

Unterstützung der Touchinteraktion durch Annäherungserkennung auf Smartphones.

Jan-Hendrik Wolf

Matrikelnummer: 2616233

Mainstraße 36

28199 Bremen

© 0151 15530761

✉ mail@jan-wolf.de

Universität Bremen

Digitale Medien B. Sc.

Sommersemester 2014

Inhaltsverzeichnis

1. Einleitung	5
1.1. Gegenwärtiger Stand der Technik mobiler Endgeräte	7
1.1.1. Hardware	7
1.1.2. Software	8
1.2. Bisherige Leitfäden für Bedienelemente	9
1.3. Lösungsansätze mittels Annäherungserkennung	10
1.3.1. Schaltflächenerkennung	10
1.3.2. Tabellenhervorhebungen	11
1.3.3. Fingerpositionsmarkierungen	11
1.3.4. Vorhersage der Touchposition	11
2. Methodik	12
2.1. Testpersonen	12
2.2. Apparat	13
2.2.1. Test 1: Listenauswahl	13
2.2.2. Test 2: Präzisionstest	14
2.2.3. Ausbalancieren von Gewöhnungseffekten	18
2.2.4. Kalibrierung der Annäherungserkennung	18
2.2.5. Fragebögen	19
2.2.5.1. Personenfragebogen	21
2.2.5.2. System Usability Scale	21
2.2.5.3. Qualitativer Fragebogen	21
2.3. Prozedur	21
2.4. Design	24
3. Ergebnisse	26
3.1. Test 1: Listenauswahl	26
3.2. Test 2: Präzisionstest	26

4. Diskussion	27
4.1. Vergleich zu den bisherigen Leitfäden	27
4.2. Ausblick	27
5. Fazit	28
A. Anhang	29
A.1. Studieninformationen	30
A.2. Personenfragebogen	31
A.3. Bewertungsbögen	33
Literaturverzeichnis	40
Abbildungsverzeichnis	42

Zusammenfassung

Zusammenfassung der GESAMTEN Arbeit. NICHT INS INHALTSVERZEICHNIS

- Erklärung über "Internet der Dinge" ein bisschen unständlich.
Smartphones sind doch auch ohne sehr beliebt.
- Richtige Zeit verwenden bei Beschreibung der Methodik
- Keine Panik-Konstruktionen verwenden
- Zugehörigkeit zu Apparat bzw. Prozess prüfen
- Aufbau an sich aber gut!

1. Einleitung

Was ist das?

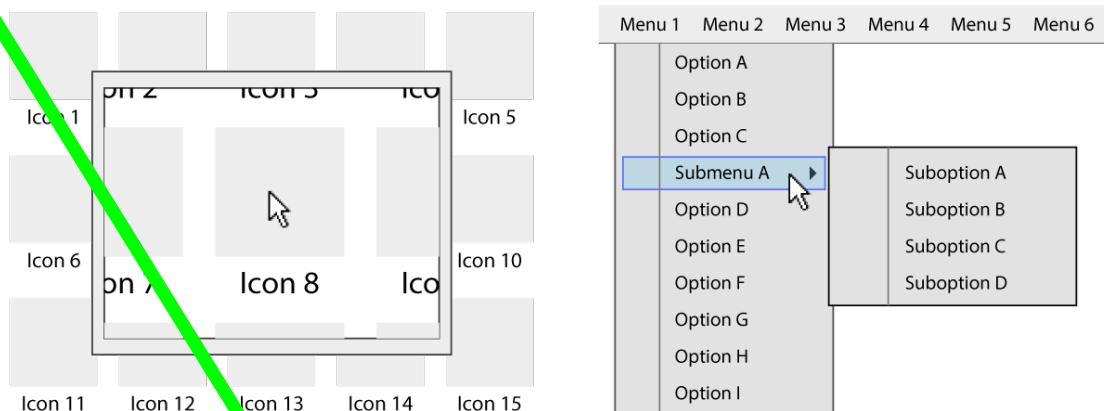
Im Alltag wird die Entwicklung des „Ubiquitous computing“ für viele Anwender zunehmend sichtbar. Viele periphere Geräte, wie z.B. in Bereichen der Haussteuerung, Überwachungstechnik und dem Unterhaltungsbereich beinhalten mittlerweile Computerchips, die sich über Smartphones vieler unterschiedlicher Hersteller konfigurieren und steuern lassen. Das Smartphone ist somit als Steuerungszentrale im „Internet der Dinge“ ein wichtiger Bestandteil geworden.

Somit ist es nicht verwunderlich, dass stetig neue Geräte dieser Art für unterschiedlichste Zielgruppen auf den Markt gebracht werden. Mittlerweile bieten sogar etablierte Hersteller bekannter Seniorenhandys Varianten in Form von Smartphones an [6]. Mit Einsteigergeräten, wie z.B. dem „Google One“ [3], werden auch Menschen mit geringem Einkommen angesprochen. Durch diese stetige Verbreitung nimmt gleichzeitig die Wichtigkeit einer adäquaten Interaktion auf mobilen Endgeräten zu.

Aufgrund der Tatsache, dass Anwender von den bisher weit verbreiteten Desktop-PCs und Laptops an große Bedienoberflächen mit umfangreichen Funktionen gewöhnt sind, besteht die Herausforderung auf mobilen Endgeräten auch darin, Anwendungen auf kleineren Bildschirmen ähnlich funktional auszustatten. Dies erfordert eine effiziente Ausnutzung der verfügbaren Bildschirmfläche.

Im Bezug auf die Eingabe zählt bei Desktop-PCs und Laptops die Maus und Tastatur zu den etablierten Eingabegeräten. Mobile Endgeräte sind jedoch nur dann besonders sinnvoll, wenn keine zusätzlichen Eingabegeräte nötig sind. Damit das Mitführen eines zusätzlichen Eingabegerätes entfällt, erfolgt die Interaktion auf vielen mobilen Endgeräten hauptsächlich über die Fingerberührung unter Mitnutzung zusätzlicher Sensoren, wie z.B. Neigungs- und Lichtsensoren. Zudem stellt sich der Finger als intuitives Eingabegerät heraus und erleichtert damit das Anwählen von Schaltflächen auf dem Bildschirm. Ab einer bestimmten Schaltflächengröße erhöht sich jedoch die Fehlerrate der Interaktion [7].

