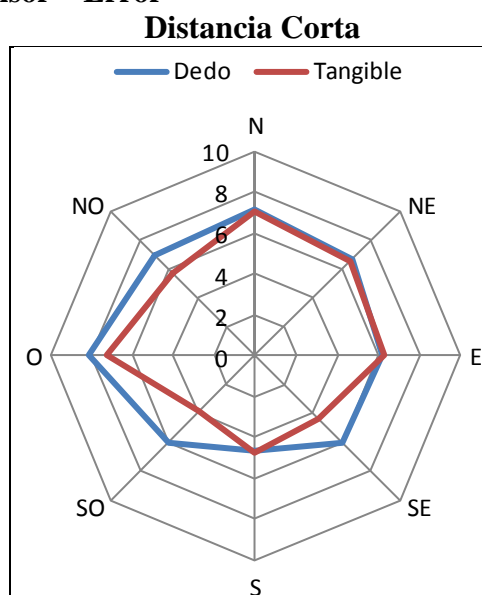
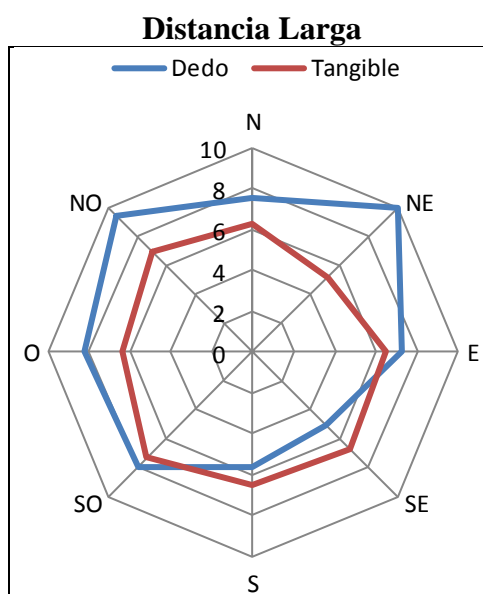


Figura 48. Error por modo de las técnicas asistidas.

Arrastre - Error



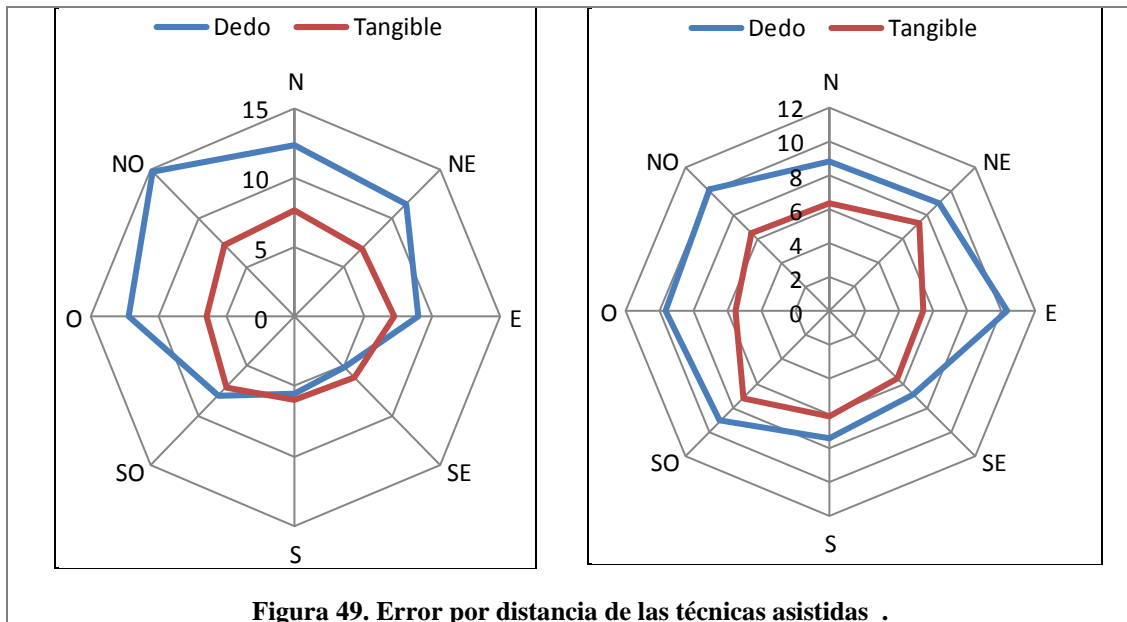
Arrastre con Visor - Error



Pulsación - Error

Distancia Larga

Distancia Corta



3.1.6.4 Ganancia en técnicas asistidas

Como hemos visto las técnicas asistidas sugieren al participante, al momento de conectar con el nodo destino, el nodo más cercano dependiendo de la implementación de cada una de ellas. Esta puede ser una posible ventaja para el participante, pues no tendrá que arrastrar desde el nodo inicial hasta el final. Con el fin de investigar esta característica de las técnicas asistidas (dirección, proximidad) hemos incluido una medida llamada “ganancia”. Es definida como la distancia desde el punto en que el usuario deja de construir la arista (punto de contacto de alzado) hasta el nodo destino b (ver Figura 50).

