



Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

AW-1 Seminar-Ausarbeitung

Jörn Siedentopp

motion-based control of mobile devices
dynamic peephole interface for mobile devices

Jörn Siedentopp

Matrikelnummer 1859774 – www.siedentopp.de – joern@siedentopp.de

motion-based control of mobile devices
dynamic peephole interface for mobile devices

Seminar-Ausarbeitung eingereicht im Rahmen der Veranstaltung AW-1
im Studiengang Master of Science Informatik
am Department Informatik
der Fakultät Technik und Informatik
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Professor: Prof. Dr. Olaf Zukunft
Gutachter: Prof. Dr. rer. nat. Kai von Luck
& Prof. Dr. rer. nat. Gunter Klemke

Abgegeben am 28. Februar 2010

Inhaltsverzeichnis

1. Einleitung & Erkenntnisinteresse	1
2. Mobile Usability	3
3. Peephole Interface.....	4
3.1 Aktuelle dynamische Peephole Implementierungen	6
3.1.1 Peephole With No Strings Attaches	7
3.1.2 ISeeU	9
3.2 Potenzial & Grenzen von dynamischen Peepholes.....	10
3.3 Ausblick	11
3.3.1 Speed-dependent Automatic Zooming.....	12
4. Resümee.....	14
Literaturverzeichnis.....	15

1. Einleitung & Erkenntnisinteresse

Handheld-Geräte sind heutzutage aus dem Alltag nur noch schwer wegzudenken. Das Smartphone wird mittlerweile zum Surfen im Internet, als TV-Zeitung, zum Abrufen von E-Mails, als Wecker, Taschenrechner, Musikplayer oder Finanzplaner, als Einkaufsliste, Wetterdienst, Routenplaner, Spielekonsole und für vieles mehr genutzt. Aus diesem Anlass veranstaltete ein Hamburger Radiosender Anfang Februar 2010 eine Woche, in der einer der Moderatoren sich für eine Woche von seinem Smartphone trennte. Den Alltag ohne Smartphone zu gestalten, war für den Radiosender so undenkbar und bemerkenswert, dass der Moderator täglich Bericht von seinem Smartphone-Entzug erstattete [vgl. 1].

Wissenschaftlich wird der Trend hin zum Smartphone als digitaler mobiler Alltagsunterstützer vom Marktforschungsunternehmen IDC belegt. So sind 15,4 % aller weltweit verkauften Handys laut der IDC-Studie vom 04.02.2010 Smartphones [2]. Diese Studie vergleicht die Verkaufszahlen und die Marktaufteilung der Smartphone Hersteller. Eindeutiger Gewinner der Studie ist die Firma Apple. Mit nur einem Gerät auf dem Markt, dem Apple iPhone, und weltweit 25,1 Millionen verkaufter Geräte, platzierte sich Apple auf Platz drei der weltweiten erfolgreichsten Smartphone Hersteller [2]. Alle Hersteller auf den übrigen Plätzen stellen unzählige Smartphone-Geräte her und trotzdem verzeichnet Apple deutliche Gewinne. Das iPhone kann jedoch nicht mit technischer Ausstattung aufwarten, sodass die Software für den Erfolg zu sorgen scheint. Software spiegelt sich für Nutzer im Interface wieder; hier ist also die Bedienbarkeit (Usability) das wichtigste Kriterium. Wie Webseiten behaupten auch viele Nutzer: "Bedienung und Design des iPhones sind Weltklasse" [3]. Hier wird, wie ich schon in meiner Bachelorarbeit feststellte [vgl. 4], sichtbar, dass der Erfolg eines Produktes stark mit der Usability verknüpft ist.

Das Kernkriterium für ein gut nutzbares Interface ist nach Donald A. Norman [vgl. 5] seine "Natürlichkeit". Ihm zufolge muss ein Interface ohne zusätzlich erlerntes Wissen nutzbar sein. Eine potentiell guter Ansatz für natürliche Interaktion mit mobilen

Handhelds ist meiner Meinung nach ein dynamisches Peephole-Interface. Die Idee hierbei ist, die Beweglichkeit von mobilen Geräten als intuitiven Steuerungsinput für die Benutzeroberfläche zu nutzen und die kleinen Geräte als bewegliches Fenster in einen großen virtuellen Arbeitsraum zu begreifen [vgl. 6]

Der Nutzer muss bei dieser Idee kein neues Wissen erlernen, anstelle dessen sind große technische Herausforderung bei der Entwicklung zu bewältigen. Um die Bewegung des Endgerätes zu ermitteln, gibt es unterschiedliches Möglichkeiten. Jede der Methode erfordert allerdings viel Rechenleistung. Im Widerspruch hierzu steht, dass die Geräte durch ihre Mobilität auf einen Akku angewiesen sind, der nur eine begrenzte Leistung bietet.