Grund hierfür ist der fehlende Mauszeiger als indirektes Eingabegerät. Während die Position des Mauszeigers die Position der Interaktion definiert, weicht die Position



(a) Zusätzliche, visuelle Ebene: Bildschirmlupe (b) Selektion ohne Ausführung: Menüauswahl

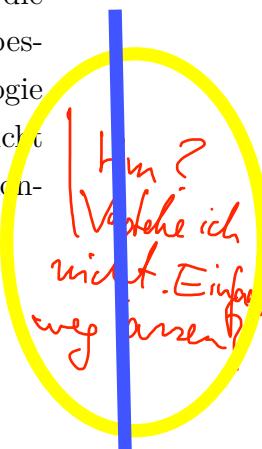
Abbildung 1.1.: Schematische Darstellung für „Hovering“ als Interaktionsform.

des Fingers von der Position der Interaktion meist leicht ab, da der Finger eine größere Berührungsfläche besitzt. Zusätzlich verdeckt der Finger durch ~~den~~ GröÙe kleine Schaltflächen, wohingegen der Mauszeiger weiterhin eine präzise Selektion ermöglicht. Um diesem sogenannten „Fat-Finger-Problem“ [8] entgegenzuwirken, ist die Mindestgröße der Bedienelemente auf diesen Bildschirmen im Vergleich zu Bedienelementen für Mausinteraktionen auf vielen Systemen vergröÙert worden [13]. Ein weiterer Vorteil der Mausinteraktion gegenüber der Interaktion durch Berührung ist der Umstand, dass die aktuelle Position des Mauszeigers auch vor dem Klicken auf der Bedienoberfläche sichtbar ist. Die Positionierung und der Klick mittels des Mauszeigers kann somit unabhängig voneinander erfolgen. Dies ermöglicht das Schweben des Mauszeigers über einer Schaltfläche und wird als „Hovering“ (engl. „Schweben“) bezeichnet. Auf Laptops und Desktop-PCs ist dies eine gängige Interaktionsform und wird meist verwendet, um die Eigenschaften von Objekten auf dem Bildschirm, wie z.B. das Aussehen, zu verändern. Durch verartige Eigenschaftsänderungen kann der Anwender das Objekt beispielsweise als Schaltfläche identifizieren. Abbildung 1.1 zeigt zwei Anwendungsbeispiele, die diese Interaktionsform nutzen.

Bisher war es jedoch nicht möglich, dass sogenannte „Hovering“ in das Repertoire der Interaktionsformen mobiler Endgeräte zu übernehmen. Dies änderte sich nachdem Sony im Mai 2012 erstmals ein Smartphone mit integrierter Annäherungserkennung für den gesamten Markt veröffentlichte. In einigen nativen Anwendungen von Sony ließ sich nun auch diese Interaktionsform auf einem berührungsempfindlichen Bildschirm verwenden [5]. Die Verbreitung und Integration der Annäherungserkennung ist bisher jedoch lediglich in einem kleinen Teil der ~~Geräte~~ erfolgt.

welcher finger?

In der vorliegenden Bachelorarbeit soll untersucht werden, ob visuelle Hover-Effekte auf berührungsempfindlichen Endgeräten mit integrierter Annäherungserkennung die Usability von Bedienelementen im Bezug auf Geschwindigkeit und Präzision verbessern können und auf diese Weise eine beschleunigte Integration dieser Technologie gerechtfertigt werden kann. Um die visuelle Darstellung der Bedienoberfläche nicht essentiell zu verändern oder zu überlagern, werden die Untersuchungen insofern konkretisiert, sodass auf zusätzliche, visuelle Ebenen verzichtet wird.



1.1. Gegenwärtiger Stand der Technik mobiler Endgeräte

Zum jetzigen Zeitpunkt ist die Anzahl der mobilen Geräte mit integrierter Annäherungserkennung stark begrenzt. Außerdem nutzen die Benutzungsoberflächen auf heutigen Smartphones mit Annäherungserkennung die Möglichkeiten der Technologie nur in vereinzelten Fällen aus. Im Folgenden soll ein kurzer Überblick auf die gegenwärtige Hardware und Software gegeben werden, die diese Annäherungserkennung unterstützt.

1.1.1. Hardware

Bereits im Jahr 1998 erwähnte der Touchpad-Hersteller Synaptics in einem Dokument die allgemeine Funktionsweise und diverse Spezifikationen zur Annäherungserkennung auf berührungsempfindliche Oberflächen und bestätigte damit die technische Möglichkeit [12]. Die Erkennung funktioniert theoretisch auf allen kapazitiven Oberflächen. Dazu wird die Kapazitätstärke des jeweiligen Fingers ermittelt und anschließend die ungefähre Entfernung des Fingers zur Oberfläche approximiert.

Die Technologie erlangte jedoch erst im Jahr 2012 die Aufmerksamkeit der Öffentlichkeit. Sony vermarktete zu dieser Zeit mit dem Xperia™ sola das erste Smartphone mit Annäherungserkennung und bewarb die Technologie unter dem Namen Floating Touch™.

Zu Beginn des Jahres 2013 brachte als zweiter Hersteller auch Samsung mit dem Galaxy S4 ein Gerät mit ähnlicher Technologie namens AirView™ auf den Markt. Im Gegensatz zu Sony, integrierte Samsung im Jahr 2014 die Annäherungserkennung auch in das Nachfolgemodell, den Samsung Galaxy S5.

das

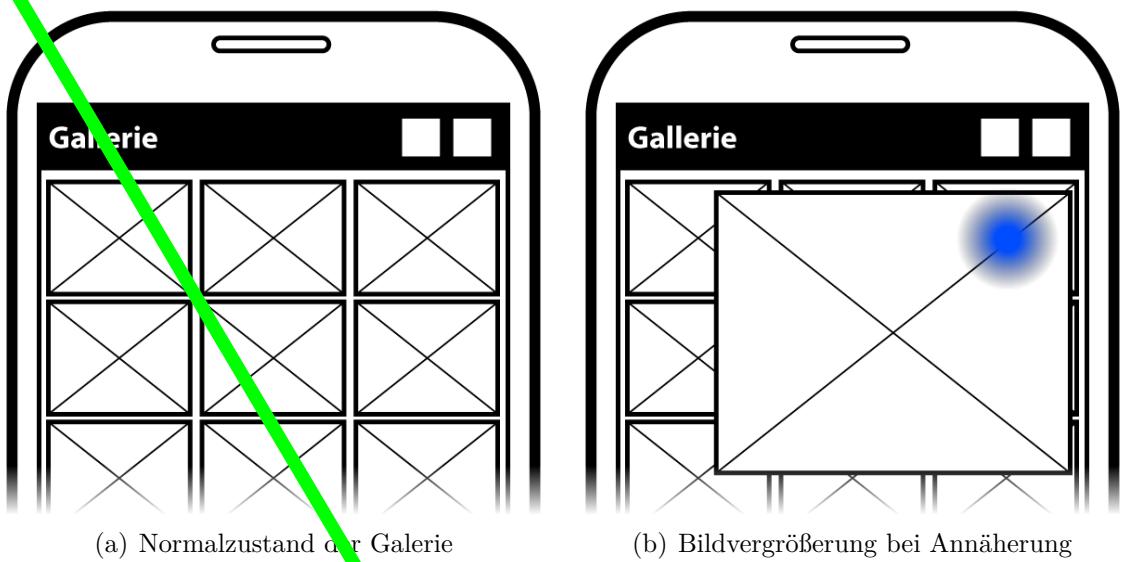


Abbildung 1.2.: Objektvergrößerungen bei Annäherung des Fingers mit aktiviertem AirView™ auf einem Samsung Galaxy S4.