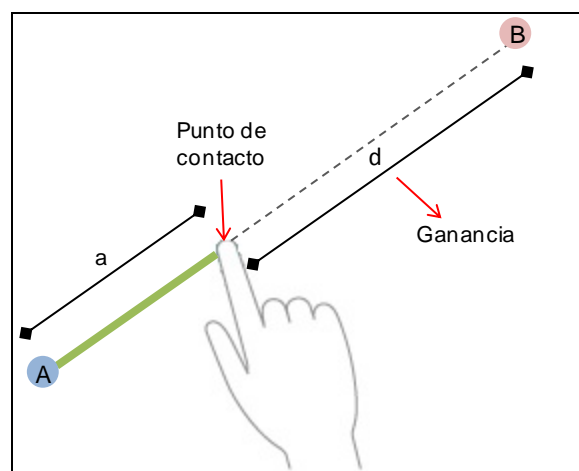


Figura 50. Definición intuitiva de ganancia para las técnicas asistidas.

La Figura 51 muestra la ganancia para las técnicas asistidas por modo, que son las únicas que tienen esta característica, esto le permite al participante

ESTUDIO EMPÍRICO

un menor esfuerzo al adquirir el nodo destino por sugerencia de la propia técnica. Esta ganancia mide la distancia en píxeles entre el punto de levantamiento del contacto y la posición del nodo destino. El hallazgo más importante observable es que el tangible está proporcionando mayor ganancia. Esto significa que el tangible tendió a ser más preciso en el momento del levantamiento del contacto. Esto es por tanto un aspecto a tener en cuenta en el diseño de técnicas de conexión de elementos. En particular, si se requiere una mayor precisión, manteniendo objetos de tamaños similares a los del dedo/tangible utilizado, el uso del tangible estaría mejor indicado.

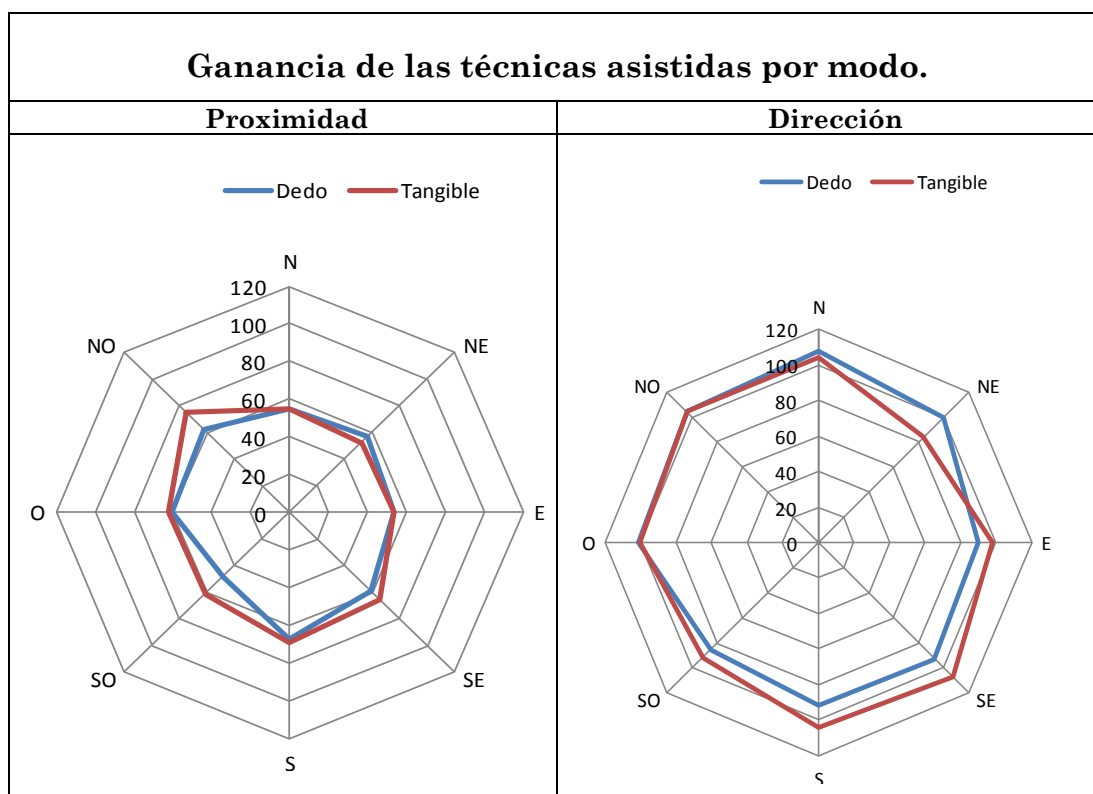


Figura 51. Ganancia de las técnicas asistidas por modo.

La Figura 52 muestra la ganancia a un mayor nivel de detalle, pero también distinguiendo por el obstáculo entre el nodo origen y del nodo destino. Un aspecto relevante que se puede observar es que el tangible con el obstáculo cerca obtiene mayor ganancia que cuando el obstáculo está lejos. Para el dedo la situación es similar, se obtiene mayor ganancia con el obstáculo cerca que cuando el obstáculo está lejos. Esto significa que el usuario al momento del levantamiento del contacto, cuando tiene el obstáculo cercano, recorre menor distancia, que si tuviera el obstáculo lejano, como era de esperar.

