FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

Mobile News Generation

Erstellen von Interaktionsmechanismen für das mobile Editieren von Videos

DISSERTATION

zur Erlangung des akademischen Grades

Doktor der Sozial- und Wirtschaftswissenschaften

eingereicht von

Mag.rer.soc.oec. Roman Ganhör
Matrikelnummer 9555082

an der Fakultät für Informatik

der Technischen Universität Wien

Betreuung: Assoc. Prof. Dipl.-Ing. Dr.techn. Hilda Tellioğlu

Diese Dissertation haben begutachtet:

Assoc. Prof. Dipl.-Inf. Dr.sc. Marc Langheinrich	Assoc.Prof. Dipl.-Ing. Dr. Klaus Schöffmann

Wien, 9. Oktober 2016

Roman Ganhör

FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

Mobile News Generation

Designing Interfaces and Interaction Mechanisms for Mobile Video Editing

DISSERTATION

submitted in partial fulfillment of the requirements for the degree of

Doktor der Sozial- und Wirtschaftswissenschaften

by

Mag.rer.soc.oec. Roman Ganhör
Registration Number 9555082

to the Faculty of Informatics

at the TU Wien

Advisor: Assoc. Prof. Dipl.-Ing. Dr.techn. Hilda Tellioğlu

The dissertation has been reviewed by:

Assoc. Prof. Dipl.-Inf. Dr.sc. Marc Langheinrich	Assoc.Prof. Dipl.-Ing. Dr. Klaus Schöffmann

Vienna, 9th October, 2016

Roman Ganhör

Technische Universität Wien
A-1040 Wien ▪ Karlsplatz 13 ▪ Tel. +43-1-58801-0 ▪ www.tuwien.ac.at

Erklärung zur Verfassung der Arbeit

Mag.rer.soc.oec. Roman Ganhör
[redacted]

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 9. Oktober 2016

Roman Ganhör

Danksagung

Danke an alle, die geholfen haben, dies zu ermöglichen.

Acknowledgements

I would like to thank Hilda Tellioğlu for supervising this thesis, Marc Langheinrich and Klaus Schöffmann for volunteering to review it. Furthermore, I would like to thank Friedrich Glock for his time to discuss all things necessary and unneccessary, Florian Güldenpfennig for his support, Brett Yarnton for proof-reading, and to Andrea Fekonja for literally kicking me forward.

Kurzfassung

In den letzten Jahren öffneten sich für Benutzerinnen und Benutzer vermehrt die Möglichkeiten, vor allem unter dem Label Web 2.0, das Internet nicht nur passiv zu konsumieren, sondern auch aktiv mitzugestalten. Zum Beispiel motivieren Zeitungen ihre Leserinnen und Leser dazu, Fotos von Ereignissen einzusenden, die der Redaktion sonst nicht vorliegen würden. Während sich das Web 2.0 entwickelte, haben sich auch Mobiltelefone zu multimedialen Multifunktionsgeräten weiterentwickelt. Obwohl die rechenstarken Geräte neben Videoaufnahmen in High Definition inzwischen auch die Bearbeitung der Videos erlauben würden, wird dies jedoch nur selten auch gemacht. Die zentrale Frage dieser Dissertation kreist in der Zusammenführung von mobiler Videobearbeitung und Web 2.0: wie können Interfaces und Interaktionen für gestenbasierte Mobiltelefone gestaltet werden, um mobile Videobearbeitung bestmöglich zu unterstützen, um letztendlich als praktikable Lieferanten für Bewegtbildinhalte verwendet werden zu können. Während sich die aktuelle Forschung vor allem auf automatische oder halbautomatische Filmerstellung konzentriert, untersucht diese Arbeit die manuelle Filmerstellung. Manuelle Filmerstellung ist immer dann wesentlich wenn im künstlerischen Anspruch (dramaturgisch, erzählerisch) an das finale Produkt möglichst wenig Abstriche gemacht werden sollen. Im Rahmen dieser Arbeit wurden drei wesentliche Arbeitsschritte für die manuelle Filmerstellung untersucht, implementiert und evaluiert. Diese Arbeitsschritte sind a) durchsuchen von Sammlungen an Videoclips b) zuschneiden von einzelnen Videoclips c) umordnen von Videoclips in Sammlungen an Videoclips. Die Anforderungen für die vorgeschlagenen Benutzerschnittstellen und Interaktionsmechanismen wurden in einem kollaborativen Prozess erarbeitet, welcher Beobachtung, Literaturanalyse, Interviews und Arbeitsablaufanalyse beinhaltet. Jede Benutzerschnittstelle und Interaktionsmechanismus wurde sowohl mit professionellen Filmeditoren als auch mit normalen Anwenderinnen und Anwendern getestet. Die Tests zeigten, dass beide Gruppen die neuartigen, durchaus komplexen Benutzerschnittstellen und Interaktionsmechanismen schnell verstanden, die professionellen Filmeditoren insgesamt jedoch mehr Interesse zeigten, wenn es um die Anwendung mobilen Videoschnitts ging.

Während der Interviews und der Designphase zeigte sich wiederholt das Fehlen einer klar definierten und anwendbaren Notation für berührungssensitive Benutzerschnittstellen. Dieses Fehlen ist besonders bei der Diskussion von neuartigen Interaktionsmechanismen (wie in dieser Dissertation) bemerkbar, die noch von keiner bestehenden Notation abgedeckt werden. Deswegen wird als Teil dieser Dissertation eine erweiterbare Notation

vorgestellt, die für berührungssensitive Benutzerschnittstellen gedacht ist. Die Notation selbst ist einfach, jedoch auch möglichst eindeutig gehalten, um die Anwendung in möglichst vielen unterschiedlichen Einsatzgebieten (kollaboratives Entwickeln, integrierte Entwicklungsumgebungen, etc.) zu unterstützen. Eine Evaluation in realen Entwicklungsprojekten demonstrierte sowohl das Potenzial und die grundsätzliche Anwendbarkeit für eine solche Notation, jedoch wurde auch mehrere Ansatzpunkte für weitere Entwicklungen aufgezeigt.

Abstract

During the last years we have increasingly seen passive consumers transformed into active producers, often as part of Web 2.0. Newspapers for example turn their readers into producers by motivating them to send in photographs of events that are not covered by a newspaper's journalist or photograph. In the same time span smartphones have become more and more powerful, allowing for high-definition video recording. However, on-site mobile video post-production capability has not yet followed this trend, and on-the-fly video editing is not a common approach among amateur or professional content producers. The central question of this thesis is how novel interface and interaction approaches can support mobile video production applications that are feasible for both amateur and professional video editors. While current research focuses mainly on automated or semi-automated film compilation based on algorithmic decisions, this thesis investigates efficient and effective interaction mechanisms for advanced manual mobile video editing. Manual control over the editing process is crucial for upholding the artistic standards an editor expects of his or her final product. Within the scope of the studies presented here three tasks vital for video editing are examined, implemented and evaluated: browsing media assets, trimming media assets and ordering media assets. The requirements for the proposed interfaces and interaction mechanisms were gathered during a collaborative process that included shadowing, interviewing, workflow analysis and literature research. Each interface and interaction mechanism was evaluated separately with professional video editors and regular user without any background in video editing. The evaluations show that professional video editors were confident about the usefulness and feasibility of the proposals, whereas regular users tend to not wanting to edit their videos manually. However, both groups easily understood the rather complex interaction mechanisms.

Furthermore, during the interviews and design sessions a lack of formal and applicable notations for touch-based interfaces and interaction mechanisms was identified. This absence is especially hindering when discussing design issues that are not platform specific or covered by any platform so far. Therefore, this thesis proposes an extensible sketching notation for mobile gestures. The proposed notation provides a platform-independent basis for the collaborative design and analysis of mobile interactions. During a conducted evaluation with real-world touch-based applications the notation proved being a feasible tool, however, indicated various starting points for further improvements.