Weiter werden die Interaktionsmöglichkeiten von Smartphones durch ihre kleine Baugröße und limitierte Eingabemöglichkeiten beschränkt. Es kommen kleine Displays und Tasten (egal ob digital eingeblendet oder physisch angebracht) zum Einsatz. Kleine Displays sind schlecht lesbar oder können nur wenige Informationen darstellen und kleine Tasten sind schwer bedienbar. Eine gute Usability ist bei kleinen Geräten deshalb nur schwer erreichbar. Andererseits sind die Geräte mobil, können also uneingeschränkt bewegt werden. Dies eröffnet Möglichkeiten der Interaktion, die Desktopgeräte nicht bieten. Hier setzen dynamische Peepholes an.

Im Folgenden möchte ich diese Idee vorstellen. Weiter werden zwei unterschiedliche Implementierungen dargestellt und die Alltagstauglichkeit dieser Interfaces-Art kritisch beleuchtet. Wichtig ist mir in dieser Ausarbeitung, den Fokus auf den potenziellen Nutzer, also auf die Usability, zu legen, denn das beste Interface ist nicht nötig, wenn es nicht genutzt wird.

2. Mobile Usability

In Fachkreisen wird der englische Begriff *Usability* anstelle des deutschen Begriffs *Gebrauchstauglichkeit* verwendet [vgl. 7]. Obwohl die direkte Übersetzung (*Be-)Nutzbarkeit* lautet, wird diese nicht verwendet, damit Verwechslungen mit dem englischen Begriff *user friendly* ausgeschlossen werden [7]. Daher soll im Folgenden ausschließlich der eingedeutschte Begriff Usability verwendet werden. Im Teil 11 der ISO 9241 wird Usability klar als



“Das Ausmaß, in dem ein Produkt durch bestimmte Benutzer in einem bestimmten Nutzungskontext genutzt werden kann, um bestimmte Ziele **effektiv, effizient und zufriedenstellend** zu erreichen” [vgl. 8]

definiert. Wichtig für ein Produkt ist dies aus wirtschaftlichen Gründen, denn der Nutzer entscheidet letzten Endes anhand der Nutzungsqualität über den Erfolg oder Misserfolg eines Produktes. Eine Bedienung, die die Kriterien der Usability erfüllt, wird als benutzerfreundlich angesehen [vgl. 7].

Probleme der Usability werden mit Hilfe unterschiedlicher Usability-Untersuchungsmethoden aufgedeckt, um diese dann beheben zu können. Der wichtigste Punkt hierbei ist die direkte Kommunikation mit dem letztendlichen Nutzer.

Der Usability-Bereich für mobile Endgeräte ist noch nicht umfassend erforscht. Mobile Geräte stellen gerade hier eine große Herausforderung dar, denn die zunehmende Funktionsvielfalt steht der Usability entgegen. Murphy et al. entwickelte diesbezüglich einen Verlauf, bei dem an einem gewissen Punkt, dem “Usability Knee”, die Usability extrem abfällt [vgl. 9].

Detaillierte Informationen zu Mobile Usability bietet unter anderem die Ausarbeitung von Christian Wagner an der HAW Hamburg [10]. An dieser Stelle soll nicht weiter auf die Details der Tests eingegangen werden, da ein Bewusstsein über die Wichtigkeit einer guten Usability für diese Arbeit ausreichend ist.

3. Peephole Interface

Jede digitale Darstellungsfläche ist – ob nun nach einigen wenigen Zentimetern oder nach vielen Metern - begrenzt. Die technische Abmessung stellt die Grenzen für das softwareseitige Interface dar: Der Bildschirm kann somit als Guckloch (dt. für Peephole) auf die digitalen Informationen bezeichnet werden. Kun et. al. definieren:



“Peepholes are virtual windows to large workspaces.” [11]

Die Eigenschaft eines Peepholes (vergleichbar mit einem Türspion) ist, dass es nur eine begrenzte Größe hat und nur einen mehr oder weniger großen Bereich der tatsächlichen Information auf der anderen Seite abbilden kann. Will man Informationen betrachten, die momentan nicht sichtbar sind, müssen diese vor das Peephole geschoben werden, oder das Peephole muss vor die Informationen bewegt werden. Bei mobilen Endgeräten ist das Peephole im Verhältnis zu den Informationen extrem klein, das bedeutet, um alle Informationen zu betrachten, muss eine der Komponenten (Peephole oder Information) stark und oft bewegt werden.

Heute gebräuchliche Software setzt auf Scrollen der Information hinter dem Peephole, dies wird als statische Peephole Navigation bezeichnet.