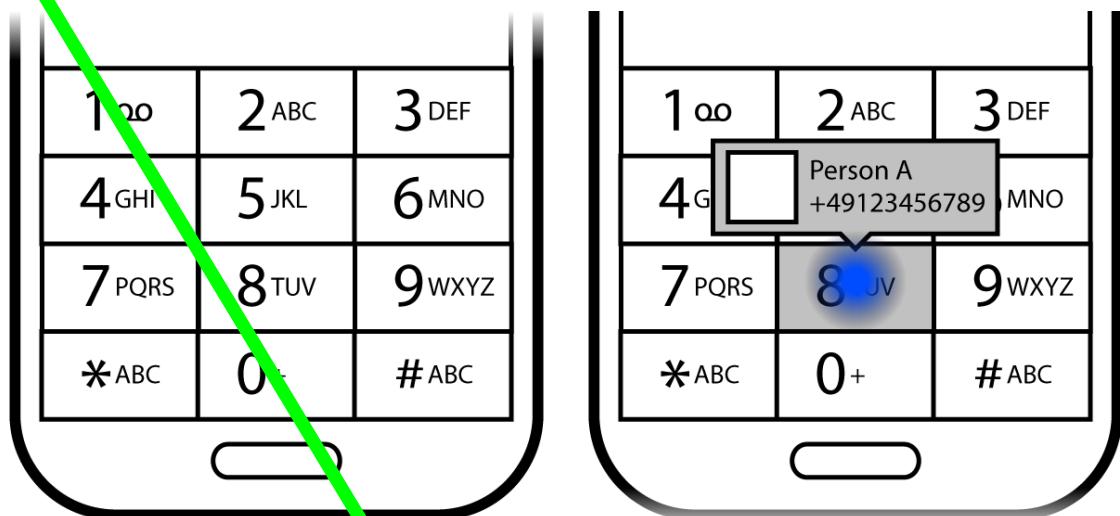
1.1.2. Software

In mobilen Endgeräten mit integrierter Annäherungserkennung wurden unterschiedliche Ansätze zur Nutzung dieser Technologie in Benutzeroberflächen verfolgt.

Sony verwendete die Technologie, um Hintergrundbilder auf dem Hauptbildschirm während der Fingerannäherung zu bewegen. Zusätzlich wurde der interne Browser des Smartphones angepasst, sodass die Annäherungserkennung auch auf Webseiten verwendet werden konnte. Die auf Webseiten häufig verwendeten und auf Annäherung sensiblen Bedienelementen waren somit erneut im vollen Umfang nutzbar.

Der Hersteller Samsung bietet in ~~einigen~~ Smartphones ebenfalls Anwendungen an, die auf Annäherung reagieren. Sie lassen sich in zwei Arten unterteilen: In der ersten werden Informationen auf kleinem Raum durch Annäherung vergrößert bzw. neu angeordnet, sodass sich die Übersicht für den Nutzer verbessert. Als Beispiel dient hier die Galerieanwendung, siehe Abbildung 1.2. In der Galerie werden Bilder bei Annäherung vergrößert, sodass der Nutzer eine Vorschau erhält. Die zweite Art der Anwendungen zeigen bei Annäherung nicht Vergrößerungen oder detailliertere Ausführungen, sondern zuvor nicht sichtbare Informationen an. Somit können weitere Zusatzinformationen eingeblendet werden, siehe Abbildung 1.3. Als Beispiel dient der Wahlbildschirm, in der bei Annäherung auf eine aktive Kurzwahl Taste der darauf jeweils gespeicherte Kurzwahlkontakt angezeigt wird.

*Verweis
auf
Top*



(a) Normalzustand des Wahlbildschirms

(b) Kurzwahlinformationen bei Annäherung

Abbildung 1.3.: Einblenden zusätzlicher Informationen bei Annäherung des Fingers auf einem Samsung Galaxy S4 mit aktiviertem AirView™.

1.2. Bisherige Leitfäden für Bedienelemente

Aufgrund der geringeren Abmessungen mobiler Endgeräte im Vergleich zu statio-nären Geräten und der direkten Bedienung mittels berührungssempfindlicher Bild-schirme, ist die Gestaltung der Bedienelemente ein entscheidendes Kriterium für die reibungslose Interaktion zwischen Mensch und Maschine. Alle Hersteller der popu-lären mobilen Betriebssysteme bieten umfangreiche Leitfäden an, die Empfehlungen für Anwendungsentwickler zur Gestaltung von Bedienelementen darstellen. In diesen Leitfäden werden u.A. Mindestangaben zu Größen und Abständen kommuniziert, siehe Tabelle 1.1. Wohingegen in Leitfäden für Desktop-Computer keine empfoh-leinen Mindestgrößen genannt werden.

Auch Studien zeigen, dass diese Leitfäden durchaus nötig sind. So arbeiten Nut-zer mit Mausinteraktionen gegenüber Berührungsinteraktionen bisher präziser und schneller. Darüber hinaus wird die Interaktion mit der Maus bei Aufgaben, die ge-naue Selektionen benötigen, bevorzugt [1]. Aus diesen Gründen sollen die genannten Untersuchungen in dieser Bachelorarbeit auch prüfen, ob die Mindestgrößen unter Anwendung der Annäherungserkennung verkleinert werden können und die Präzisi-on somit gesteigert werden kann.



Plattform	Minimale Größe	Empfohlene Größe
Android [11]	7mm	9mm
Apple [2]	7mm	7mm
Windows [10]	7mm	9mm

Tabelle 1.1.: Mindestgrößen von Bedienelementen.

1.3. Lösungsansätze mittels Annäherungserkennung

Durch die Annäherungserkennung werden Interaktionen auf einer weiteren Ebene ermöglicht und erweitern das Repertoire für den Designer einer Anwendung. Sony und Samsung haben mittlerweile Anwendungszenarios für die Annäherungserkennung demonstriert und damit gleichzeitig interessante Lösungen vorgestellt. Dieser Abschnitt soll diese Palette an Möglichkeiten erweitern und gleichzeitig den Bezug zu den behandelten Tests herstellen.

1.3.1. Schaltflächenerkennung

Das Aussehen einer Schaltfläche variiert zwischen den unterschiedlichen Handymodellen, Betriebssystemen (z.B. iOS, Android) und Gerätetypen. Beispielsweise besitzen Schaltflächen auf einem mobilen Endgerät unter iOS andere Erscheinungsmerkmale als mobile Endgeräte unter Android. Dieser Umstand wird durch die Etablierung des Natural User Interfaces [1] verstärkt, da es die Abgrenzungen zu klassischen Schaltflächen aufweicht. Anstatt von Schaltflächen werden durch Wischgesten innerhalb bestimmter Bildschirmbereiche neue Aktionen ausgeführt. So werden mittlerweile für digitale Karten die klassischen Schaltflächen verstärkt durch Zoom- und Wischgesten ersetzt und ähneln damit der analogen Nutzung von Papierkarten.

Die Folge daraus könnten Schaltflächen sein, die für einzelne Nutzer nicht mehr als solche zu erkennen sind. Während auf Geräten mit Mausinteraktion die Listeneinträge bei Überlagerungen mit dem Mauszeiger hervorgehoben werden und damit als Schaltfläche erkennbar werden, fehlen diese Indikatoren bisher bei Bedienoberflächen für Geräte mit berührungsempfindlichen Bildschirmen. Die Annäherungserkennung könnte diese Lücke füllen und die aktuelle Fingerposition, ähnlich der Mauszeiger-

position, in das Erscheinungsbild der Bedienoberfläche einbeziehen.