„Have you any news?“

The second message transmitted by
Samuel B. Morse, inventor of the telegraph
May 24, 1844, Washington D.C.

Contents

CHAPTER 1

Presentation of the Problem

When a multidisciplinary group of scientists and engineers at the Xerox PARC presented in 1981 their commercial vision of a computer with a graphical user interface (GUI) they had already achieved a great deal. The computer was called Xerox Star and was intended to "produce, retrieve, distribute, and organize documentation, presentations, memos, and reports" (Johnson et al., 1989). And in contrast to then existing computers, the Xerox Star was targeted for business people working in regular offices instead of specialized computer engineers working in dedicated laboratories. While the Xerox Star itself was unsuccessful, the ideas it introduced such as windows, mouse input and direct object manipulation later became mainstream cornerstones through computers such as the Apple Macintosh or the operating system Windows 95 (Myers, 1998).

Today, as smartphones become more powerful these devices allow users to perform tasks originally intended for desktop computers with a standard GUI. However, when transferring a complex application from a desktop computer to a smartphone several issues have to be addressed. Firstly, one of the important differences between a desktop GUI and a mobile device's GUI is the input method. While the desktop GUI relies on a mouse as its favorite pointing device, a smartphone's main input device is a user's fingertip. Besides other differences, a mouse pointer does not clutter a significant portion of the screen, in contrast to a hovering finger. Another critical issue is the limited screen space of mobile devices. Further, even the resolution available on a mobile device increasingly limits a user's ability to see small fonts and layouts and target them with their finger.

The underlying problem we face is that we apply old patterns (and solutions) to new challenges. A traditional interface and interaction design for a desktop GUI is based on windows, icons, menus, exact pointing devices and an assumption of sufficient screen estate to distribute all the necessary components. And however appropriate this approach is for a desktop GUI, it does not anticipate the limitations of small mobile devices or, perhaps worse, it does not respect the capabilities of these types of machines. This thesis

addresses the lack of literature and discussion when implementing complex real-world applications for mobile devices with small screens.

1.1 Motivation

This thesis is motivated by technical and social developments in the last 15 years. On a fundamental level, these changes reflect the transformation of passive consumers into active producers, often subsumed under the term Web 2.0. One popular example of a successful Web 2.0 project is Wikipedia; it is one of the world's largest websites while its content relies mostly on user contributions. As impressive as the mass of knowledge and information accumulated in Wikipedia is, the operator of the website (Wikimedia Foundation) is well aware of the obstacles a user has to overcome to contribute to the world's largest online encyclopedia. „Removing avoidable technical impediments associated with Wikimedia's editing interface" is seen as a pre-condition to attract new contributors (Wikipedia, 2016). This statement indicates the importance of a viable interface and interaction design for the Wikimedia Foundation to build up a relationship with their users. Thus, whenever encouraging users to contribute, it is advisable to think about the needs and requirements of the interface and interaction design in use.

The Internet also changed the relationship between journalists and audiences from a one-way, asymmetric model of communication to a more participatory and collective system, where citizens have the ability to participate in the news production process (Hermida, 2010). Alfred Hermida explains the term of ambient journalism as an awareness system that opens a variety of channels to collect, communicate, share and display news and information from both professional and non-professional sources. In his work on peace journalism, Burkhard Bläsli emphasizes the „support [of] independent media structures instead of the conglomeration of media corporations" to lessen the impact of bigger news corporations (Bläsi, 2004). However, independent media structures do also exist without a reference to corporations. With the rise of blog hosting software and services, non-professionals can publish on the Internet and reach a respectable, sometimes impressive audience (Astell, A., 2008). Be it personal blogs or special interest channels on social media platforms, the passive audience has turned into an active audience by publishing, commenting on and generally discussing articles and other content. Besides non-professionals utilizing the possibilities of the Internet for their needs, newspapers also try to add value to their products by incorporating the capabilities the new media offer. A familiar example is newspapers allowing and encouraging readers to comment or discuss articles online or even to contribute novel content.

nother development of the last years which motivated this thesis is the expansion of mobile broadband connectivity enabled by advanced wireless communications technologies such as UMTS (Universal Mobile Telecommunication System) and LTE (Long-Term Evolution). The additional capabilities of the now widely implemented communication infrastructure triggered new developments on the market for user's mobile devices. High speed data transmission networks and advanced end user mobile devices allow for non-

stationary uploading and downloading of significant quantities of data (Cisco Corporation, 2016). Mobile communication via email or chats, and mobile consumption of websites and other media content have become not just possible but affordable for the masses, making modern smartphones an essential device for our digital life as they are capable of fulfilling many of our digital needs (Chaffey, 2016). Users not only use smartphones for a good part of their daily communication, smartphones have also become reasonable cameras, disrupting the market for traditional small digital pocket cameras (Lee et al., 2016). However, while photo and video creation is common among smartphone users, editing and sharing audiovisual content is normally limited to predefined filters and automatisms.

Thus, the motivation for this thesis is based on following observations. First, smartphones are becoming more powerful and are applied to increasingly complex fields of applications. Second, a powerful data transmission infrastructure allows smartphones to transmit large multimedia files over mobile broadband. Third, media producers such as newspapers incorporate video files to their online presence and motivate their readers to contribute content. This thesis proceeds on the assumption that professional video editing on smartphones can be one of the links between the mentioned observations. Allowing video journalists to produce their videos on their smartphones eliminates the need to transfer the videos from the mobile device to a designated machine like a laptop computer and can quicken the process of online news video generation. However, while a volume of work exists for automated mobile video editing there is little contribution for professional mobile video editing so far. Additionally, when reviewing the related work a lack of literature was found regarding notation for mobile applications that can be used during the design process. To this author's knowledge, the only literature currently available that tackles the topic of touch notation are guidelines from the manufacturers of touched based operating systems or interactions between designers and artists who compile collections of touch gestures. Contributing a first framework for the notations of touch gestures is a second motivation for this thesis. Therefore, this thesis provides usable and viable prototypes for the different steps in professional mobile video editing for online news videos as well as an extensible notation framework for touch-based gestures on mobile devices.

1.2 Problem Statement

Smartphones allow us to carry out even complex tasks in-situ; however, these devices also bear challenges for researcher, designer and user. While stationary desktop computers and semi-stationary laptop computers offer sufficient screen estate, a mechanical keyboard and at least one precise pointing device, smartphones are much more limited. Their screen estate is small, they often lack of a mechanical keyboard and their main pointing device is the user's finger. Due to these obvious differences it is not feasible to simply shrink an interface and interaction concept from desktop or laptop computers to fit on the screen of a smartphone. On contrary, appropriate interface and interaction concepts

for smartphones should be designed with the limitations and capabilities of these devices in mind from the beginning.

When designing and implementing professional video editing software for mobile devices, programmers face several challenges not found on the consumer market. Video editors are trained professionals who have a good understanding of the software they use as well as a clear expectation of the final video they are going to produce with the software. Therefore, professional video editing software on mobile devices should be able to carry out the vital tasks an editor needs, even if the interface size and the interaction mechanism are fundamentally different from desktop or laptop computers. Video editing software for desktop or laptop computers demands a lot of screen space for its interface elements to appear properly. Consequently, an apparent problem is the distribution of all needed interface elements on a small screen. Additionally, when using mobile devices in-situ the surrounding conditions are not predictable. Thus, the interface and interaction design has to take into account the contextual situations where mobile video editing can be carried out. These conditions can be generally shaky, bumpy or rough and the interaction mechanics should still allow precise and clear user control. Another challenge is the raw computing power needed for video processing, as modern video encoding algorithms are complex and CPU-intensive. Even modern smartphones do not necessarily provide all the computing power needed for high-level video stream manipulation. Thus, these limitations need to be considered and overcome when implementing novel interaction designs that manipulate encoded video streams.