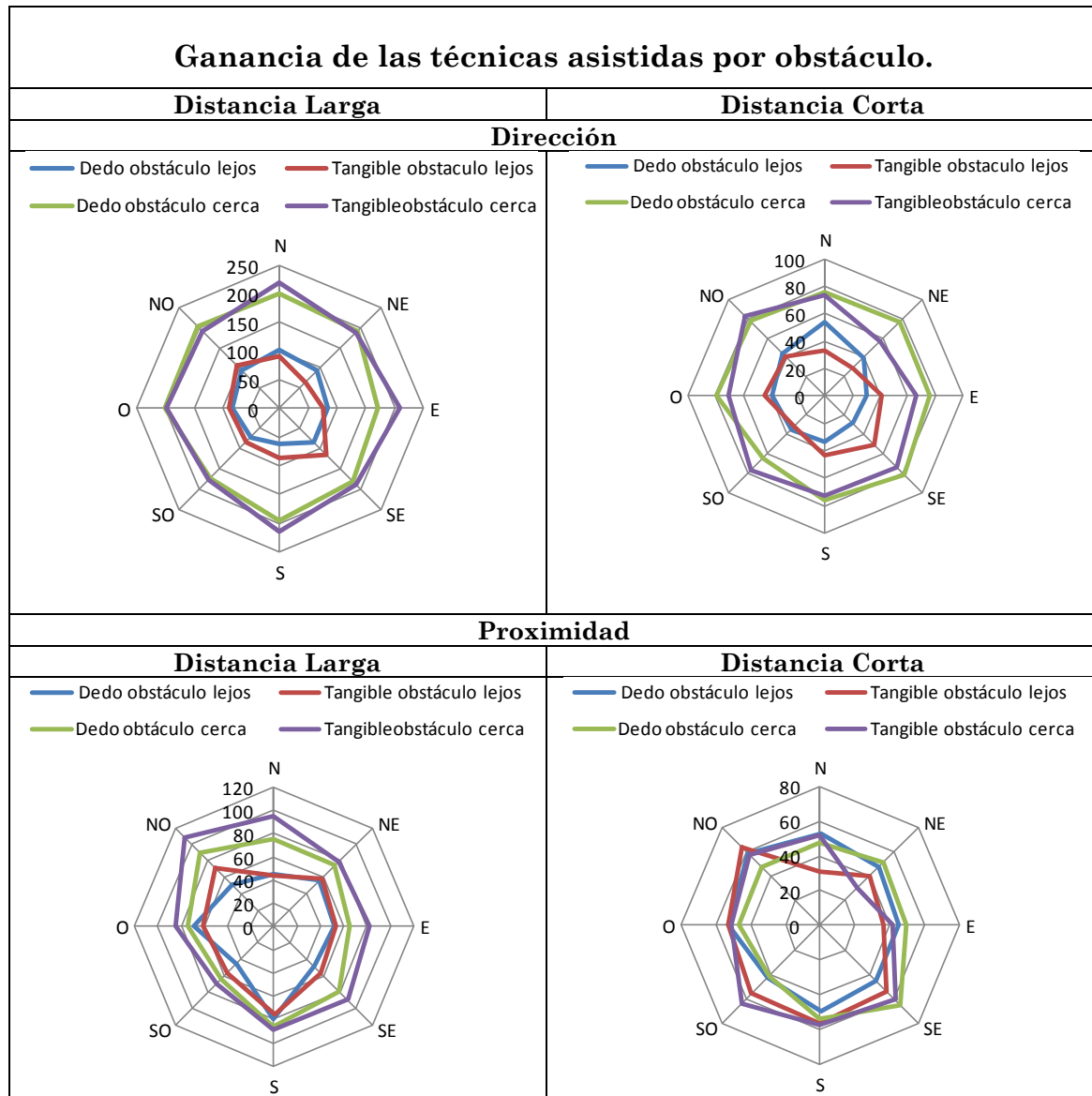


Figura 52. Ganancia de las técnicas asistidas por obstáculo.

En la Figura 53 y Figura 54 se muestran los centroides, donde se levantan las líneas para conectar los nodos en distancias largas y cortas respectivamente. Se observa que la técnica de dirección recorre menos distancia que la proximidad, en distancias largas y en distancias cortas también, como era de esperar, dado que el punto de levantamiento del contacto en la técnica asistida de dirección es más lejano del nodo destino, mientras que el punto de levantamiento del contacto con la técnica de proximidad es más cercano al nodo destino. Es por eso que con la técnica de proximidad se recorre una mayor distancia en pixeles.