“Static peephole navigation means moving the spatial layout behind a static peephole, i.e. traditional scrolling.” [6]

Die räumliche Ausdehnung der Informationen wird hier für gewöhnlich per Scrollbalken dargestellt und auf Handys oft mittels Miniaturjoystick oder bei einem Touchscreen mittels Fingergesten verschoben. Gerade bei den Gesten kommt es teilweise zu Missverständnissen: Einige Implementierungen nutzen eine Wisch-Geste um den virtuellen Informationsraum weiter zu schieben, während andere Umsetzungen dagegen die selbe Geste nutzen, um das Peephole zu verschieben.

Die zweite Variante ist eher der Ansatz des dynamischen Peephole:



“Dynamic peephole navigation means moving the peephole across a static spatial layout.” [6]

Bei einem dynamischen Peephole ist allerdings gemeint, dass das physikalische Gerät bewegt wird, um den sichtbaren Bildschirmausschnitt gleichermaßen zu verschieben. Vergleichbar ist dies mit der Nutzung einer Lupe zum Betrachten einer Landkarte oder eines Buches. Diese direkte Übertragung der tatsächlichen Bewegung in die virtuell dargestellte Information bietet eine gute “Natürlichkeit” des Interfaces nach Donald A. Norman, da hier keine neuen Gesten oder Ähnliches gelernt werden müssen. Die heutzutage genutzte Variante ist somit nicht als dynamisches Peephole zu bezeichnen, da hier durch neu zu erlernende Gesten nur der Bildschirmausschnitt manipuliert wird und der physische Device an seiner Position bleibt.

Ein dynamisches Peephole bietet weiter nicht nur die Möglichkeit für zweidimensionale Eingaben, sondern stellt ferner ein *Six degrees of freedom* (6DoF) Input dar. Als 6DoF wird die Fähigkeit eines Objektes im dreidimensionalen Raum, sich auf allen drei Achsen zu bewegen und sich um alle drei Achsen zu drehen, bezeichnet [vgl. 12]. Ein mobiler Handheld kann theoretisch all diese Achsen als Steuersignal nutzen, doch das technische Ermitteln aller Daten ist schwierig und meist sehr rechenaufwändig.

Ein weiterer Grund für ein dynamisches Peephole Interface ist, dass eine statische Peephole Navigation zu einem Orientierungsverlust führen kann [vgl. 13]. Die feststehende virtuelle Information erlaubt dem Nutzer eines dynamischen Peepholes, seinem räumlichen Gedächtnis zu glauben und so einfacher Relationen zu Objekten außerhalb des sichtbaren Bereiches herzustellen. Bakker et. al. zeigte mit einer virtual reality Anwendung, wo ein head-mounted Screen als dynamisches Peephole genutzt wurde, dass die räumliche Orientierung tatsächlich besser ist [vgl. 14].

Ein direkter Vergleich der Eigenschaften von dynamischen und statischen Peepholes wurden 2006 von Mehra, Werkhoven und Worrying an der Universität von Amsterdam mittels einer Usability-Studie durchgeführt [vgl. 15]. Beide Peephole-Varianten wurden mittels Computer und Maus realisiert. Die Maus diente jeweils der 2D-Bewegung. Des Weiteren wurde der Bildschirm schwarz abgedunkelt und die Information war nur in

einem kleinen Peephole sichtbar. Bei der statischen Variante verschoben die Probanden die Information mittels der Maus. Bei der dynamischen Variante war die Information fixiert und das Peephole wurde über dieser verschoben.

Die Probanden mussten bei diesem Experiment mit jeder Peephole-Variante die Längendifferenz zweier dargestellter Linien bewerten. Sie sollten jeweils bewerten, ob die Linien gleich lang sind oder nicht. Hierzu war es nötig, die Informationsfläche zu erkunden, da die Linien so angeordnet waren, dass nur ein Teil einer Linie zur gleichen Zeit mit dem Peephole betrachtet werden konnte. Die Linien-Längendifferenz wurde bei verschiedenen Durchläufen variiert.

Das Ergebnis war eindeutig positiv für das dynamische Peephole Interface, denn die Schwelle für die richtige Bewertung der Linien (Discrimination-Threshold) lag hier 50-75% niedriger als beim statischen Interface [15]. Des Weiteren ergab die Probandenbefragung, dass 80% ein dynamisches Peephole bevorzugen würde [15].

3.1 Aktuelle dynamische Peephole Implementierungen

Im Jahre 1993 ist das erste mobile dynamische Peephole Interface für ein Handheld von Fitzmaurice et al. entwickelt worden [vgl. 16]. Yee griff diese Arbeit sechs Jahre später auf und erweiterte sie mittels einer Stifteingabe [vgl. 17]. Zudem führte er auch eine Usability-Untersuchung durch und fand heraus, dass dynamische Peepholes effizienter sein können [vgl. 17].