Beside the technical issues, another challenge exists which has thus far hardly been addressed by the scientific community. Discussing novel interface and interaction designs is mostly an interdisciplinary process that often involves specialists from various subject areas. While the advantage of such an approach is the expertise the participants can provide, the same participants normally lack basic knowledge and terminology when discussing specific interface or interaction designs. Thus, there is a need for a common notation system that can serve as a base for further research helping scientists, designers and users to estimate the viability of an interaction design before it is implemented.

1.3 Expected Goals

This thesis contributes to the scientific fields of Mobile Multimedia and Human Computer Interaction in two ways. First, an evaluated design proposal for video editing on mobile devices with touch interaction is provided. Second, a proposal for an extensible notation for describing interaction design for touch-based devices is outlined. The expected results of this thesis relate to the field of complex multimedia interfaces, interaction on mobile devices, and to the methods used during the design phase of touch-based interfaces.

The ultimate goal is to provide applicable user interfaces for video editing on mobile phones which allow for on-the-spot editing of news footage. The user interfaces and interaction design presented in this work are aimed at professional or semi-professional

video editors. Hence, rather than automatizing as much as possible, the goal is to provide as much control as possible over the iterative editing process.

To achieve the goal of a feasible mobile video editor, I have gained gain insights into the applicability of proposed user interfaces and associated interaction techniques, collected through:

a) Self-generated application examples, used to illustrate the envisioned style of interaction
b) User studies, focused on evaluating the novel interaction ideas and techniques
c) Expert feedback, commenting on the applied interface

The result will be a set of working prototypes anticipating the various tasks needed when editing videos on mobile phones. These prototypes are examples of real world implementation, allowing for qualitative (and sometimes quantitative) evaluation by and with the target group. The outcome can be used as input and feedback when discussing guidelines for time-based media in the mobile domain.

An additional goal is to provide an extensible notation for discussing and describing interaction designs for touch-based mobile devices. The notation is intended for researchers and designers to explain and notate their ideas in an easy and unambiguous way. The notation can be extended to meet the needs of novel interactions and gestures or to clarify existing interactions and gestures. This extensible notation can support upcoming scientific work when quantifying and evaluating novel interaction designs.

To achieve the goal of an extensible notation for touch-based gestures I propose a notation that is based on existing guidelines and scientific work in this field. The process itself consists of following steps:

a) Literature research to find overlaps, distinctions and areas not covered until now
b) Initial feedback from a substantial amount of alpha users
c) Refinement of the notation
d) Qualitative user studies in real world applications

The result will be a proposal for an extensible notation to depict interaction design unambiguously. The main purpose for the notation is the use during the design phase; however, it can also be utilized for evaluation and automatization purposes.

1.4 Structure of the Work

This work is built on top of five scientific papers written (three as sole and two as first author) by this author and presented at international conferences in the field of

human-computer interaction and mobile multimedia. These conferences were hosted or co-organized by either the Association for Computing Machinery (ACM) or the Institute of Electrical and Electronics Engineers (IEEE). Both administrative bodies maintain a strict policy regarding paper structure and paper length, the exact replication of which here would not benefit the cohesion and accessibility of this document. Rather, this thesis has been organized to provide a broader view on the topic of designing and implementing real world multimedia interfaces and interaction concepts on mobile devices. Chapter 2 (Context) discusses how mobile Internet and user generated content can have an impact on the future of multimedia online news. Chapter 3 (State of the Art) outlines the history and the current state of research in the fields of interface and interaction design, video editing and design tools. Chapter 4 (Methodology) provides an overview and an explanation of the methods used in this thesis. Chapter 5 (Summary of the Scientific Papers) sums up the aforementioned research that form the base of this work. Chapter 6 (Scientific Contribution to the Field) illustrates the contribution to scientific discussion in the respective fields concerned with this research.

CHAPTER 2

Background and Context

This thesis is located in the domain of business informatics specialized in describing and explaining real world workflows connected to information and communication systems (Heinrich et al., 2007). When a specific workflow has been described and explained, its context has been established. Concretely, this context provides information about the setting, the work, the people and the technology that are involved. When we are aware of a specific context, we can consciously design the means and the tools to support and improve a given workflow. In this case, the workflow in question is mobile news production, and more specifically video editing on mobile devices. Further, this thesis is built on the assumption that modern mobile devices are already powerful enough to allow the production of news footage without the need of any additional device. Before we start designing the interfaces and the interaction mechanics for mobile video production we first take a closer look at the main elements that will be involved in a mobile news production workflow such as smartphones (tool), user generated content (artefact) and news making (workflow).

2.1 Smartphones

Whenever a new technology arises it is hard to forecast its impact. While the usefulness of mobile telephone calls for a regular user was easily conceivable, the main purpose of short text messages (SMS, short message service) was seen for telephone providers to carry out service and maintenance tasks. As it turned out SMS became a success story of its own with 350 billion messages sent annually by regular users (The Open University, 2014). As technology has further developed, SMS itself is being superseded by Internet based protocols such as instant messengers and the like.

Other services and gadgets bundled with mobile phones have evolved more slowly for a variety of reasons, in some cases for reasons somewhat analogous to the problems facing mobile video editing. Photo cameras attached to mobile phones (phone cameras) represent

such a case. In the beginning phone cameras had low resolution and worked poorly in low light settings, being technically inferior in comparison to an already existing product. However, with every new mobile phone generation better sensors and better optical parts have been built into mobile phones and as of today, in 2016, phone cameras are believed as a main reason for falling sales figures for designated digital still cameras (Lee et al., 2016). Not only are modern smartphones able to shoot acceptable photos in terms of technical features/numbers, these very same photos can be altered and manipulated on the spot and shared with friends and family. Even though the specifications of phone cameras are criticized by technical reviewers, the quality seems good enough to satisfy user need, as billions of photos and videos show that are shot and shared annually (Marr, 2015).

Perhaps most relevant for this work, the recent expansion of mobile broadband connectivity enabled by advanced wireless data communications technologies such as UMTS (3G) or LTE (4G) now allows a user not only to consume (download) multimedia content but also to produce (upload) multimedia content on/with their mobile device.

2.2 User Generated Content

The widespread availability of phone cameras led to new social phenomena like reader reporters. Reader reporters are ordinary/common people who send photos of an interesting incident or of a current event to a newspaper or magazine for publication. Such photos are typically shot with and sent via a smartphone. This newspaper/audience partnership can benefit on both sides: the newspaper receives photographic material it normally would not be able to access while the reader earns a fee and credits for every photo bought and published by an editorial office.

A user's engagement can vary from making an anonymous comment on an online article to contributing multimedia content. Several studies show how news media houses already try to incorporate their customers into their workflow. For example, in 2009 the US-based American Public Media developed the „Public Insight Network“ where readers, listeners and viewers are encouraged to register. In July 2015 around 230,000 registered users were in contact with 63 TV, radio and print media companies to exchange thoughts on a number of topics regarding user involvement. According to a media representative the platform can give valuable feedback on topics and stories, and, the registered persons often suggest interesting stories on their own worth covering (Kraus, 2014). News corporations and agencies such as CNN and Reuters are experimenting and adopting their workflow to content delivered by users' smartphones, as smartphones are considered a viable device for professional news making when dealing with quick from-the-scene reporting (Väätäjä, 2010; Vihavainen et al., 2011).