ESTUDIO EMPÍRICO

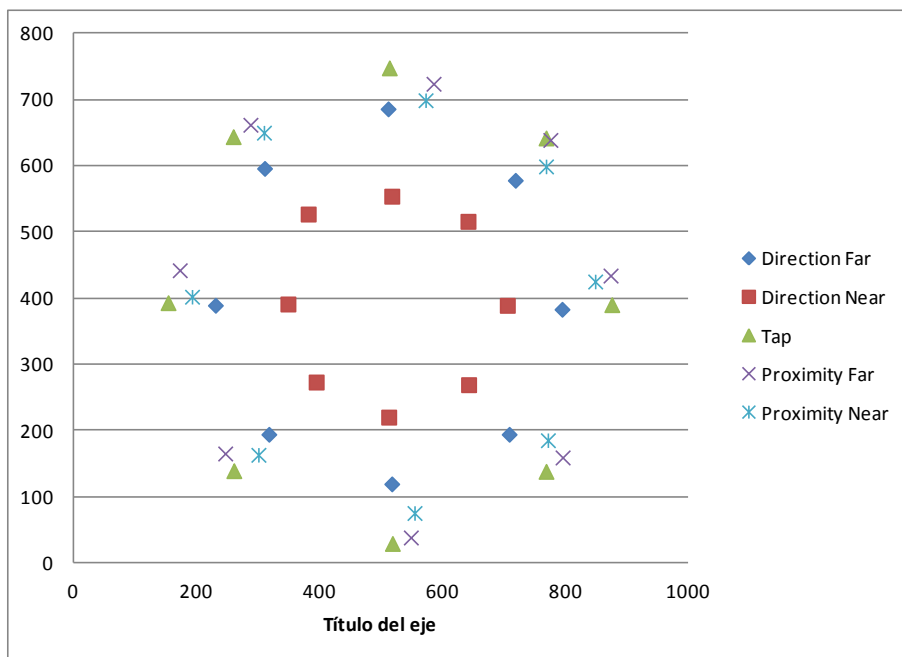


Figura 53. Centroides donde se levantan las líneas para conectar en distancias largas.

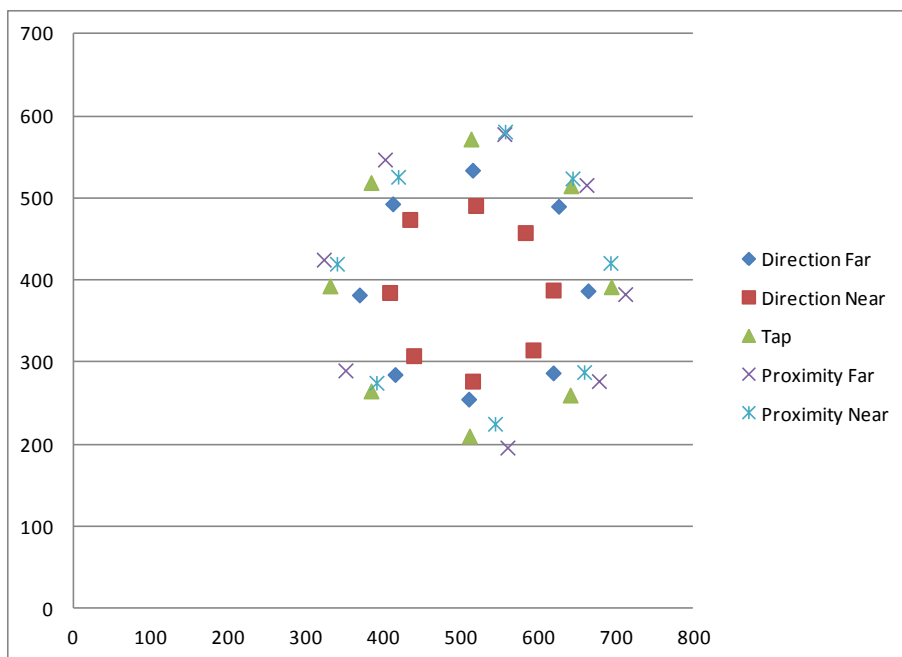


Figura 54. Centroides donde se levantan las líneas para conectar en distancias cortas.