Auf die beiden aktuellsten zwei dynamic Peephole Implementierungen mit Usability-Test soll im Folgenden eingegangen werden. Von Kun et. al. stammt die Forschungsarbeit "Peephole With No Strings Attached: Using Image Processing as Input in Camera-phones" (PNSA) [11] und als zweites soll auf "ISeeU: Camera-based User Interface for a Handheld Computer" von Sohn und Lee [18] eingegangen werden.

3.1.1 Peephole With No Strings Attaches

Kun et. al. nutzen für die Schätzung der Bewegung des Handhelds die integrierte Kamera. Aus zwei Hauptgründen ist die Kamera als "Bewegungssensor" ausgewählt worden: Erstens ist in jedem Smartphone eine Kamera eingebaut und zweitens ist die Bewegungsanalyse in der Computer Vision gut erforscht. Als Testumgebung wurde eine Kartenanwendung genutzt. Diese ist ideal für einen Test, da bei der Navigation durch eine Karte verschieden große Strecken in verschiedene Zoomstufen zurück gelegt werden. Zugleich muss die Steuerung sehr genau sein, wodurch hohe Anforderungen an jede Art von Interface gestellt werden.

Als Rahmenbedingungen wurden somit festgehalten: Das Interface muss in Realzeit arbeiten und darf keine merkbare Verzögerung aufweisen, da sonst Irritationen für den Nutzer entstehen. Der Rechenaufwand soll zudem sehr gering gehalten werden und die Anwendung muss ohne weitere Bedingung (with no Strings Attached), wie beispielsweise einem speziellen Hintergrund, funktionieren.

Als mathematisches Verfahren wurde die Pixelkorrelation eingesetzt. Jedes Bild wird mit dem gleichen Raster in kleine Teile eingeteilt. Jeder aktuelle Teilbereich wird dann mit dem gleichen Teilbereich im zeitlich vorausgehenden Bild mittels der Pixelkorrelation analysiert. Somit wird für jeden Teilbereich die Bewegung festgestellt. Die einzelnen Ergebnisse werden zu einem gemeinsamen Ergebnis formuliert und dann als Bewegung in der Karte umgesetzt.

Die Rechenintensität und die Genauigkeit stehen sich hier gegenüber und werden beide durch die Rastengröße bestimmt. Für eine genaue Bewegungserkennung müsste das Raster möglichst nur einen Pixel je Teil umfassen, dies ist auf einem mobilen Endgerät nicht mehr in Realzeit zu berechnen.

Ein weiteres Problem des Verfahrens ist, dass es extrem fehleranfällig auf Änderungen im aufgenommenen Bild selbst reagiert. Hinzu kommt, dass bei dieser Implementierung eine zeitliche Glättung der Bewegung eingebaut wurde, damit Fehler in der Bewegungserkennung ausgeglichen werden. Bei einem Richtungswechsel führt dies zu einer zeitlichen Verzögerung, welche laut Usability-Test den Nutzer irritiert.

Wie schon an anderer Stelle herausgefunden sollte auch hier herausgestellt werden, dass dynamische Peephole Interfaces den Nutzer schneller zum gewünschten Ziel führen. Weiter sollte gezeigt werden, dass ein solches Interface alltagsfähig ist.

Die Usability-Studie zeigte allerdings ein anderes Ergebnis: Die Nutzer waren nicht begeistert und das Interface war nicht schneller als das zum Vergleich herangezogene Joystick Interface. Die Probanden beschwerten sich über müde Arme, da große Bewegungen gemacht werden mussten, um einen Steuerbefehl auszulösen. Die Probanden fühlten sich auch "lächerlich" durch diese Bewegungen. Bei geringen und langsamen Bewegungen funktionierte das Interface allerdings zuverlässig und hier fand die Idee die Zustimmung der Probanden.

Hieraus lässt sich ableiten, dass der Algorithmus zu ungenau arbeitet, welches sicherlich zu einem großen Teil von den vielen Rundungen und Angleichungen der ungenauen Bewegungserkennung herrührt. Wie Shneiderman schon feststellte, wird das beste Interface vom Nutzer abgelehnt, wenn dieser nicht das Gefühl der Kontrolle hat [vgl. 19]. So ist auch hier zu erklären, dass die Probanden nicht die Interface-Idee als solche ablehnten, sondern die schlecht funktionierende Implementierung. Ein weiterer Grund für die Ungenauigkeit ist die Auslastung des mobilen Gerätes, welches keine genauere Berechnung zuließ. Somit sollte hier über alternative Methoden zur Bewegungserkennung nachgedacht werden.

Dennoch zeigt dieser Versuch, dass es möglich ist bzw. 2007 schon möglich war, ein solches Interface auf mobilen Geräten ohne weitere Hilfsmittel einzusetzen. Weiter wurde gezeigt, dass die Nutzer Wert darauf legten, nicht "lächerlich" zu erscheinen. Um dies zu erreichen, könnte man das dynamische Peephole Interface für kurze Distanzen einsetzen. Gleichzeitig würde so den Ermüdungserscheinungen vorgebeugt werden. Dies ist, wie in meiner BA-Thesis herausgestellt [4], ein wichtiges Kriterium für Interfacegestaltung.