The highest level of user involvement is printing articles and showing videos made by the users themself. In Austria, the regional weekly publication "Mein Bezirk" runs a program encouraging readers to turn into writers (Styria Media, 2015). These reader/writers are called Regionauts emphasizing the idea of being regionally rooted and thus, having

valuable insight in regional topics. Another example is the daily Austrian newspaper *Der Standard* which started a denoted online section exclusively for user generated content in 2014 (Burger, 2014).

While the majority of these efforts target written content, Internet based publication can easily merge additional types of media such as pictures, video or audio, which would not be possible with traditional paper print. Furthermore, microblogs, blogs and social media networks allow non-professionals to cover ongoing yevents with multimedia content (Bruns and Highfield, 2015). These real-time or close to real-time video contributions often do not fulfill the same technical, aesthetical or qualitative standards as footage usually seen on television; however, it seems the audience is aware of the production conditions and values the information that is delivered over its packaging.

The term participatory journalism is defined in Bowman and Willis (2003) as "... the act of a citizen, or group of citizens, playing an active role in the process of collecting, reporting, analyzing and disseminating news and information. The intent of this participation is to provide independent, reliable, accurate, wide-ranging and relevant information that a democracy requires."

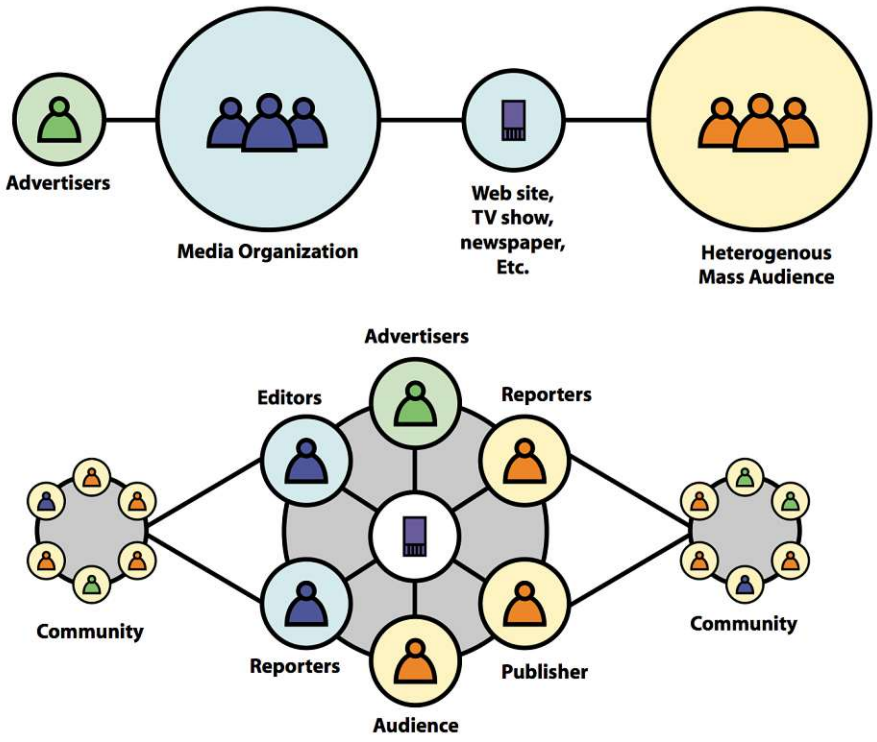


Figure 2.1: Broadcast, top down news vs. Intercast, bottom up news (Bowman and Willis, 2003)

The transformation from traditional broadcasting to participatory intercasting is depicted in Figure 2.1. Broadcasting is a one-to-many connection, with a media organisation

acting as a filter, whereas intercasting dissolves the distinction of senders and receivers.

As Van Every (2004) puts it "the rise of the internet and the increasing availability of low cost means to create digital media have created an environment and an appetite in the audience for meaningful interaction with mass media. Television news is an area that holds great potential for community based programming and can be made to allow the audience a direct role in the production of such programming." However, the rise of the Internet and the introduction of interactive elements not only attract the interest of an active audience. Companies of different sizes tried and still try to incorporate the internet and adapt their workflows to the possibilities the internet offers (Ursu et al., 2008).

From these trends, two target groups emerge for mobile multimedia editing. First, professional reporters for newspapers and online magazines could use this service, as many newspapers and magazines already add multimedia content to their articles. And second, amateur contributors who witness or attend newsworthy events, deliberately or accidently, could expand upon their already increasing contributions. However, before we start designing a pocket sized video editing studio we should take a look into the workflow of current news productions and discuss the differences between multimedia production of television broadcasters and online newspapers.

2.3 Shadowing News Making

Even though the western world has a strong consensus of how a visual news production should look (Lindstedt et al., 2009), newspapers often lack the trained personnel or the specialized equipment for producing multimedia content that achieve the expectations of the audience. However, standardized IT products with specialized software have started to replace expensive dedicated devices for media production (Roeder et al., 2006), allowing "newcomers" such as online newspapers to technically compete with established media competitors such as television broadcasters.

To get a sense of actual work practices and to better understand individuals and groups within traditional television broadcasters and online newspapers, a research study was set up. Recruiting participants for this study was done through personal contacts, with finally eight individuals participating. One participant was silently followed ("shadowed") throughout a working day and after that an in-depth interview was conducted. The insights gained about technology and the technical vocabulary in that particular business set the ground for the remaining seven interviews. The interview took place at the individual participants' workplaces or at places the participants suggested and lasted between 60 and 90 minutes. During the interviews notes were taken by hand. During all interviews, participants were repeatedly encouraged to think aloud and to speak freely.

The first participant is an editor for a nation-wide TV news show broadcasted weekdays in the early evening. The news-room and the technical equipment of the TV station represent the state-of-the-art for producing TV content in high definition (HD). The

focus of the day-long shadowing is on everyday work the editor has to do, documenting how and in what ways various tools are used. Furthermore, it is of interest how the workload is prioritized to fulfill strict deadlines (broadcast time) and what tools the editor is missing. It is suggest that some of the findings are specific to this observed context. However, during the additional interviews cross-references to this observation ("shadowing") were made to strengthen or weaken the findings.

The observation includes the work needed for one TV report from its very beginning (genesis of the idea) to its broadcast as two minute segment. The editor (from now on referred to as Ed) was shadowed over all phases involved in a typical media production life cycle: pre-production, production, post-production and distribution of the report (Hardman, 2005).

2.3.1 Pre-Production: finding a topic

The observation took place on a regular Wednesday in June and started at 8:30 in the morning in a conference room located at the TV station's headquarters. At this time two people were present: the chief editor and the chief editor's assistant. Both engaged in small talk and discussed the weather situation as it had been hot for the last few days and the forecast predicted an even hotter day for that day. In the background a large flat television set showed random news. During the next 30 minutes 12 editors joined in and sat around the table. Half of the group had laptops; the rest were equipped with paper and pencil. A few editors had brought a newspaper of the day or a weekly magazine. Both were meant as research tools for potential topics.

The meeting was officially started at 9:00 by the chief editor. Everyone had read a newspaper, a magazine or had listened to radio on their way to the conference room to gather ideas for the early-evening TV show. The chief editor asked the attending editors for any upcoming ideas. The editors spoke out loud and the chief editor makes notes with paper and pencil. A repeated comment of the chief editor was "if you cannot tell the idea of the story in one sentence, it is probably the wrong story." The assistant of the chief editor also took notes on his laptop. The proposed ideas were checked for footage needed and footage potentially available in-house via a media-database. It is more likely that a story is approved and consequently aired when appropriate footage is available in house for free. Another point that has to be considered when selecting a topic is the overall program structure, which is a balanced mixture of politics and economics on one hand and society reports and sports on the other.