1.3.2. Tabellenhervorhebungen

Durch die Annäherungserkennung bietet sich ein interessanter Lösungsansatz für eine mögliche Vereinfachung der visuellen Orientierung in vielspaltigen Tabellen an. Beim Suchen von einzelnen Einträgen in großen Tabellen ist bei manchen Nutzern zu beobachten, dass einzelne Zeilen mit dem Finger überfahren werden. Vielleicht könnte dieses Verhalten ausgenutzt werden, um den Nutzer in dieser Situation durch das Hervorheben ganzer Tabellenzeilen zu unterstützen. Im Rahmen dieser Bachelorarbeit wird eine derartige Tabellenhervorhebungen implementiert und getestet.

1.3.3. Fingerpositionsmarkierungen

Tastatur! In vielen Situationen kann es vorkommen, dass mehrere Schaltflächen auf engen Raum nebeneinander platziert sind, z.B. innerhalb unangepasster Webseiten auf kleinen Bildschirmen oder in Form von selektierbaren Straßenzügen oder Landesgrenzen in Kartenanwendungen bei großen Maßstäben. Durch die Annäherungserkennung ließe sich die aktuell durch den Finger überflogene Schaltfläche hervorheben (vgl. Schaltflächenerkennung) und dadurch eventuell die Selektierung durch den Nutzer vereinfachen. Eine mögliche Verbesserung der Usability durch Fingerpositionsmarkierungen soll im Rahmen dieser Bachelorarbeit überprüft werden.

1.3.4. Vorhersage der Touchposition

Die Position korrigiert nicht. Das macht die Software. Als Ergänzung zur Fingerpositionsmarkierung wäre mittels der Annäherungserkennung eine Vorhersage der nachfolgenden Touchposition möglich, indem beispielsweise innerhalb eines definierten Radius die Schwebeposition des Fingers die Touchposition korrigiert. Schwebt der Finger eines Nutzer für eine Weile über eine sich hervorhebende Schaltfläche, so würde nach dieser Interpretation selbst eine anschließende, leicht verfehlte Berührung neben ~~die~~ *der* Schaltfläche ausreichen, um diese auszulösen. Diese Ergänzung soll ebenfalls in einen der Tests dieser Bachelorarbeit evaluiert werden. Es ist dabei zu prüfen, ob somit die Genauigkeit auch bei kleinen Schaltflächen erhöht werden kann, die die Mindestgrößen laut der Leitfäden für Bedienelementen unterschreiten.

2. Methodik

In diesem Kapitel wird erläutert, wie die oben genannten Ansätze anhand von verschiedenen Usability-Tests überprüft werden.

2.1. Testpersonen

Die Prototypen wurden in Usability-Tests von 18 Personen aufgiebig untersucht. Die Personen waren zwischen 20 und 35 Jahre alt (med: 23 Jahre, Ø: 25,5 Jahre). Beide Geschlechter (8 weiblich, 10 männlich) waren ausgewogen am Test beteiligt. Das berufliche Umfeld der Testpersonen konzentrierte sich überwiegend auf den universitären Bereich. Zudem besaßen alle Testpersonen gute Erfahrungen in der Nutzung von Desktop-PCs, Laptops, Tablets oder Smartphones. Die dazugehörige Gewichtung im Bezug auf die Nutzung von Desktop-PCs, Laptops und anderen berührungsempfindlichen Geräten sind in Tabelle 2.1 dargestellt. Lediglich eine Person nahm innerhalb der letzten sechs Monate an Studien zu Interaktionen an Mobilgeräten oder an Studien der Digitalen Medien AG der Universität Bremen teil.

Studenten oder Mitarbeiter?

	Desktop-PC	Laptop	Tablet	Smartphone	Sonstige
Weniger als 1x pro Monat	50%	0%	59%	6%	56%
Weniger als 1x pro Woche	0%	6%	24%	0%	22%
Weniger als 1x pro Tag	17%	0%	12%	0%	17%
Mehrmals am Tag	33%	94%	6%	94%	6%
Am häufigsten benutzt	17%	44%	0%	39%	0%

Tabelle 2.1.: Gerätenutzung der Testpersonen.

Die Auswahl sollte (VPN) lässt keine Angaben mit einer solchen Funktionalität zu.

~~2.2. Apparat~~

Die Testumgebung besteht aus einem Testgerät und der zugehörigen für die Untersuchungen konzipierten Testanwendung.

Bei dem Testgerät handelt es sich um ein Samsung Galaxy S4. Das darauf installierte Betriebssystem basiert auf Android in der Version 4.4 (API-Level: 19). Der kapazitive Touchscreen des Smartphones mit einer Bildschirmdiagonalen von 5 Zoll besitzt eine Auflösung von 1920×1080 Pixeln und unterstützt die Annäherungserkennung.

~~Prozessur!~~ Das Smartphone darf während der Tests von den Testpersonen festgehalten, jedoch nicht von der Tischfläche angehoben werden.

Die Testumgebung beinhaltet als zweite Komponente die Testanwendung, die den Testpersonen unterschiedliche Aufgaben stellt. Die Testanwendung wurde in Java entwickelt und eigens für das Samsung Galaxy S4 konzipiert. Dazu wurde zudem die Bedienoberfläche der Testanwendung auf die entsprechende Bildschirmgröße angepasst und die von Samsung dokumentierte API für AirView™ in die Testanwendung integriert. Schließlich wurde die Testanwendung mittels des Android-SDKs in Version 19 kompiliert und auf ~~das~~ Gerät installiert. Der Quelltext liegt in digitaler Form auf dem beiliegenden Datenträger dieser Bachelorarbeit vor. Zusätzlich wird ~~an dieser Stelle~~ auch eine installierbare APK-Datei zur Verfügung gestellt.

Innerhalb der Testanwendung wird zu Beginn jeder Testsitzung eine Identifikationsnummer angezeigt, die für die jeweilige Testsitzung bestimmt wurde. Während jeder Testsitzung werden von der Testanwendung alle relevanten Interaktionen des Nutzers protokolliert und mit der zugehörigen Identifikationsnummer in eine auf dem Gerät angelegten Textdatei gespeichert.

Die Testanwendung ist in zwei unterschiedliche Tests unterteilt und wurde so gestaltet, dass vor jedem Testbeginn eine entsprechende Testbeschreibung angezeigt wird, siehe Abbildung 2.1.

2.2.1. Test 1: Listenauswahl

Die Testanwendung ermöglicht im ersten Test, die Wirkung von Hervorhebungen der Listeneinträge bei Annäherung zu überprüfen, siehe Abbildung 2.2.