3.1.2 ISeeU

ISeeU geht den gleichen Weg wie PNSA, um die Bewegung des Handhelds zu ermitteln, und nutzt eine Kamera als Bewegungssensor. Des Weiteren sind zwei Knöpfe am Gerät befestigt, welche zur Aktivierung und Deaktivierung des dynamischen Peephole-Mechanismus dienen. Der wichtigste Punkt ist aber, dass das mobile Endgerät ständig mit einem PC über ein Kabel verbunden ist. Der separate PC führt alle Berechnungen zur Bewegungsanalyse aus und sendet das Ergebnis an den Handheld zurück.

Die genutzte Kamera ist, anders als bei PNSA, auf den Nutzer gerichtet, da die Entwickler sagen, dass die Steuerung nicht mit Bezug auf die Umgebung sondern mit Bezug zum Nutzer erfolgen soll [vgl. 18]. Hierbei ist anzumerken, dass die Kamera zwar hauptsächlich das Nutzergesicht sieht, der Bezug zum Nutzer und nicht ausschließlich zum Gesicht hergestellt werden soll. Mit jedem Einzelbild, das die Kamera lieferte, werden mehrere Verarbeitungsschritte durchgeführt [vgl. 18], wobei der Kern der Bewegungsberechnung die Lucas-Kanade-Methode ist. Dies ist ein von Bruce D. Lucas und Takeo Kanade entwickeltes Verfahren zur Berechnung des optischen Flusses jedes einzelnen Pixels im Bild. Mit diesen Daten wurde dann auch wieder eine gemeinsame Bewegung berechnet, die als Steuersignal genutzt wurde. Durch das Zurückführen der einzelnen Punkte in die Standardabweichung kann ISeeU die Bewegung auf der Z-Achse als Zoomfaktor interpretieren.

Nicht alle Nutzer interpretierten die Zoomsteuerung gleich, wie der Usability-Test zeigte. Hier waren zwei Interpretationen vertreten: So war es für einige Probanden natürlich, dass sich der Zoomfaktor erhöhen würde, wenn der Device näher an den Körper geführt wird (vergleichbar mit einem Buch, das nahe vor das Auge gehalten wird). Andere Probanden fanden es natürlich, dass die Zoom-Stufe größer wird, wenn das mobile Gerät vom Körper wegbewegt wird. Diese Variante ist vergleichbar mit dem Heranführen einer Lupe an eine Landkarte. Da hier nicht eine der Varianten als falsch beurteilt werden kann, sollte es eine Einstellmöglichkeit geben, damit jeder Nutzer ein natürliches Interface erhält.

Der mit dem ISeeU durchgeführte Usability-Test konnte im Gegensatz zum Test des PNSA wiederum belegen, dass eine dynamische Peephole-Navigation effizienter als die

statische Variante ist. Der Test selbst bestand darin, einen Eintrag aus einer langen Liste auszuwählen und eine Subway-Station auf einem unbekanntem Plan zu finden. Verglichen wurde ISeeU mit einem herkömmlichen Pfeiltasten-Interface. Als Leistungsbewertung dient die durchschnittliche Zeit, die bis zum Erreichen des Ziels verstrichen war. Hier stellte sich heraus, dass ISeeU beim Scrollen in der Liste 37% schneller ist und bei der Suche auf der Map 27% schneller ist. Ein wichtiger Faktor dieser Implementierung ist meiner Meinung nach die Funktion der Tasten zum einfachen Aktivieren der dynamischen Funktionalität. So kann der Nutzer bequem das gefundene Ergebnis fixieren und schießt nicht über das Ziel hinaus. Dies passiert, bedingt durch die zeitliche Angleichung der tatsächlichen Bewegungen beim PNSA, immer.

Bei diesem Verfahren ist allerdings zu berücksichtigen, dass die Rechenleistung eines separaten PC gebraucht wird, um den aufwändigen mathematischen Algorithmus zu berechnen. Somit ist das Verfahren nicht mobil einsetzbar.

3.2 Potenzial & Grenzen von dynamischen Peepholes

Die Stärke der PNSA-Umsetzung liegt sicherlich darin, dass gezeigt werden konnte, dass es mit der Hardware von 2007 möglich ist, eine beinahe lauffähige dynamische Peephole-Navigation allein durch die Rechenleistung eines mobilen Endgerätes zu ermöglichen. Des Weiteren wurde durch die Usability-Studie aufgedeckt, dass der Nutzer sich nicht "lächerlich" fühlen will und nicht unendlich Kraft im Arm mitbringt. Auch wurde gezeigt, dass die Echtzeitfähigkeit von großer Bedeutung ist.