At 9:41 the chief editor summed up all reports that were likely to be aired. Likely, because it was not assured at that time that the needed footage can be delivered in time. Every editor was assigned a story. Every editor wrote down his or her story on paper, even the editors who had laptops. The more interesting stories (according to the chief editor) received a 2 minute slot for the final program, whereas the less interesting stories (according to the chief editor) received a 90 second slot. Overall, the program lasted 20 minutes and consisted of 10 reports.

2.3.2 Production: gathering the information

The editor to be shadowed (Ed) was assigned a topic about tax issues. During the first 15 minutes Ed looked for a free place in a shared work environment to settle down. A few minutes past ten Ed asked the chief editor on the mobile telephone (even though there were landline telephones available on each desk) about the direction the storyline should go. Ed wondered whether the story should be more political (which political party is for it and who is against it) or more descriptive (the cost and the impact for the citizens). They both agreed on a more descriptive approach.

Ed opened a new Microsoft Word document to collect all the necessary data during the day. He researched the names of the institute and the researchers who published the article about the costs and savings of the new tax law. A telephone call to the research institute was made and the lead scientist was asked for a short TV interview. The lead scientist had no time for an interview but a co-author was present at the institute and was able to give a TV interview.

At 10:45 a production demand document is filed. The TV station maintains agreements with several service agencies and each of these service agencies coordinate several local independent freelancing camera and production teams. Ed sent an email to a service agency which is located close to the researcher's institute. The content of the email was partly copied from the word document and consisted of: 1) who is to be interviewed; 2) interview location; 3) interview topic; 4) interview questions; 5) additional contact information. After a few minutes the service team was called by telephone to confirm the order. This call was done in addition to the email to assure that the order had arrived and was carried out properly. During the telephone call the details for the interview were repeated to clarify potential misunderstandings (time, place, contact, reason, etc.). Ed called the chief editor to report on the current status and the chief editor suggested to contact the in-house graphics department to include some graphics and statistics in the report. A few minutes later the service agency confirmed the order and sent the contact data for the camera team in charge. Ed refined the briefing he sent to the service agency and sent it to the camera team and the service agency. A few minutes later the mail was confirmed independently by the camera team and the service agency.

Around 11:15 Ed started gathering statistics on the topic. After a quick search on the internet and dissatisfying results Ed tried another approach. He called the public statistics institute and asked for appropriate statistics. He passed on details about his email address and contact information while the statistics institute assured him they would call him back when they found the charts and numbers in question.

The camera team calls Ed at 11:38 to report that they were already on the way and would arrive within the next hour at the researcher's institute. Ed started a short oral briefing and what "quotes" are expected to be in the interview. Ed advised the camera team to make sure that the researcher mentioned some keywords such as a special name (e.g. of a politician) or a specific number (e.g. total costs of the planned changes). This extra briefing was intended to level the different experiences and skills of the different

camera teams, assuring a viable overall-quality of reports.

At 11:46 Ed wrote an informal email to the in-house footage archive asking for appropriate footage for the report. The archive would later store the selected footage at a predetermined computer folder. This folder would also be used by the camera team when they transferred their footage via Internet.

A few minutes before 12:00, the statistics institute called and informed Ed that they had already sent an email containing the information (numbers and figures) he had asked for. During the lunch break between 12:00 and 13:00, Ed met with other editors. Each of them were already running their own reports by that time. They talked about their reports and they gave tips to each other where additional information could be found and the like.

After lunch Ed checked national and international press agencies for news for additional insights. He also did further research on how the numbers from the statistics institute could be arranged to produce an informative and understandable chart. He realized, however, that figures for another country would give the chart more meaning. Since the national statistics institute had only the numbers for one country Ed had to find the additional numbers by himself. Around half past one Ed found a newspaper article online with a suitable chart. He copied the URL from the browser to the Word document (the document is still nameless and unsaved thus far). The numbers, in contrast, are not added to the Word document. The numbers are written on his sketchpad with a pencil. Even though he copied them wrong twice, he did not write them in the digital document.

Having decided on the story's basic outline and numbers, Ed started to write the off-text for the TV report in the Word document. In between, he did further online research on the topic. The off-text for the TV-report was copied to a specialized media program. This is the first time Ed used a tool tailored for media production.

As the chart was going to be animated, graphics specialists had to be involved. Ed opened an online PDF-form that is used as a working order for the in-house graphics department. All known and needed information is typed into the form, i.e. name of orderer, name of TV report, date, headline of needed chart, numbers and figures for the chart. All the numbers and figures were copied back from the sketchpad to a digital format. A few minutes before 15:00, the camera team reported that it had finished the interview but that it had trouble transferring the video files over the Internet. Ed advised them to try again and printed out the working order for the graphics department. He brought the working order to the graphics artist personally. Ed arrived at the graphics artist's office a few minutes past three. He explained the basic idea of the chart, what information it should convey and what the animation of the chart should look like. A short discussion between Ed and the graphics artist started.

A few minutes later Ed was back at the shared space and kept on refining the off-text (again in the Word document) for the TV report for the next one and a half hours. Ed regularly copied the off-text from Word to a production tool which calculated the time needed when reading a written text out loud. Considering direct quotes from the

interview, Ed calculated the time for the off-text to be around 1 minute. Therefore Ed honed sentences and tried to shorten them wherever possible. Ed is only interrupted once, at 16:00, when the camera team reported their successful attempt to transfer the video files.

2.3.3 Post-Production: bringing everything together

The post production facility includes multiple video cutting systems and audio recording booths. One video cutting system (including a trained video editor) and an audio recording booth was booked for 16:00. At 16:15 the post production facility was still occupied by another editor (for another television show the same day). The senior duty editor stopped by, read the off-text and gave feedback on it. With 38 minutes delay the post-production facility was ready, a "longer than normal overrun" as Ed explained. After a cordial welcome Ed and the video editor skimmed through the footage made and sent by the camera team and the footage provided by the in-house archive to which the video editor had been granted access by the system. While browsing the footage at double its normal speed, Ed briefly explained the intended report. Simultaneously they tried to select concise direct quotes from the researcher for the report.

Twenty minutes later, around 17:00 a first raw cut was discussed between Ed and the video editor. While watching the raw cut Ed read out loud the compiled off-text to check the "spoken length". Via intercom Ed got in contact with the graphic editor who made the charts for the report. All graphic workstations are directly connected with the post production facilities. Thus, modifications made by the motion graphic editor can be viewed in real time on the monitors in the post production facility.

Ed asked the senior duty editor for 15 more seconds air-time as he cannot squeeze all the information in the scheduled time slot. The senior duty editor agreed to the additional time and adapted the entire airing schedule. Ed went to the voice-recording booth and dubbed the latest version of the report. The voice recording needed several takes. Some sentences were corrected for grammatical accuracy; some sentences were repeated as specific words were unintelligible. During the voice recording, the video editor acted as a final reviewer for the off-text.

About 15 minutes to 18:00, which was air-time, the senior duty editor sent the latest airing schedule via email. According to this schedule the slot for Ed's report was 1 min 48 sec. A few minutes later a first rough cut of the complete TV report was available. A discussion started right away whether or not a specific word in the off-text could be understood. A minute later the discussion was settled and it was decided that the word in question was understandable. At 17:53 the senior duty editor took a look at the final cut of the report and approved it.

In the adjacent studio the host of the TV show prepared her opening sentences (lead-in) and read through the announcement of the reports that would be aired in a few minutes time. At 18:00 the TV show goes "live" and a few minutes later the report (we followed so far) was aired.