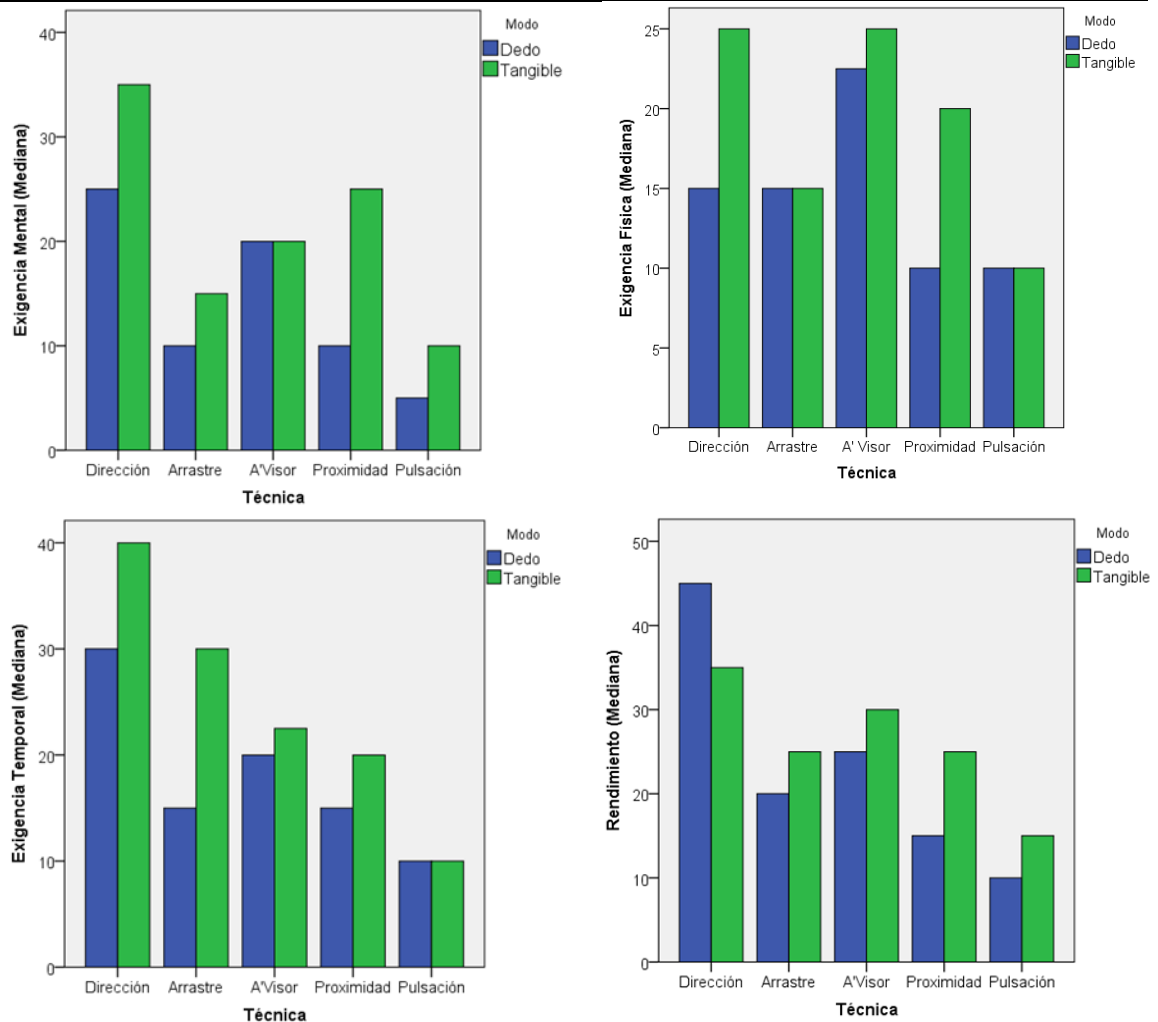
3.1.6.5 Cuestionarios

Al finalizar cada tarea el participante respondió a un cuestionario basado en el NASA Task Load Index instrument [Har2006]. Este cuestionario permitió a los participantes valorar y evaluar una serie de conceptos, relacionados con la

tarea terminada. Estos conceptos son: Exigencia Mental, Exigencia Física, Exigencia Temporal, Rendimiento, Esfuerzo y Frustración. Esta data fue procesada y analizada.

En la Figura 55 se observa, en general que las técnicas que usan como dispositivo de entrada el modo tangible, han requerido de mayor exigencia mental, más exigencia física, por parte del usuario, que las técnicas manipuladas por el dedo. De las técnicas asistidas se visualiza además de el tema del tangible, que genero mayor frustración, más exigencia temporal y mejor rendimiento por parte del participante en la técnica de dirección, respecto a la de proximidad.

Resumen del cuestionario TLX



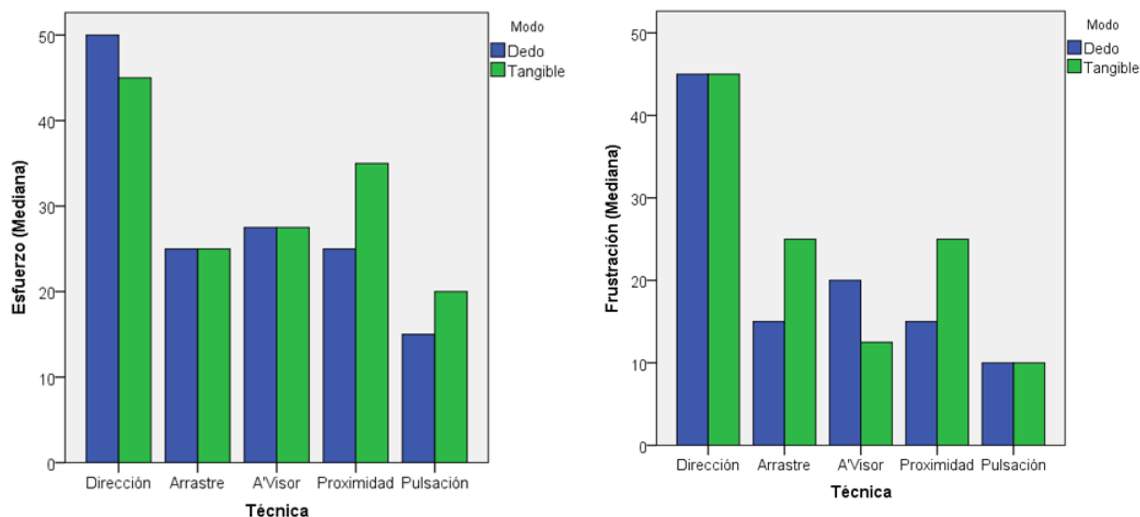


Figura 55. Cuestionarios basados en el NASA Task Load Index instrument

3.2 Discusión

Existen dos aspectos primordiales que se han tenido en cuenta en la tarea experimental, el primero construir una arista entre dos nodos en el menor tiempo posible, y el segundo hacerlo tratando de no cometer fallos de conexión al momento de unir dos nodos.

Se estudió el tiempo y el esfuerzo que los usuarios requieren para establecer conexiones, también se estudió cuán dificultosa es cada técnica, asumiendo que será menos problemática cuando el usuario cometa menos errores en el uso de las técnicas.

Otro aspecto importante a estudiar es el efecto del dispositivo de entrada dedo o tangible, tal que se pueda conocer la influencia de estos, en el rendimiento de cada técnica. Se ha analizado también el beneficio que se obtiene cuando se usan las técnicas asistidas.

Este capítulo ha presentado el análisis y evaluación empírica a las que se han sometido un conjunto de técnicas de interacción para la conexión de nodos en una mesa interactiva, involucrando a usuarios sin notable experiencia con esta tecnología.

En términos generales, la técnica que presenta mejores resultados en términos de tiempo es la técnica de Pulsación. Esto puede ser debido a que esta técnica es mucho más intuitiva y de fácil manejo para el participante. En las técnicas asistidas de dirección y proximidad se observó, que el número de fallos de conexión en modo tangible era superior al modo dedo (aproximadamente una proporción de 3,5).