Mit ISeeU wurde bewiesen, dass der Ansatz des dynamischen Peephole-Interfaces eine natürliches Steuerkonzept für mobile Geräte ist. ISeeU konnte weiter belegen, dass es zusätzlich zur erhöhten Usability auch effizient benutzbar ist. Wie schon oben erwähnt ist das einfache Aktivieren und Deaktivieren der dynamischen Peephole-Steuerung meines Erachtens ein entscheidender Faktor. In Verbindung mit dem Ergebnis von PNSA, dass Nutzer die dynamische Steuerung gerne für kurzen Distanzen einsetzen, bietet sich das dynamische Peephole-Interface wahrscheinlich gut für die schnelle Erkundung der näheren Umgebung eines aktuellen Kartenausschnittes an.

Zusammenfassend konnten durch PNSA und ISeeU gezeigt werden, dass eine dynamische Peephole-Navigation die Interaktion mit einem mobilen Endgerät natürlicher gestaltet, wenn die Implementierung fehlerfrei funktioniert. Für den Einsatz auf einem mobilen Endgerät wurde gezeigt, dass es wichtig ist, echtzeitfähig zu sein, sehr wenig Ressourcen zu verbrauchen, keine besonderen Umgebungsbedingungen zu erfordern, sehr präzise Ergebnisse zu produzieren und sich auf eine Relation zwischen Nutzer und Gerät zu beziehen. Als wichtigste Kriterien für die Alltagstauglichkeit wurden meiner Meinung nach die Echtzeitfähigkeit und die Präzision herausgestellt.

3.3 Ausblick

Rohs und Essel haben 2007 eine vergleichende Studie durchgeführt, in der sie verschiedene Sensor-Technologien für die Bewegungsermittlung gegenüberstellten [vgl. 6]. Als drei Techniken wählten sie erstens eine Kombination aus Beschleunigungssensor und Magnetfeldsensor (Kompass), zweitens eine optische Markterkennung mittels Kamera und drittens eine Bewegungsberechnung aus dem Kamerabild ohne spezielle Marker und Bedingungen. Des Weiteren wurde das reine Peephole-Interface mit einem Peephole-Interface verglichen, welches zusätzlich eine Visualisierungstechnik namens Halo [vgl. 20] zusätzlich nutzte. Diese Ergebnisse sind in der Studie voneinander trennbar und hier nicht weiter von Bedeutung.

Es wurden zwei Studien mit jeweils 12 und 6 Teilnehmern durchgeführt. Die Teilnehmer mussten mit jeder Technologie ein Ziel unter verschieden vielen falschen Zielen finden.

Kumulativ wurde festgestellt, dass die Technologie mit optischen Markern führend ist. Diese ist für den mobilen Einsatz ungeeignet, da spezielle Umgebungen gebraucht werden. Bei den Technologien ohne spezielle Umgebungsvarianten leistet die Kombination von Kompass und Beschleunigungssensor mehr als die rein kamerabasierte Bewegungserkennung. Bei der Kombination von Kompass und Beschleunigungssensor-Technologie ermüden die Probanden später, was laut den Entwicklern von der frei veränderbaren Armposition herrührt.

Für mich zeigen diese Ergebnisse, dass die weiter oben vorgestellten Implementierungen (welche nach meiner Recherche auch die Aktuellsten sind) auf die falsche Technologie gesetzt haben. Aus den gemessenen Ergebnissen und aus der Bemerkung über die Ermüdung würde ich hier ansetzen, um eine alltagstaugliche Implementierung zu schaffen. Denn alle drei Technologien haben immer noch den Faktor der "Lächerlichkeit" durch ausladende Bewegungen. Unabhängig vom Peinlichkeitsempfinden des Nutzers sind diese in manchen Situationen, wo der mobile Nutzer in Eile oder in Bewegung ist, auch grundsätzlich nicht praktikabel. Ein gutes Beispiel hierfür ist die fehlende Nutzbarkeit in einer vollbesetzten U-Bahn, wo der Nutzer die anderen Passagiere durch seine ausladenden Bewegungen belästigen würde.

3.3.1 Speed-dependent Automatic Zooming

Zur Reduktion der Bewegungsgröße ist es denkbar, das Abbildungsverhältnis zu verändern. So setzen alle aktuellen Implementierungen auf ein 1:1-Verhältnis von realer Bewegung des Peephole-Gerätes zur Verschiebung des Sichtfensters im virtuellen Raum.

Das Peephole-Problem, dass der darzustellende Inhalt viel größer ist als der Bildschirm abbilden kann, gibt es schon lange gerade auch beim Webbrowser. Ein Lösungsansatz ist hier das "Speed-dependent Automatic Zooming" (SDAZ) [21].