2.4 Interviews and Discussion

The process just described provides a first approximation of the workflow of professional video news production. However, other players in the news-market do not have their expertise in producing video footage. This is of relevance as the Internet opens up an opportunity for different players in the news-market, all with different backgrounds. Television broadcaster can run websites with comprehensive textual in-depth descriptions and newspaper can augment their textual articles with short video reports. We are especially interested in the latter, how newspapers adopt to the challenges of (thus far) unfamiliar approaches to disseminating news. Therefore, eight interviews were conducted with reporters employed in the area of video news making. Four people were in the area of "traditional" television news broadcasting, whereas four people were employed by online newspapers. The interviews with the people in television news broadcasting should strengthen and complement the input gathered from the shadowing process, eventually establishing a baseline of the current status of (traditional) video news production. In contrast, the people employed by online newspapers should contribute their approach as video is relatively new for a medium that is known and reputed for textual content.

Interview 1 to Interview 4 were conducted with people working at traditional television broadcasters. It turned out that the workflow described in the section above was described in quite a similar fashion by all participants. While the basic workflow was always almost described the same the number of persons involved varied. Even though the number of persons varied, the entirety of responsibilities and skills was equivalent throughout. Each production consisted of a director (senior duty editor), editor, director of photography (film team), sound engineer, narrator, and cutter (video editor). A consistent skill set and variations in the number of persons indicates that one person can have more than one responsibility during a production. Nevertheless, whenever a person was assigned to one or more responsibilities, the person had at least a minimum of formal training to be able to adequately anticipate and fulfill their tasks. For example the director had formal training in research and inquiry, the director of photography had formal training in operating video cameras and sometimes film aesthetics, the narrator has formal speech training and so on.

Interview 5 to Interview 8 were conducted with people working as video journalists at online newspapers. What stood out most, was, all interviewees worked alone and not as a team. Every newspaper video journalist was not only responsible for the whole production process, he or she was also in charge of carrying out all necessary working steps by themselves. Consequently, and almost inevitably, all interviewees lacked formal training in one area or another. Most had formal training as journalists, but no training as video cutters; a few had technical knowledge in video editing, but lacked formal training as journalists.

When discussing the differences of television broadcast and online newspaper journalists, we experienced a gap in how the topic of a video report is approached. While video reports are the core business for television broadcasters, it is a relatively new genre for

online newspapers. Online newspapers sometimes see this as an advantage to start with a relatively small and streamlined production process and concentrate on fast availability of the content or content that is not suitable for television broadcasters. Fast availability gives online newspapers a head start when competing with television broadcasters, and since the production process is so streamlined there is little organizational overhead. And newspaper journalists seem to prefer stories with a more local than national reference. This local focus was explained as follows: first, it is a market niche not well addressed by television companies so far, and second, due to the flexibility newspaper journalists can spontaneously cover stories when they are appear, such as a fire or car accident. Discussing the drawbacks of the workflow in newspaper video journalism several issues were brought up. One of the most obvious is the lack of feedback loops with other persons. While in the workflow described in detail above the editor (Ed) cooperated with several specialists such as the video editor and the graphics artist, the online newspaper video journalists had no similar interactions during production. Another drawback that is especially interesting for this work is the lack of mobile applications online newspaper video journalists were able to draw on to further improve on the competitive advantage they had in delivering close to real-time video reports.

CHAPTER 3

State of the Art

Interface and interaction design connect a human's ability to create with a machine's ability to provide suitable information and interaction techniques. While applications for the first generation of touch-based smartphones were rather straightforward, the complexity has evolved ever since. As touch-based smartphones have become increasingly powerful, these devices have become increasingly capable of accomplishing even complex tasks. At the same time, research in the area of human-computer interaction has attempted to find novel ways to deal with this rising complexity, with varying levels of success. When analyzing successful applications it often turns out that one factor of their success is a limited and plain interface and interaction design.

While customers and users intuitively assess an application, researchers and designers need more replicable measures to evaluate a given interface and interaction design. For desktop designs a range of such measuring tools exist, such as GOMS (Goals, Objectives, Methods, Selection rules) and KLM (Keystroke-Level Model). For mobile devices with small screens and touch interaction an adaption of KLM has even been introduced, TLM (Touch-Level Model), to fit the circumstances of this interaction technique (Rice and Lartigue, 2014). However, all of these methods are intended to be viable when predicting the usefulness and execution time of interface/interaction design patterns which already exist.

This section gives an overview of the evolution of desktop and mobile interface and interaction design in general and of video editing in particular. Furthermore, it introduces tools and methods to assess and evaluate a given interface and interaction design, with a focus on mobile devices.

3.1 Interface and Interaction Design

Early computers were operated by trained personnel only. Whoever interacted with such a device, or machine, had to undergo some sort of formal training first. The user interfaces at that time consisted of switches, punched cards, cables and light bulbs. These user interfaces were hard to interact with and barely interactive. Making computers easier to operate, making them more interactive and more accessible for untrained users is still a task researchers, scientists and designer are working on. However, as technology progressed, new means for input and output emerge, opening new areas for further research.

3.1.1 Desktop Interfaces

Vannevar Bush published an article in the magazine Atlantic Monthly in 1945 with the title "As we may think". In his article Bush imagined a machine ("Memex") in the shape of an ordinary desk, capable of storing and retrieving information by means of microfiche. The interaction takes place by via a keyboard, buttons, levers and a stylus. The output was projected on translucent screens for reading. Even though this machine was never built, it employed concepts which are still valid, including exploiting user management of known artefacts such as a desk or a stylus (Bush, 1945).

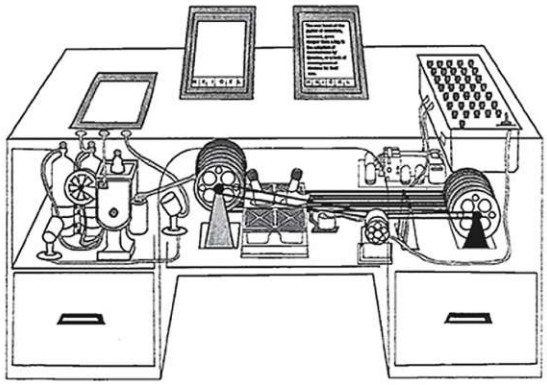


Figure 3.1: Memex, envisioned by Vannevar Bush in his Article "As We May Think"

Ivan Sutherland (1963) built Sketchpad allowing users to manipulate objects directly on the screen with a stylus. Even though this system was mainly for demonstration purposes, it introduced pioneering ideas such as the direct object manipulation and laid the foundations for object oriented programming (Saffer, 2010).

Douglas Engelbart (1968) gave a 90 minute long demonstration of his work what became known as "The Mother of All Demos". In his demonstration Engelbart introduced several design and interaction paradigms to a wider audience that influenced the next decades of human computer interaction, such as the computer mouse. And eventually the mouse

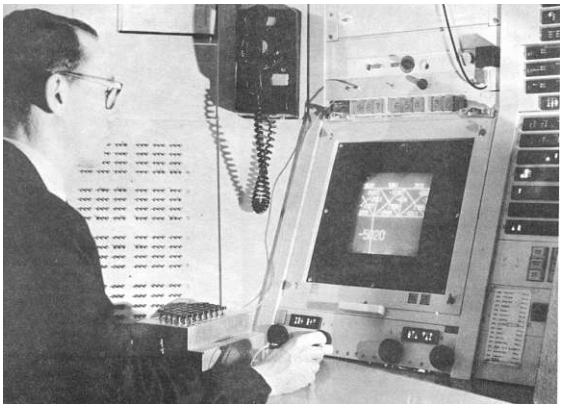


Figure 3.2: Sketchpad presents new interface and interaction paradigms and introduces the foundations for object-oriented programming.

allowed the implementation of new interaction techniques like point and click (Barnes, 1997).