Hierzu wird dem Nutzer ein Anweisungsbildschirm mit zwei pseudozufälligen Personeneigenschaften eingeblendet. Nachdem der Nutzer sich die Eigenschaften eingeprägt und anschließend auf „Starten“ gedrückt hat, wird eine mehrspaltige Perso-



Abbildung 2.1.: Testbeschreibung des ersten Tests.

nenliste mit sieben fiktiven Personen und ihren Eigenschaften angezeigt. Nähert sich der Finger des Nutzers nun einem Listeneintrag, so wird dieser hervorgehoben. Der Test ist so aufgebaut, dass pro Anweisung nur exakt eine Person die gesuchten Eigenschaften besitzt. Der Nutzer hat anschließend eine korrekte Person auszuwählen, indem dabei die Schaltfläche mit dem entsprechenden Personenname (Buchstaben von A-G) berührt wird. Sollte die falsche Schaltfläche berührt werden, so wird der Nutzer mittels einer temporären Texteinblendung (konkret: „Toast-Notification“) über die Falschberührung informiert und erneut an die zu suchende Eigenschaften erinnert. War die Personenauswahl korrekt, so werden dem Nutzer in weiteren Runden neue Eigenschaften auf dem Anweisungsbildschirm angezeigt. Der Test ist beendet, sobald der Nutzer sieben unterschiedliche Anweisungen befolgt hat.

*Dahinterste
unknkt?*

Um zu überprüfen, ob die Hervorhebungen der Listeneinträge bei Annäherungen einen Einfluss auf die Aufgabenerfüllung gehabt haben, wird der Test ein zweites Mal wiederholt. Bei der Wiederholung wird je nach Testvariante dann die Annäherungserkennung zu- oder abgeschaltet.

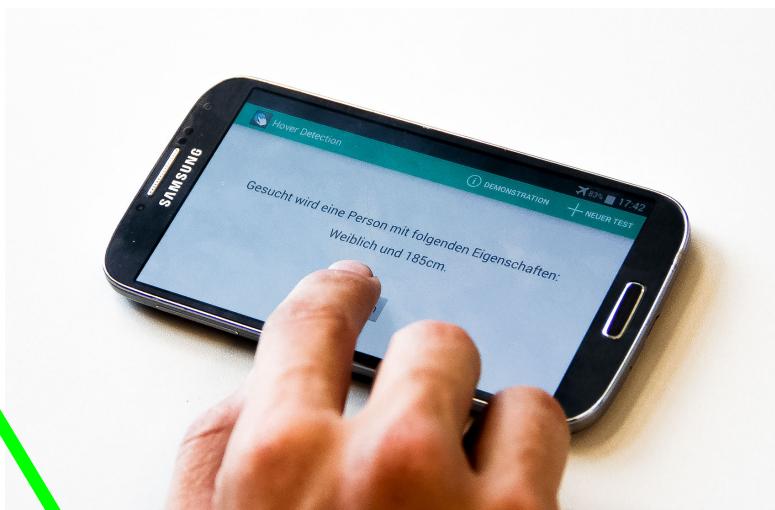
*d.h. + Trials ohne Hover und
dann bei 9VPN nochmal ohne und
bei 9VPN mit?*

2.2.2. Test 2: Präzisionstest

Die Testanwendung ermöglicht im zweiten Test, die Wirkung von Hervorhebungen auf die empfohlene Mindestgröße von Bedienelementen zu überprüfen, siehe Abbildung 2.3.

Die Bedienoberfläche zeigt einen Anweisungstext an, die den Nutzer auf das Halten der grünen, runden Schaltfläche am unteren Bildschirmrand hinweist. Um diese

*auf die
Präzision
bei der Aus-
wahl kleiner
Bedienele-
mente*



(a) Phase 1: Zu suchende Person wird beschrieben.



(b) Phase 2: Anzeige einer pseudofälligen Personenliste.

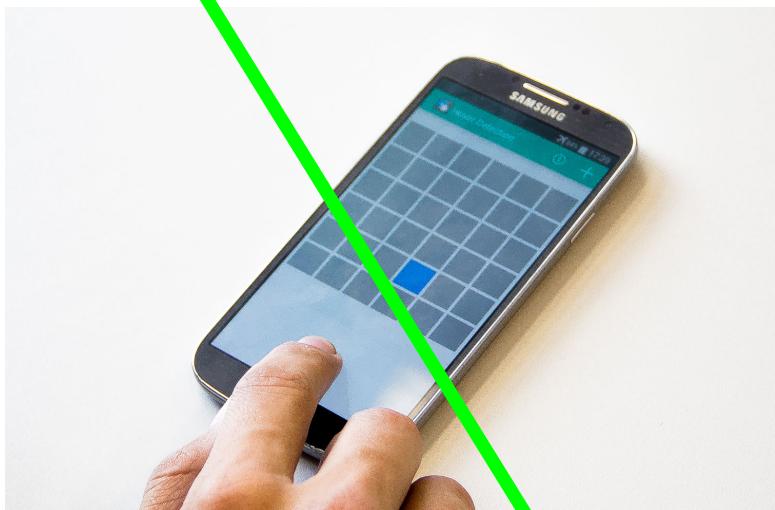


(c) Phase 3: Gesuchte Person in der unteren Leiste auswählen.

Abbildung 2.2.: Aufbau von Test 1: Drei unterschiedliche Phasen pro Durchgang.



(a) Phase 1: Halten der grünen Schaltfläche.



(b) Phase 2: Blaue Schaltfläche leuchtet pseudozufällig auf.



(c) Phase 3: Berühren der blauen Schaltfläche.

Abbildung 2.3.: Aufbau von Test 2: Drei unterschiedliche Phasen pro Durchgang

Ach?

Nein, das
ist schon
die ganze
Zeit da
hervorge-
hoben

Schaltfläche zu halten, muss diese berührt und der Finger nicht angehoben werden. Anschließend läuft für eineinhalb Sekunden ein Fortschrittsbalken ab. Die Testanwendung blendet im nächsten Schritt ein quadratisches Gitter mit quadratischen, grauen Schaltflächen ein. Eine graue Schaltfläche wird pseudozufällig von der Testanwendung ausgewählt und blau koloriert. Der Nutzer hat diese Schaltfläche im Anschluss zu berühren. Ist die Annäherungserkennung aktiviert, wird diejenige Schaltfläche aktiviert, die die momentane Fingerschwebeposition überspannt. Wird nun der Bildschirm berührt, so wird dem Nutzer über die Einblendung eines Anweisungsbildschirms die erfolgreiche oder misslungene Berührung mitgeteilt. Die Berührung gilt als erfolgreich, wenn die blaue Schaltfläche als erstes berührt worden ist. Ist die Berührung misslungen, so wird die Kolorierung solange zurückgesetzt, bis der Nutzer die grüne Schaltfläche erneut und für eine zusätzliche Sekunde gehalten hat. Der Nutzer hat anschließend vier weitere Versuche, um die blaue Schaltfläche zu treffen, andernfalls wird der Test vorläufig beendet. Neben den bereits erwähnten Benachrichtigungen, wird bei einer erfolgreichen Berührung dem Nutzer die Erhöhung des Schwierigkeitsgrades signalisiert. Gleichzeitig wird die grüne Schaltfläche wieder am unteren Bildschirmrand eingeblendet und der Nutzer kann den Test im nächsten Schwierigkeitsgrad fortsetzen. Der Test wird beendet, sobald alle Schwierigkeitsgrade von Nutzer bewältigt worden sind.