En la técnica arrastre en modo tangible, el tiempo mayor se obtuvo cuando el nodo destino estaba ubicado en el S, mientras que para el modo dedo fue lo contrario. Quizás sea por el hecho de que el participante al hacer este desplazamiento al nodo destino en la orientación sur con el dedo, es un movimiento habitual o cotidiano, que ha sido entrenado inconscientemente.

Por lo general, para las técnicas no asistidas podríamos decir que las áreas que requieren menor tiempo para conectar nodos son las relativas al S, SO, O. Puede influir el hecho de requerir movimiento más cercanos al cuerpo y que no presenta tanta oclusión en esa región. Las regiones del norte se observaron que podían llegar a ser problemáticas porque el usuario tiene que hacer un esfuerzo mayor con el brazo.

Los mejores tiempos al evaluar el valor promedio de conexiones fallidas en modo tangible, se ubicaron en tercer cuadrante (SO,O) y para el modo dedo, se localizaron en la orientación S y SO.

Se observó que con el tangible se obtuvo mayor precisión (menor error). Es un aspecto a tomar en cuenta en el diseño de técnicas de conexión de elementos, si se requiere una mayor precisión, con objetos de tamaños similares a los del tangible.

El tangible proporcionó una mayor ganancia en las técnicas asistidas, eso implica que el usuario obtuvo un menor esfuerzo para realizar la tarea.

Se observó que la técnica de proximidad recorre una mayor distancia que la técnica de dirección, en distancias largas y en distancias cortas, dado que el punto de levantamiento del contacto en la técnica de dirección es más lejano del nodo destino, mientras que el del contacto con la técnica de proximidad es más cercano al nodo destino.

CONCLUSIONES Y TRABAJOS FUTUROS

4.1 Conclusiones

Los diagramas de conexión de nodos (grafos), son usados en muchos ámbitos de la vida real para describir y explorar relaciones entre datos. En este trabajo se han analizado, implementado y evaluado de forma empírica un conjunto de técnicas de interacción, en un marco de trabajo genérico, que permite representar grafos visualmente de forma muy básica sobre una mesa interactiva, involucrando a usuarios sin notable experiencia con esta tecnología.

En este trabajo se han evaluado las técnicas clasificándolas por técnicas no asistidas (Arrastre, Pulsación, Arrastre con Visor) y asistidas (Dirección, Proximidad).

Las técnicas de Arrastre y Pulsación, ya existen en esta tecnología. En el caso de la técnica de Nodo magnético, es una adaptación que sugiere al usuario, el nodo final para realizar la conexión. En términos generales, la técnica que presenta mejores resultados en términos de tiempo es la técnica de Pulsación. Esto puede ser debido a que esta técnica es mucho más intuitiva y de fácil manejo para el participante.

En general, no se han obtenido los resultados esperados, respecto de la técnica manual, arrastre con visor. Ya que ha sido una de las que más tiempo ha requerido en los dos modos de interacción (dedo, tangible). Una posible causa es porque el participante no está habituado a esta tecnología.

Una aportación importante a este trabajo es el diseño de técnicas asistidas, tal es el caso de Dirección y Proximidad, las cuáles son totalmente nuevas. En esta técnica se sugiere al usuario la conexión al nodo destino, sin que él tenga que hacer el desplazamiento completo. Puede decirse que son técnicas

novedosas, no desarrolladas hasta el momento y que persiguen dotar de cierta asistencia al proceso de establecimiento de aristas.

Existen dos aspectos fundamentales que se han tomado en cuenta en la tarea experimental, construir una arista entre dos nodos con el mejor rendimiento (menor tiempo posible) y hacerlo tratando de no cometer fallos de conexión al momento de unir dos nodos.

También se analizó la influencia del dispositivo de entrada (dedo, tangible), en el rendimiento de cada técnica. En las técnicas asistidas de dirección y proximidad se observó, que el número de fallos de conexión en modo tangible era superior al modo dedo.

Se ha explorado el efecto de la orientación sobre el tiempo requerido de conexión para las diversas técnicas. En la técnica arrastre en modo tangible, el tiempo mayor se obtuvo cuando el nodo destino estaba ubicado en la orientación Sur (la más cercana al usuario), mientras que para el modo dedo fue lo contrario. Quizás sea por el hecho de que el participante al hacer este desplazamiento al nodo destino en la orientación sur con el dedo, es un movimiento habitual o cotidiano, que ha sido entrenado inconscientemente.

En general, para las técnicas no asistidas, el área que se ve más favorecida (menor tiempo) en la conexión de nodos, es la localizada al Sur, Suroeste, Oeste. Puede influir el hecho de requerir movimiento más cercano al cuerpo y que no presenta tanta oclusión en esa región. Las regiones del N se observaron que podían llegar a ser problemáticas porque el usuario tiene que hacer un esfuerzo mayor con el brazo.

Respecto a la orientación por modo, los mejores tiempos al evaluar el valor promedio de conexiones fallidas en modo tangible, se ubicaron en tercer cuadrante (Suroeste, Oeste) y para el modo dedo, se localizaron en la orientación Sur y Suroeste.