Igarashi und Hinckley fanden heraus, dass Nutzer beim Suchen von Inhalten auf langen Webseiten ständig zwischen Zoomen und Scrollen wechselten. Dadurch erregt das Interface, welches eigentlich nur Mittel zum Zweck sein sollte, ein erhöhtes Maß an Aufmerksamkeit. Ein weiteres Problem von Scrollbars ist, dass diese bei sehr großen Dokumenten sehr empfindlich reagieren. So führen minimale Bewegungen an der Scrollbar zu großen Sprüngen im Dokument, was wiederum zu einer Irritation des Nutzers führt [vgl. 21]. Der Ansatz zu SDAZ liegt im distanzbewerteten Scrollen: Der Abstand zwischen der aktuellen Mausposition und der Position beim Beginn des Scrollvorgangs wird zur Steuerung der Scrollgeschwindigkeit genutzt. Sehr schnelles Scrollen führt hier allerdings zu einer Desorientierung. SDAZ nutzt diese Technik nun dazu, den Zoomfaktor ohne weitere oder andere Eingaben zu verändern. Das heißt, je größer die Distanz wird, umso schneller scrollt das Dokument und gleichzeitig wird aus

dem Dokument herausgezoomt. Das Ziel ist die wahrgenommene Scrollgeschwindigkeit auf ein- und demselben Level zu halten. Wird dies erreicht, kann der Nutzer unbegrenzt schnell scrollen, ohne die Orientierung zu verlieren.

Um das Ziel der verbesserten Usability zu erreichen, stellen Igarashi und Hinckley fest, dass sich der Zoom exponentiell zum Scrollen entwickeln muss und nicht wie erst angenommen proportional. Für die Nutzer ist es weitaus angenehmer, wenn der Zoom nicht direkt bei jeder kleinen Scrollbewegung einsetzt. Somit ist eine Zoom-Start-Schwelle nötig.

Ein Problem stellte das Loslassen der Maustaste dar. Diese aktivierte das SDAZ und somit führte die Freigabe zu einem abrupten Zoomsprung, welcher den Nutzer verwirrt. Als Lösung wurde eine Abkling-Funktion eingebaut, die den Zoom langsam auf das Ausgangslevel zurückführt. Diese lässt den Nutzer allerdings teilweise am gesuchten Objekt vorbei“fahren”.

Die Usability-Untersuchung nutzte einen Webbrowser, eine Kartenanwendung, einen Bildbrowser, ein Wörterbuch und einen Sound-Editor, um die Nutzbarkeit zu bewerten. Hier mussten nun jeweils vorgegebene Punkte oder Inhalte möglichst schnell gefunden und angeklickt werden. Die Studie wurde mit sechs Probanden durchgeführt und jeweils mit einer herkömmlichen Steuerung verglichen. Nach der Studie wurden die Probanden zu jeder Anwendung befragt, ob sie das SDAZ nutzen würden.

Die Ergebnisse waren nicht eindeutig, aber dennoch konnte gezeigt werden, dass die Nutzer das Interface, welches mit weniger separaten Steuerbefehlen auskommt, bevorzugen. Bei der Webapplikation, bei der nur auf einer Achse gescrollt wird, waren die Nutzer immer schneller als mit herkömmlichen Scrollbars. Fast alle würden dieses Interface auch bevorzugt nutzen. Die Kartennavigation fiel den Probanden allerdings schwerer und dies spiegelte sich in den gemessenen Zeiten wieder. Das Hauptproblem bestand in der schon erwähnten automatischen Abkling-Funktion, durch die die Nutzer oft weit über das Ziel hinwegschwebten. Trotzdem würden über die Hälfte auch hier das SDAZ nutzen. Ungeeignet ist die Idee von SDAZ beim Durchsuchen eines Wörterbuches, da hier beim Scrollen mit verkleinerter Ansicht nicht mehr jedes einzelne Wort gesehen und so gefunden werden kann.

4. Resümee

In dieser Ausarbeitung habe ich gezeigt, dass ein dynamische Peephole-Interface gerade für mobile Endgeräte eine gute Möglichkeit ist, die mit solchen Geräten einhergehenden Schwächen zu minimieren. Geräte die schlechte Usability von Smartphones kann durch ein gut funktionierendes dynamisches Peephole-Interface stark verbessert werden.

Die große Schwäche der dynamischen Peephole-Navigation liegt im großen beanspruchten Bewegungsraum und der damit verbundenen Ermüdung des Nutzers. Hier ist meine Idee ein dynamisches Peephole-Interface mit der Technik des Speed-dependent Automatic Zooming zu erweitern. Somit kann der Bewegungsraum hoffentlich reduziert werden und eventuell ist es sogar möglich, mit einer kleinen zweidimensionalen Bewegungsebene anstelle eines dreidimensionalen großen Raumes auszukommen.