Die Testanwendung ist so konzipiert, dass beliebig viele Schwierigkeitsgrade und Schaltflächengrößen getestet werden können. Die Größe der Schaltfläche kann für jeden Schwierigkeitsgrad individuell angepasst werden. Die Testanwendung ist so gestaltet, dass neben dem obligatorischen Zeitpunkt der Interaktion sowohl die Position der grünen und blauen Schaltflächen auf dem Bildschirm, als auch die genauen Berührungspunkte auf diesen Schaltflächen selbst protokolliert werden. Im Falle einer eventuellen Formelanwendung nach Fitts [9] sind somit alle Parameter für jeden Versuch und Schwierigkeitsgrad vorhanden oder kalkulierbar.

Um auch in diesem Test die Auswirkungen der Annäherungserkennung zu untersuchen, wird dieser Test ebenfalls wiederholt. Dieser Test benötigt aufgrund der drei Testvarianten zwei Wiederholungen. Während einer Wiederholung werden die Hervorhebungen zu- oder abgeschaltet, in der anderen die softwareseitige Selektionshilfe.

Die softwareseitige Selektionshilfe manipuliert dabei die Eingabe des Nutzers. Anstatt die tatsächliche Berührungsposition des Fingers zu berücksichtigen, wird die Schwebeposition der Annäherungserkennung verwendet. Diese Manipulation wird

jedoch nur getätigt, wenn der Positionsunterschied der Schwebefingerposition und der Berührungsposition X_{Kmm} unterschreitet.

2.2.3. Ausbalancieren von Gewöhnungseffekten

Damit alle erhobenen Messwerte, u.A. die Auswahlgeschwindigkeiten, nicht durch etwaige Gewöhnungen während vorhergehender Testdurchgänge verfälscht werden und dass die statistische Auswertung verfälschungsfrei ist, muss die Reihenfolge auf die Testteilnehmer gleichmäßig variiert werden.

Hierzu sind die Testpersonen gemäß des Latin Squares in unterschiedliche Gruppen einzuteilen. Aus diesem Grund wurde bei der Konzeption der Testanwendung darauf geachtet, dass die Tests in unterschiedlichen Reihenfolgen gestartet werden können. Zudem ist die Reihenfolge der Ereignisse in jedem einzelnen Test pseudozufällig, d.h. aus Sicht des Nutzers rein zufällig, ausgewählt worden. Tatsächlich durchläuft jeder Nutzer in einer bestimmten Gruppe jedoch die exakt gleichen Aufgaben. Dies soll sicherstellen, dass etwaige Gewöhnungen auch gleichmäßig auf die Testvarianten aufgeteilt werden.

Beispielweise wird im ersten Test jede Testvariante gleichmäßig auf die erste, zweite oder dritte Wiederholung gelegt. Gleichzeitig werden dadurch die Anweisungen aufgrund ihrer fixen Reihenfolge auf jede Testvariante ausgeglichen verteilt. Der selbe Ansatz wird für die Personenliste genutzt: Entsprechend der Anweisung werden den Personen andere Eigenschaften zugeordnet. Dementsprechend ist auch im zweiten Test die Positionsreihenfolge der blauen Zielschaltflächen an die jeweilige Schaltfläche gekoppelt. Somit werden auch im zweiten Test alle Positionen gleichmäßig von jeder Testvariante verwendet.

Fehler sind jedoch nicht ausbalanciert!

2.2.4. Kalibrierung der Annäherungserkennung

Während der Implementierungsphase des Prototyps konnten im Bezug auf die ermittelte Schwebeposition des Fingers auffallende Verschiebungen zur tatsächlichen Fingerposition beobachtet werden. Es war somit anzunehmen, dass bei der Annäherungserkennung noch bestehende Ungenauigkeiten vorliegen. Weiterhin kann je nach Anstellwinkel des Fingers der dichteste Punkt zum Bildschirm variieren und dies möglicherweise eine zusätzliche Ungenauigkeit zur Folge haben. Beim internen Testen des Prototypen zeigten sich teilweise deutliche Unterschiede zwischen den Schwebefingerpositionen und den Touchpositionen. Resultierend daraus wurden die kleinen Quadrate

im zweiten Test unter Umständen nicht mehr korrekt hervorgehoben, bevor diese schließlich berührt werden würden.

Um die Verschiebungen zwischen der Schwebefinger- und Touchposition des Fingers nachzuvollziehen, wurde dem Testprototypen eine optional zuschaltbare Ebene innerhalb der Benutzungsoberfläche hinzugefügt, die die letzte Touch- und Schwebeposition anhand von Kreuzen sichtbar machte. Bei internen Versuchen mit dem Testprototypen wurde deutlich, dass in bestimmten Bereichen die beiden Kreuze nicht aufeinander lagen und die Touch- und Schwebeposition sich somit unterschiedlich stark unterschieden.

Um mögliche Zusammenhänge der Verschiebungen zwischen der Touch- und Schwebeposition des Fingers nachzuweisen, wurde eine Testanwendung implementiert, die die letzte Schwebefinger- und Touchposition aufzeichnete. Dabei sollten Nutzer eine Abfolge von verschiedenen Punkten auf dem Bildschirm berühren. Die jeweiligen Schwebefinger- und Touchpositionen waren für die Testperson zu diesem Zeitpunkt nicht sichtbar und wurden erst nach dem eigentlichen Test visualisiert. Anschließend wurden die Werte mittels eines eigens geschriebenen Hilfsprogramms ausgewertet und visualisiert.

Die Abbildung 2.4 zeigt die relativen Verschiebungen zwischen der Touch- und Schwebeposition an und verdeutlicht gleichzeitig die teilweise sehr starken Schwankungen beider Werte. Obwohl an mehreren Stellen Tendenzen zur Verschiebungsrichtung erkennbar sind, besitzen die Werte dennoch eine große Streuung. Die Möglichkeit, die Schwebepositionen auf die Touchpositionen zu verschieben (und andersherum), wären theoretisch möglich, z.B. durch lineare oder polynomiale Regression. Da die Ermittlung dieser Variablen den Umfang dieser Bachelorarbeit jedoch überstiegen hätte, wurde eine Kalibrierung im Rahmen dieses Tests nicht verfolgt. Stattdessen wurde für den zweiten Tests eine weitere Testvariante mit aktivierter Selektionshilfe implementiert.

~~Touchkinematik~~

2.2.5. Fragebögen

~~Was ist genügend?~~

Um für die spätere Auswertung ~~genügend~~ Informationen über die jeweiligen Testpersonen zu erhalten, ~~sollten~~ jeder Testperson mehrere Fragebögen ausgehändigt werden.