Con el propósito de evaluar la precisión de los usuarios en la selección del nodo destino de la conexión, se midió la distancia (error) entre el punto de levantamiento del contacto y la posición del nodo destino. El hallazgo más importante observable en este sentido es que el tangible está proporcionando mayor precisión (i.e. menor error). Esto significa que el usuario tendió a levantar el tangible cuando estaba más cercano al centro real del nodo que cuando se hacía lo propio con el dedo. Esto es por tanto un aspecto a tener en cuenta en el diseño de técnicas de conexión de elementos. En particular, en el supuesto de que se requiera una aplicación con mayor precisión, manteniendo objetos de tamaños similares a los del dedo/tangible utilizado, el uso del tangible estaría mejor indicado en ese caso. Por supuesto si se consiguiera mitigar de alguna forma la problemática de la pérdida de seguimiento de tangibles cuando se interacciona con rapidez.

Con el fin de analizar el esfuerzo requerido con las técnicas asistidas, se ha definido una medida llamada ganancia (distancia desde el punto en que el usuario deja de construir la arista en el punto de contacto de alzado hasta el nodo destino). Esto significa que a mayor ganancia, menor distancia recorrida, por lo tanto menor esfuerzo. Lo que se observó es que en el modo tangible se obtuvo una mayor ganancia.

Cuando incorporamos un obstáculo entre el nodo origen y del nodo destino, se observó que con el modo de interacción tangible, teniendo el obstáculo cerca, se obtiene mayor ganancia que cuando el obstáculo está lejos. Esta misma situación de causa y efecto se repite cuando la interacción se realiza con el dedo para las dos técnicas asistidas.

Se hizo también un análisis comparativo entre las técnicas asistidas y la técnica no asistida de proximidad. Cuando los nodos se encuentran a dos diferentes distancias (lejos y cerca), se observó que la técnica de dirección recorre menos distancia (mayor ganancia) que la proximidad, en distancias largas y en distancias cortas

Además cabe destacar que con la mayoría de las técnicas, salvo la de Pulsación se pueden llevar a cabo tareas colaborativas, que es una de las aportaciones más significativas de la tecnología multitáctil.

4.2 Trabajos futuros

Planeamos llevar a cabo la incorporación de las técnicas de interacción que hemos implementado en otras aplicaciones, tal como en el caso de un editor de expresiones de flujos de datos [Pon2013]. Esto es factible ya que el marco es lo suficientemente genérico.

Los resultados obtenidos en este estudio, nos motivan a replicar este primer estudio experimental, incluyendo más sujetos y extendiendo las sesiones de entrenamiento. También se pueden incluir nuevas tareas, donde el obstáculo no esté alineado con los nodos origen y destino de la conexión, permitiendo la evaluación de la técnica de dirección, donde el layout no se ha establecido de forma automática.

Para grafos realmente grandes y extensos, implementar conceptos relacionados con la escala, implementar funciones de zoom, según sea el caso.

Poder fraccionar el grafo, de manera que se pueda hacer la implementación de ventanas, que permita visualizar el grafo por secciones y que el usuario de esta manera, pueda moverlo y manipularlo a su conveniencia.

BIBLIOGRAFÍA

Finalmente planeamos también diseñar y explorar nuevas técnicas asistidas que puedan ser bimanuales y evaluar su eficiencia y efectividad.

BIBLIOGRAFÍA

- [Api2005] Apitz, G., Guimbretière, F.: CrossY: a crossing-based drawing application. ACM .Transactions on Graphics 24(3) (July 2005) 930–930
- [Ben2001] Ben-Joseph, E., Ishii, H., Underkoffler, J., Piper, B., and Yeung, L. Urban Simulation and the Luminous Planning Table: Bridging the Gap between the Digital and the Tangible, Journal of Planning Education and Research, 21(2), 196-203, 2001.
- [Bat1998] G. Battista, P. Eades, R. Tamassia, and I.G. Tollis. Graph Drawing: Algorithms for the Visualization of Graphs. Prentice Hall PTR, 1998. Prentice Hall PTR, 1998.
- [Die2001] Dietz, P. and Leigh, D. DiamondTouch: a multi-user touch technology. In Proc. of UIST'01, ACM, pp. 219-226, 2001.
- [Fri2010] Diagram Editing on Interactive Displays Using Multi-Touch and Pen Gestures Mathias Frisch, Jens Heydekorn, Raimund Dachsel
User Interface & Software Engineering Group Otto-von-Guericke-University Magdeburg, 39106 Magdeburg, Germany {mfrisch, jheyde, [dachsel](mailto:dachsel@isg.cs.uni-magdeburg.de)}@isg.cs.uni-magdeburg.de
- [Gar2013] Fernando Garcia-Sanjuan, Javier Jaen, Alejandro Catala, Evaluating heuristics for tabletop user segmentation based on simultaneous interaction, Expert Systems with Applications, Volume 40, Issue 14, 15 October 2013, Pages 5578-5587, ISSN 0957-4174, 10.1016/j.eswa.2013.04.011. (<http://www.sciencedirect.com/science/article/pii/S0957417413002480>)