Das klare Ziel des gesamten Prozesses soll es sein, dem Nutzer ein natürliches und somit intuitives Interface für den Alltagsgebrauch zu bieten.

Literaturverzeichnis

1. Radio Hamburg: Eine Woche Entzug: Horst ohne Handy. In: Radio Hamburg. (Accessed Februar 02, 2010) Available at:
<http://www.radiohamburg.de/content/view/full/176979>
2. IDC: Press Release prUS22196610. In: IDC. (Accessed Februar 04, 2010)
Available at: <http://www.idc.com/getdoc.jsp?containerId=prUS22196610>
3. Billiger-Telefonieren.de: Billiger-Telefonieren.de. In: iPhone von Apple: Infos zum Kulthandy iPhone. Available at: <http://www.billiger-telefonieren.de/iphone/>
4. Siedentopp, J.: Interfacerevolution bei Videospielekonsolen: Entwicklung von Heuristiken zur Gestaltung revolutionärer Interfaces. Hochschule für Angewandte Wissenschaften Hamburg, Hamburg, Germany (2009)
5. Norman, D.: Dinge des Alltags: gutes Design und Psychologie für Gebrauchsgegenstände. Campus-Verlag, Frankfurt/Main [u.a.] (1989)
6. Rohs, M., Essl, G.: Sensing-based Interaction for Information Navigation on Handheld Displays. In : MobileHCI 2007, Singapore (2007)
7. Wikipedia-Autoren: Gebrauchstauglichkeit (Produkt). In: Wikipedia. (Accessed Januar 04, 2010) Available at:
[http://de.wikipedia.org/w/index.php?title=Gebrauchstauglichkeit_\(Produkt\)&oldid=68835402](http://de.wikipedia.org/w/index.php?title=Gebrauchstauglichkeit_(Produkt)&oldid=68835402)
8. DIN ISO EN 9241-11.
9. Murphy, J., Kjeldskov, J., Howard, S., Shanks, G., Hartnell-Young, E.: The converged appliance: "I love it. but I hate it". In : 17th Australia conference on Computer-Human Interaction, Narrabundah (2005)
10. Wagner, C.: Mobile Usability Test für Pervasive Games. Hochschule für Angewandte Wissenschaften Hamburg, Hamburg, Germany (2008)
11. Kun, L., Lee, K.-H., Marsden, G.: Peepholes with no strings attached: using image processing as input in camera-phones. In : 2007 annual research conference of the South African institute of computer scientists and information technologists on IT research in developing countries, New York, p.8 (2007)
12. Wikipedia contributors: Six degrees of freedom. In: Wikipedia, The Free Encyclopedia. (Accessed Februar 13, 2010) Available at:
http://en.wikipedia.org/w/index.php?title=Six_degrees_of_freedom&oldid=343828509

13. Guiard, Y., Beaudouin-Lafon, M., Bastin, J., Pasveer, D., Zhai, S.: View size and pointing difficulty in multi-scale navigation. In : Working conference on Advanced visual interfaces, New York, pp.117 - 124 (2004) nach [15].
14. Bakker, N., Passenier, P., Werkhoven, P.: The effects of head-slaved navigation and the use of teleports on spatial orientation in virtual environments (VE). *Human Factors* 45, 160-169 (2003) nach [15].
15. Worrying, M., Werkhoven, P., Mehra, S.: Navigating on handheld displays: Dynamic versus static peephole navigation. *ACM Transactions on Computer-Human Interaction (TOCHI)* 13(4), 448 - 457 (Dezember 2006)
16. Fitzmaurice, G., Zhai, S., Chignell, M.: Virtual reality for palmtop computers. *ACM Transactions on Information Systems (TOIS)* 11(3), 197 - 218 (Juli 1993)
17. Yee, K.-P.: Peephole displays: pen interaction on spatially aware handheld computers. In : Conference on Human Factors in Computing Systems, New York, pp.1-8 (2003)
18. Sohn, M., Lee, G.: ISeeU: camera-based user interface for a handheld computer. In : 7th international conference on Human computer interaction with mobile devices & services, New York, vol. 111, pp.299 - 302 (2005)
19. Shneiderman, B.: User interface design Übers. aus dem Amerikan. von Jürgen Dubau & Arne Willner edn. Ill. mitp-Verlag, Bonn (2002)
20. Baudisch, P., Rosenholtz, R.: Halo: a technique for visualizing off-screen objects. In : SIGCHI conference on Human factors in computing systems, New York, pp.481 - 488 (2003)
21. Igarashi, T., Hinckley, K.: Speed-dependent automatic zooming for browsing large documents. In : 13th annual ACM symposium on User interface software and technology, New York, pp.139 - 148 (2000)