Figure 3.3: Engelbart demonstrating mouse and interactive text editing

Xerox Parc, a research outlet of Xerox Corp., consolidated these ideas with many others into key standard aspects available in virtually every modern computer system. Highlights include the graphical user interface (GUI), WYSIWYG text editors and the WIMP paradigm denoting windows, icons, menus, pointing devices (Johnson et al., 1989).

3.1.2 Mobile Interfaces

One of the members at Xerox Parc was Alan Kay, who envisioned and theoretically described in detail a portable computer, mainly for learning purposes, in 1968 (Saffer, 2010). In his article “A Personal Computer for Children of All Ages” Kay explains Dynabook, a small and transportable device capable of displaying text in the quality of book pages, playing multimedia files and downloading media files to its own file storage

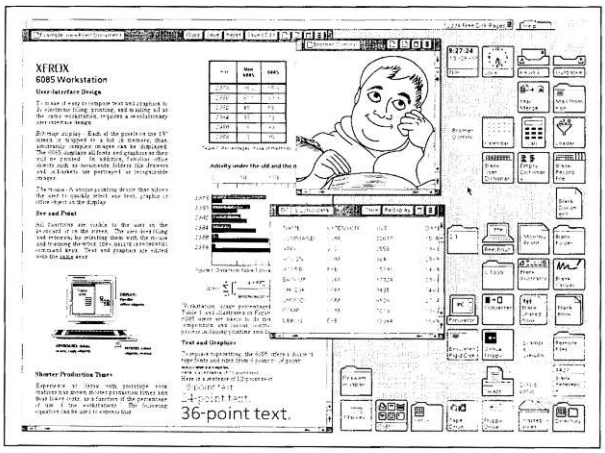


Figure 3.4: Xerox Star demonstrating the WIMP paradigm

(Kay, 1972). As mentioned, the Dynabook was intended to be a learning device, however, the author mentioned the possible practical use far beyond being "just" a useful tool for learning.

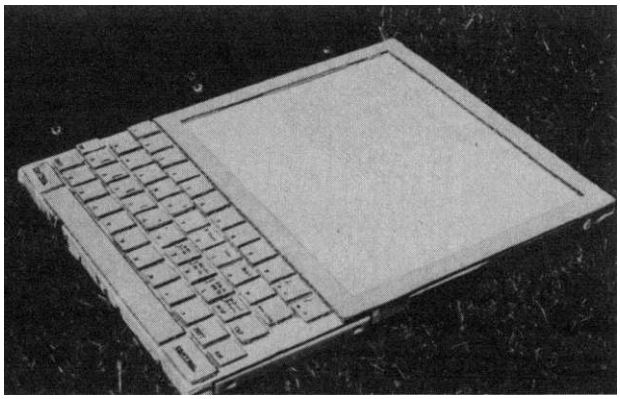


Figure 3.5: Dynabook by Kay (1972)

In 1987 the then CEO of Apple Computer, John Scully, described a visionary device which nowadays can be seen as an imagined ancestor of tablet computers. The device was shown in a number of corporate videos, made by LucasFilm, showcasing the features of the envisioned device, called the Knowledge Navigator (Richards, 2008). The Knowledge Navigator had a touch sensitive display and a software agent capable of understanding complex commands given in natural language. The videos also exhibited collaborative networking features allowing two distant users to easily work on one common document (Dubberly, H., 2007). The Knowledge Navigator was never built as the technology was not available at that time.

In the early 1990s a new computer category was rising, the handheld PC or the personal

Figure 3.6: Knowledge Navigator: the concept of tablet computing

digital assistant (PDA). Tandy Corporation presented the Tandy Zoomer in 1992 and Apple Computer introduced their Newton platform with the MessagePad in 1993 (Lewis, 1993). Both, the Zoomer and the MessagePad, had no mechanical keyboard and used a stylus and handwriting recognition software as their primary means of input. Even though the MessagePad handwriting recognition software was still flawed, Apple decided to present its product at the Boston MacWorld tradeshow (Butter and Pogue, 2002). Although neither device was a major commercial success, both were pioneers of the PDA-like devices which emerged in the years to come.

Figure 3.7: Message Pad: stylus as the main interaction device

One notable company in the area of PDA was Palm, Inc. Founded in 1992 (and acquired by U.S. Robotics in 1995) as a software provider for Tandy's Zoomer, the company released their own first device in 1996, the Palm Pilot (Wiggins, 2004). The Palm Pilot utilized an improved, yet simplified, handwriting recognition software known as Graffiti. Instead of urging the user to mimic the exact outline of a character Graffiti used a simplified one-stroke outline for a character avoiding cross-strokes and the like. Furthermore, a user wrote each letter on top of the previous letter instead of the natural left-to-right writing style. Due to the write-atop approach, a user would not run out of space when writing longer words. The end of a word, or the beginning of a new word, was indicated with a one-stroke gesture indicating a space.

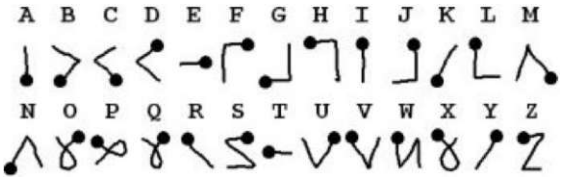


Figure 3.8: Graffiti: a simplified alphabet for handwriting recognition

Graffiti is a revealing example how an interaction mechanism can aim for better user experience by adapting to new input modalities while trying to avoid constraints. Instead of relying on the users to mimic sentence and letter layout, the Graffiti software shifted some of the burden from the user to the recognition and prediction algorithms. Even though a user had to learn the Graffiti alphabet, the "new" alphabet was close enough to the common alphabet to allow for a manageable learning curve. The commercial success of the devices is evidence that users are willing to take a bit of training into account when the tradeoff is a more fluent workflow.

In 1993 IBM presented the Simon Personal Communicator, which can now be seen as the predecessor of modern smartphones (Woyke, 2014). Simon had a touchscreen, calendar, address book and email. It had a physical extension slot for cartridges, allowing it to run additional programs (apps) such as a music player or a navigation program (Sager, 2012). A few years later, in 1996, Nokia introduced the Communicator 9000, which featured a web browser and a hardware keyboard. Finally, it now seems that Apple's iPhone, presented in 2007, paved the way for software emulated keyboards instead of hardware keyboards and, more generally, fully gesture-based interaction via finger tracking.

Figure 3.9: Simon (left, IBM, 1993) and Nokia Communicator (right, Nokia, 1996)

This brief overview just scratches the surface, however, it does give an idea of how concepts, prototypes and real-world implementation over the last 80 or so years have repeatedly revolutionized the way we use and interact with electronic computing devices. Modern smartphones somewhat resemble the ideas of Vannevar Bush and Alan Kay, even though nowadays they are used for much more than information retrieval and learning. It stands to reason that the success of today's smartphones is partly due to their versatility,

allowing programmers worldwide to utilize built in sensors (e.g. camera, GPS) and distribute finished applications effortlessly via app-stores. And the more powerful the devices are, the more they are capable of being deployed for even complex tasks. Despite users adaptations to new interaction paradigms, a human's input channels (eyes, ears, skin, etc.) and output channels (fingers, facial expressions, etc.) are set and limited. Thus, it is not only technological progress that determines the success of an invention; it is for a substantial part the steady leveling of technological possibilities and human capabilities.

3.2 Video Editing

Video editing is the process of transferring existing, unordered raw footage into a single film that conveys an intended story in the form of a feature film, a news report or more generally, a piece of art. In the beginning of filmmaking editing was done by physically cutting and cohering film material, later electronically with analogous videotapes and today mostly digitally in the form of video files. As technology advances the possibilities do, and with the growing possibilities the complexity grows. To tame the complexity, research has basically followed two strands, automatization and novel interface and interaction techniques. In the following I give a brief overview on the evolution of video editing in general and focus on contemporary research in the area of interface and interaction design in the area of mobile video editing in particular.