~~würden~~

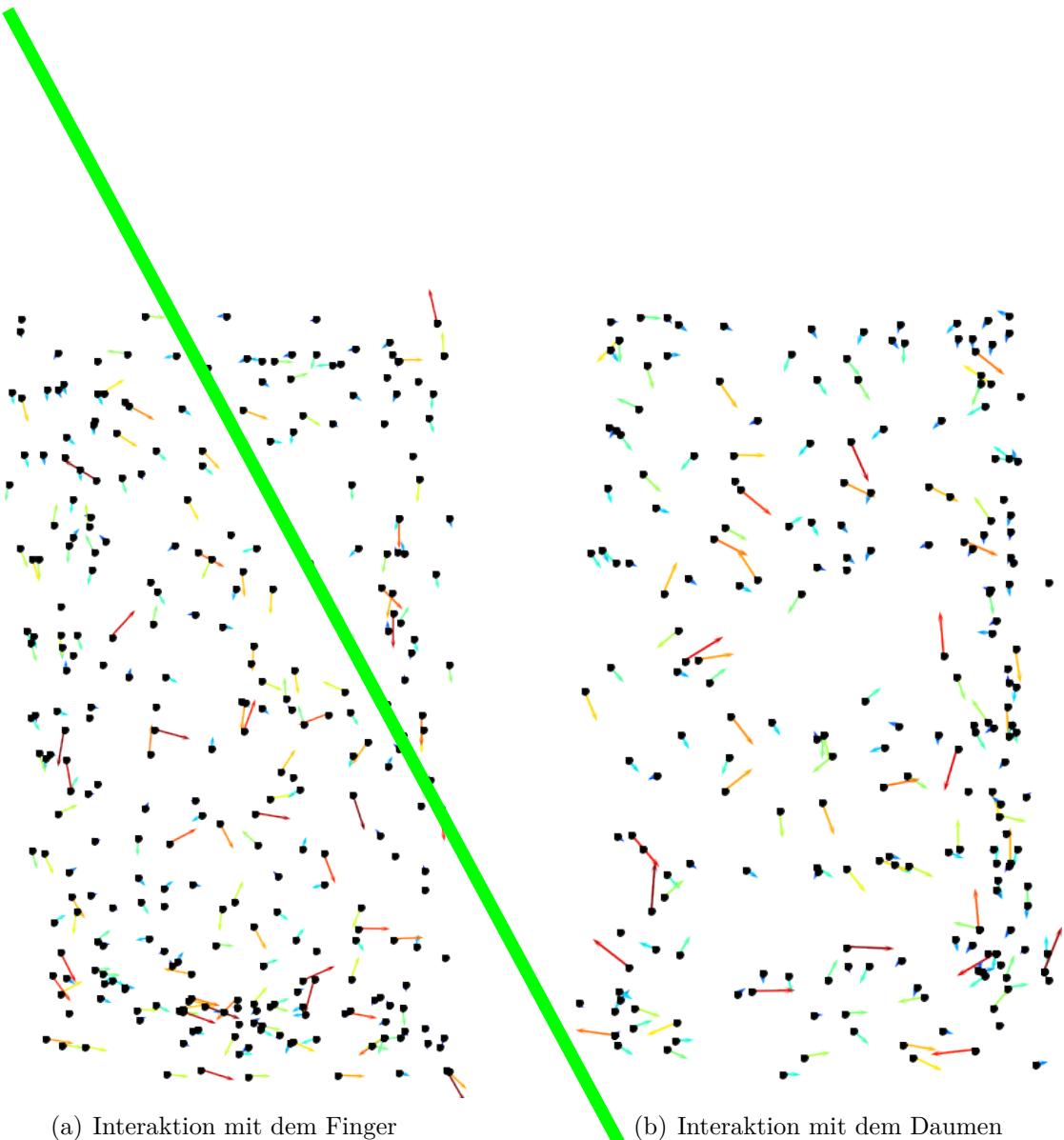


Abbildung 2.4.: Visualisierung der relativen Verschiebungen zwischen der Touch- und Schwebeposition. Die Pfeile inkl. Farbtemperatur deuten die Distanz der Verschiebung an.

Nan, sie ziehen sich ganz genau!

~~Nicht kategorisieren, nur beschreiben~~

2.2.5.1. Personenfragebogen

Allgemeine Personeninformationen, wie Alter und Berufsfeld, ermöglichen es, die jeweilige Testpersonen in Personengruppen einzurichten. Dies ist beispielsweise sinnvoll, sobald sich im Bezug auf die Testergebnisse signifikante Unterschiede zwischen den verschiedenen Personengruppen ergeben. Aus diesem Grund wurde für die Studie ein Personenfragebogen angefertigt, siehe Anhang A.2.

Zusätzlich wird nach der Sehstärke, während der Testsitzung getragene Sehhilfen und den Erfahrungen mit berührungsempfindlichen Bildschirmen gefragt. Durch diese Daten können etwaige Probleme während der Testsitzung erklärt werden, z.B. schlechtes Abschneiden durch ungenügende Sehschärfe oder mangelnder Erfahrung. Damit der Testleiter mögliche Störeinflüsse während des Tests aufnehmen kann, verfügt der Personenfragebogen zusätzlich über ein entsprechendes Textfeld.

2.2.5.2. System Usability Scale

Der System Usability Scale ist eine verbreitete Möglichkeit die Usability eines Systems zu testen. Somit kann diese sowohl allgemein eingeordnet, als auch für jeden Test einzeln verglichen werden. Dies setzt voraus, dass für jeden Test und der dazugehörigen Testvariante ein System Usability Scale angefertigt wird, siehe Anhang A.3.

2.2.5.3. Qualitativer Fragebogen

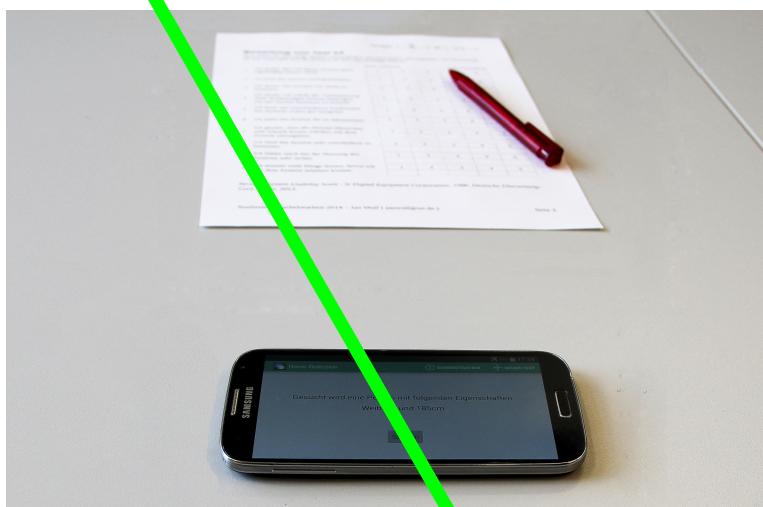
Der qualitative Fragebogen bestehend aus überwiegend frei beantwortbaren Fragen, soll die positiven und negativen Gedanken erfassen, die die Testperson mittels des System Usability Scale nicht erfassen konnte. Auch Verbesserungsvorschläge oder sonstige Anmerkungen der Testperson können hier aufgenommen werden. Zusätzlich werden die Testpersonen für den Ausblick nach möglichen Anwendungsbereichen gefragt, siehe Anhang A.3.