- [Han2005] Han, J. Y. 2005. Low-Cost Multi-Touch Sensing through Frustrated Total Internal Reflection. In Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology
- [Han2004] Hancock, M.S., Booth, K.S.: Improving menu placement strategies for pen input. In: Proc. GI. (2004) 221–230
- [Har2006] Hart, S. G. (2006). NASA-Task Load Index (NASA-TLX); 20 Years Later. Proceedings of the Human Factors and Ergonomics Society 50th Annual Meeting, 904-908. Santa Monica: HFES.
- [Her2000] Ivan Herman, Guy Melançon, and M. Scott Marshall. 2000. Graph Visualization and Navigation in Information Visualization: A Survey. IEEE Transactions on Visualization and Computer Graphics 6, 1 (January 2000), 24-43. DOI=10.1109/2945.841119 <http://dx.doi.org/10.1109/2945.841119>
- [Hin2002] Ken Hinckley. 2002. Input technologies and techniques. In *The human-computer interaction handbook*, Julie A. Jacko and Andrew Sears (Eds.). L. Erlbaum Associates Inc., Hillsdale, NJ, USA 151-168.
- [Hor2012] Michael Horn, Zeina Atrash Leong, Florian Block, Judy Diamond, E. Margaret Evans, Brenda Phillips, and Chia Shen. 2012. Of BATs and APes: an interactive tabletop game for natural history museums. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems (CHI '12)*. ACM, New York, NY, USA, 2059-2068. DOI=10.1145/2208276.2208355
<http://doi.acm.org/10.1145/2208276.2208355>
- [Mos2009] Tomer Moscovich, Fanny Chevalier, Nathalie Henry, Emmanuel Pietriga, and Jean-Daniel Fekete. 2009. Topology-aware navigation in large networks. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09). ACM, New York, NY, USA, 2319-2328. DOI=10.1145/1518701.1519056
<http://doi.acm.org/10.1145/1518701.1519056>

- [Ink2006] Inkpen, K., Dearman, D., Argue, R., Comeau, M., Fu, C.L., Kolli, S., Moses, J., Pilon, N., Wallace, J.R.: Left-handed scrolling for pen-based devices. *International Journal of Human Computer Interaction* 21(1) (2006) 91–108
- [Microsoft-Surface] Microsoft Corporation. Microsoft Surface. <http://www.microsoft.com/surface>.
- [Pau2006] GUESS: A Language and Interface for Graph Exploration Eytan Adar. University of Washington, Computer Science and Engineering .101 Paul G. Allen Center, Box 352350, Seattle, WA 98195 .eadar@u.washington.edu
- [Pon2013] Patricia Pons, Alejandro Catalá, Javier Jaén. TanRule: A Rule Editor for Behavior Specification on Tabletops. *Extended Abstracts of the ACM Tangible, Embedded and Embodied Interaction (TEI 2013)*, 2013.
- [Pur2004] Purchase, H.C. (2004) Evaluating Graph Drawing Aesthetics: defining and exploring a new empirical research area. In: DiMarco, J. (ed.) *Computer Graphics and Multimedia: Applications, Problems and Solutions*. Idea Group Publishing, pp. 145-178.
- [Ram2003] Ramos, G., Balakrishnan, R.: Fluid interaction techniques for the control and annotation of digital video. In: *Proc. ACM UIST*. (2003) 105–114
- [Sch2008] Schöning, J., Brandl, P., Daiber, F., Echtler, F., Hilliges, O., Hook, J., Löchtfeld, M., Motamedi, N., Muller, L., Olivier, P., Roth, T. and von Zadow, U: *Multi-Touch Surfaces: A Technical Guide*. Technical Report TUM-I0833: Technical Reports of the Technical University of Munich, (2008).
- [Sch2010] Sebastian Schmidt, Miguel A. Nacenta, Raimund Dachsel, and Sheelagh Carpendale. 2010. A set of multi-touch graph interaction techniques. In *ACM International Conference on Interactive Tabletops and Surfaces (ITS '10)*. ACM, New York, NY, USA, 113-116. DOI=10.1145/1936652.1936673 <http://doi.acm.org/10.1145/1936652.1936673>

- [Ull2001] Ullmer, B. and Ishii, H. Emerging Frameworks for Tangible User Interfaces, In *Human-Computer Interaction in the New Millenium*, Ed. John M. Carroll, Addison-Wesley, pp.579-601, 2001.
- [Vog2007] Vogel, D. & Baudisch, P. Shift: A Technique for Operating Pen-Based Interfaces Using Touch. *Proc. CHI'07*, pp. 657-666.
- [Vog2012]. Daniel Vogel and Géry Casiez. 2012. Hand occlusion on a multi-touch tabletop. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems (CHI '12)*. ACM, New York, NY, USA, 2307-2316. DOI=10.1145/2208276.2208390
<http://doi.acm.org/10.1145/2208276.2208390>
- [Won2003] Nelson Wong, Sheelagh Carpendale, and Saul Greenberg. 2003. EdgeLens: an interactive method for managing edge congestion in graphs. In *Proceedings of the Ninth annual IEEE conference on Information visualization (INFOVIS'03)*, Tamara Munzner and Stephen North (Eds.). IEEE Computer Society, Washington, DC, USA, 51-58.
- [Won2007] Nelson Wong ; Sheelagh Carpendale; Supporting interactive graph exploration using edge plucking. *Proc. SPIE 6495, Visualization and Data Analysis 2007*, 649508 (January 29, 2007); doi:10.1117/12.704612.
- [Ze2006] Zeleznik, R.C., Miller, T.: Fluid inking: augmenting the medium of free-form inking with gestures. In: *Proc. GI. (2006)* 155–162
- [Zha2012]Zhang, H., Yang, X.-D., Ens, B., Liang, H.-N., Boulanger, P., & Irani, P. (2012). See me, see you: a lightweight method for discriminating user touches on tabletop displays. In *Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 2327–2336). New York, NY, USA: ACM.