2.3. Prozedur

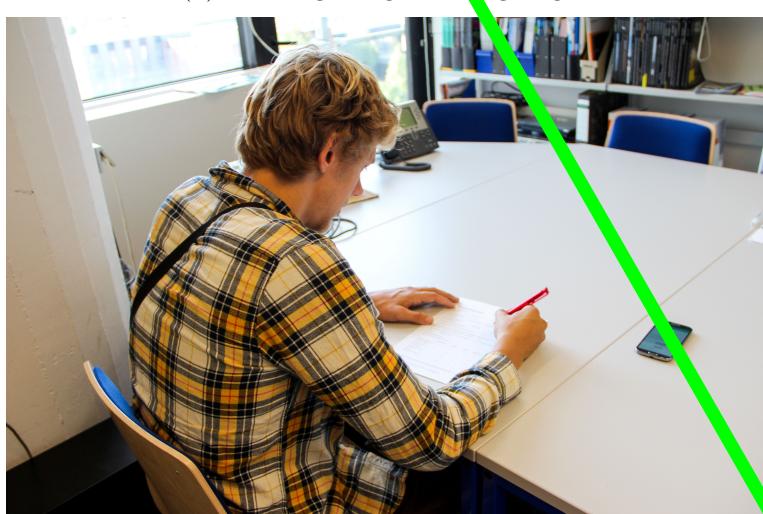
Die Studie findet innerhalb eines ruhigen und hellen Raumes statt, siehe Abbildung 2.5. Der Testleiter sitzt rechts neben der Testperson an einem Tisch mit ausrei-



(a) Räumlichkeiten der Testsitzung



(b) Testumgebung und Fragebögen



(c) Testperson füllt Personenfragebogen aus

Abbildung 2.5.: Räumlichkeiten, Aufbau und Lichtverhältnisse während der Studie

chend Platz. Es soll hiermit sichergestellt werden, dass der Test von der Testperson ungestört und bei guten Lichtverhältnissen durchlaufen werden kann.

Zeit...

Jede Testperson hat ausschließlich freiwillig am Test teilzunehmen. Nach Betreten des Raumes erhält jede Testperson etwas zu trinken und wird zu ihrem Wohlbefinden befragt. Im Anschluss wird die Thematik sehr allgemein und knapp beschrieben. Danach wird der Testperson ein Studieninformationsblatt ausgehändigt, dass u.A. Anhaltspunkte über den allgemeinen Ablauf und die Datenverarbeitung enthält, siehe Anhang A.1.

klickt nach Zeitung

Als nächstes wird ein Personenfragebogen der Testperson überreicht, die diesen direkt und mit ausreichend Zeit auszufüllen hat.

Damit die Testperson mit der Annäherungserkennung vertraut gemacht wird, stellt der Testleiter eine entsprechende Liste zur kurzen Demonstration vor. Die Testperson hat die Möglichkeit mit der Liste bei eingeschalteter Annäherungserkennung zu interagieren.

Das freit

Anschließend wird die Testanwendung konfiguriert. Der Testleiter startet einen neuen Test mit der Angabe der korrekten Testgruppe und trägt dann die in der Testanwendung dargestellte Identifikationsnummer in die vorhergesehenen Felder auf den Fragebögen ein. Die Testumgebung wird nun der Testperson übergeben. Die Testperson wird gebeten, die Testbeschreibung durchzulesen und mögliche Fragen zu äußern. Sobald alle Fragen geklärt sind, beginnt die Testperson mit dem ersten Test.

Hat die Testperson den Test absolviert, wird ihr der System Usability Scale überreicht. Die Testperson hat diesen nun nach „Bauchgefühl“ auszufüllen und kann im Anschluss mit der Wiederholung fortfahren.

Unklar

Wurde auch für die zweite Testvariante der System Usability Scale ausgefüllt, leitet der Testleiter den zweiten Test ein. Die Testperson wird erneut gebeten, die Testbeschreibung durchzulesen und mögliche Fragen anzuzeigen. Danach kann die Testperson mit dem Test beginnen. Die Testanwendung ist so konfiguriert, dass in diesem Test von der Testperson bis zu 26 Schwierigkeitsgrade absolviert werden können. Die Schaltflächengröße wird mit jedem Schwierigkeitsgrad kontinuierlich verringert. Insgesamt wird der Test zwei Mal wiederholt und jede Testvariante ebenfalls mit einem System Usability Scale abgeschlossen.

Wurden alle Tests der Testanwendung vollständig absolviert, befragt der Testleiter die Testperson abschließend gemäß des qualitativen Fragebogens und notiert die gegebenen Antworten für die Testperson auf dem Fragebogen, um eine möglichst

Cann passiv?
Wann passiv?
Passiv = FÖSE

Change?

Apparat!

ausführliche Rückmeldung zu erhalten.

2.4. Design

Die gesamte Testsitzung ist aufgeteilt in zwei Tests. Der erste Test besitzt einen Faktor mit zwei Levels und der zweite Test ist ein 3x26 Design. Beide Test sind within-subjects.

Der erste Test berücksichtigt die unabhängige Variable Annäherungserkennung mit den nominalen Zuständen ausgeschaltet und eingeschaltet. Der erste Test besitzt somit zwei Settings. Die abhängige Variable Zeit beschreibt die Zeitspanne (in ms) zwischen der Listeneinblendung bis zur richtigen Selektion der gesuchten Person. Bei Fehlselectionen wird die Variable nicht zurückgesetzt.

Nur, bis
die korresp.
Sondierende
Schaltfläche
berührt wird

Der zweite Test berücksichtigt zwei unabhängige Variablen, Schaltflächengröße und Annäherungserkennung. Die unabhängige Variable Schaltflächengröße besitzt 26 rationale Zustände, beginnend bei 10mm und endend bei 1.4mm. Die unabhängige Variable Annäherungserkennung besitzt die nominalen Zustände ausgeschaltet, eingeschaltet und eingeschaltet mit Selektionshilfe. Der Test ermittelt für jedes Setting die Werte von vier abhängigen Variablen: Die Gesamtzeit (in ms) gibt die Zeitspanne zwischen der Blaufärbung der Zielschaltfläche bis zur Berührung dieser Schaltfläche an - bei Fehlberührungen wird die Variable nicht zurückgesetzt. Die abhängige Variable Zeit wird dabei vor jedem Neuversuch zurückgesetzt. Die Fehlerzahl gibt die Anzahl der Fehlberührungen an. Diese Variable wird den Wert 5 nicht übersteigen, da nach dem 5ten Fehlversuch der Test vorzeitig beendet und der Wert auf 5 erhöht wird. Bei einem Testabbruch wird zusätzlich die abhängige Variable Abbrüche von 0 auf 1 gesetzt.

Wannum?
Einf doch
Wur fehler
Sollten einige Settings aufgrund eines Testabbruchs überprungen worden sein, so bleiben für diese Settings die abhängigen Variablen Zeit und Gesamtzeit in der Auswertung unberücksichtigt. Die abhängige Variable Fehlerzahl wird in diesem Fall auf den Wert 5 und die abhängige Variable Abbrüche auf den Wert 1 gesetzt.

Um ein Counterbalancing zu erwirken, wurden gemäß des Latin Squares die Tests in mehreren Reihenfolgen durchlaufen. Der erste Test benötigt gemäß der Zustandsanzahl der unabhängigen Variable Annäherungserkennung zwei Reihenfolgen. Der zweite Test muss für ein entsprechendes Counterbalancing aufgrund des zusätzlichen Zustands eingeschaltet mit Selektionshilfe mit drei verschiedenen Reihenfolgen

getestet werden.

Angabe der Ausbalancierung?

A	B	C
B	C	A
C	A	B

(oder so ähnlich)