Linear video editing describes the event of direct copying the source tape to a destination tape to modify and rearrange video material. The task of editing in a linear manner consists of three steps. First, position the source tape at the position the copy should start. Second, position the destination tape at the position where the source content should be copied to. Third, start the source tape (player) and the destination tape (recorder) synchronously. As the source material is not stored or cached during the copy process the synchronicity of player and recorder is ensured by a third party apparatus.

Non-linear editing, in contrast, does not copy the material directly. Instead the sequences on the source material are marked and arranged without the need to copy them instantaneously Browne (1998). When all sequences are marked and rearranged the non-linear editing system ingests the needed source material and computes the final output. After the final output is computed it can be played out on tape, any other storage medium or directly broadcast. Today, virtually all video editing systems are non-linear video editing systems.

In 1969 the television company CBS together with the hard disk manufacturer Memorex Corporation developed a first step what would later become non-linear video editing. The machine built (CMX 600) was not only capable of holding of 30 minutes of black and white video material in mediocre quality, it also allowed the operator to locate any frame within a second. The CMX 600 could not produce the final master tape itself, however. Instead it produced a paper tape with the time codes encoded on it (Rubin, 2000). This

makes the CMX 600 a hybrid: a non-linear user interface for the video editing process and a linear, but automated, copying process.

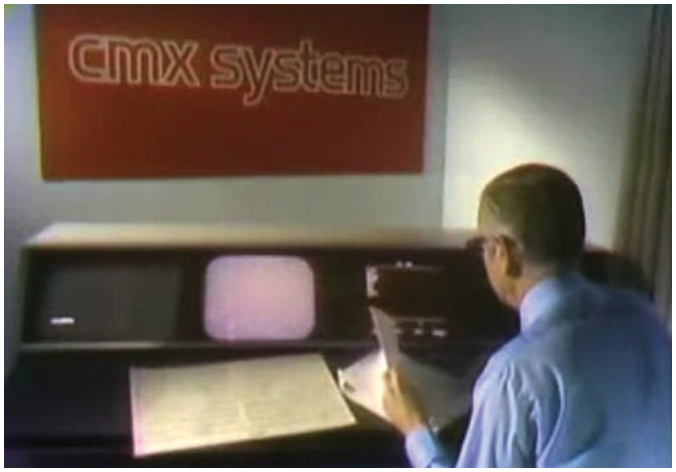


Figure 3.10: The CMX 600 was the first non-linear video editing system allowing the operator to access any frame in a 30 minute video sequence within 1 second.

A next step in the development of non-linear editing systems was the EditDroid presented in 1984 by Lucasfilm (Kirsner, 2008). The EditDroid system used Laserdiscs to simulate a non-linear video editing experience and adopted the physical layout of existing film editing machines for its GUI. This included the video tracks running from right to left (Goldman, 2007). Even though EditDroid was not a commercial success, it influenced up and coming video editing systems. Modern video editing software for desktop computers still utilize similar visualization techniques as the EditDroid, however, the video tracks are now running from left to right (Goldman, 2007).

Figure 3.11: EditDroid was developed by LucasFilm and used Laserdiscs to allow fast random access to any frame in a video sequence.

The company Avid replaced the Laserdiscs with hard disk drives and introduced the Avid/1 system at the National Association of Broadcasters (NAB) convention in 1988 (Warner, 1988; Luff, 2007). The first systems of Avid were based on the Apple MacIntosh platform. Subsequently however their systems also became available on Microsoft Windows platforms.

Like the CMX 600, the Avid/1 could not edit and manipulate the raw video material in its highest quality available. Instead the Avid/1 ingested the raw video footage (online quality) and made a smaller proxy (offline quality) for editing. Thus, the editors worked with the offline video material which offered lower quality, however, it was technically easier to handle in terms of processing power and storage capacity. Soon after the first demonstration at NAB, Avid was looking for advanced hardware compression and decompression to improve the offline video quality of their system (Buck, 2011).

3.2.1 Browsing in a Video Clip

A specific characteristic of multimedia documents, and hence video documents, is their time dependent nature. This is in contrast to static documents such as text or images. Therefore browsing time dependent documents requires different approaches to browsing than time independent documents (Hürst et al., 2004). This raises two challenges: 1) how to represent time dependent documents in a practicable way and 2) how to browse continuous data. To answer the first challenge, representation or visualization of a time dependent document can be done via content-compression or time-compression.

Content-compression extracts particular segments of the data stream, converting the dynamic data stream into static representations of it, often called keyframes. The advantage of keyframes is that all static interaction browsing mechanics can be applied, the downside is the loss of dynamic information contained in the data stream. All the information that is between two keyframes is lost for the user and cannot be taken into consideration when interacting with the system while browsing the data stream.

Time-compression allows the user to access any position in the data stream and replay any portion of the data stream at any speed. A common interaction technique for setting the position in the data stream is a scrollbar element. The advantage of the slider is its similarity with a standard GUI slider. The downside of a scrollbar is its inability to perform well on larger documents. While the scrollbar gives a good overview on small documents and allows reasonable and precise positioning, a user quickly loses this precision in larger documents as the scrollbar gets very small (if it scales with the document size). Thus, a small step with the scrollbar results in a big leap in the document.

However, interaction elements for time-based media can also merge both representation techniques, resulting in a blended interaction mechanism. Such a mixed approach is utilized by the online video platform youtube.com.

In the following various video browsing approaches are introduced. Some of them extend existing paradigms such as the different implementations of the scrollbar while others

Figure 3.12: YouTube shows a small thumbnail next to the cursor when scrubbing the timeline.

are unique and novel in their approach and their interaction mechanics. However, all of them can be assigned to techniques of data compression, time compression or both.

Tapestry
One approach to expanding a still keyframe is to summarize a video in a multiscale image. Such a multiscale image is continuous in both, the time the spatial domain (Barnes et al., 2010). The calculated multiscale image has no hard borders between discrete moments (Figure 3.13). The authors named this technique tapestries since their continuous nature is akin to medieval tapestries and the like.

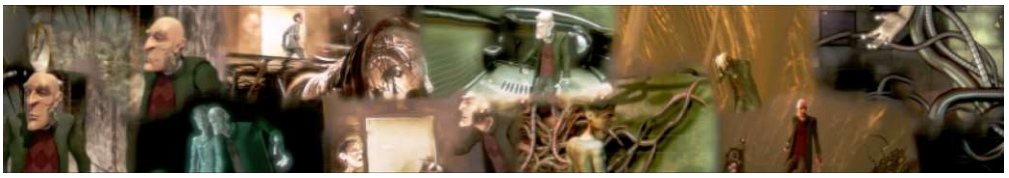


Figure 3.13: Tapestry paints one artistic picture of a video sequence (Barnes et al., 2010).

A user can zoom in and out of a given tapestry, thereby switching to another zoom level. In a user study the authors conducted, the participants found the proposed interface aesthetically pleasing and were able to carry out the assigned browsing tasks sufficiently. However, the participants were all familiar with the videos that were used for evaluation. It is not clear how users would interact with tapestry when confronted with footage unknown to them. Furthermore, a tapestry is compiled automatically to stitch a complete and aesthetically appealing overview. Again, it is not clear how well such an automatized compilation picks the 'right' moments of a video and how they fit a user's expectation.

Video Summagator
Video Summagator aims to summarize a single sequence instead of the whole video by using a volume-based interface. As video summagator allows real time navigation, the application facilitates quick content identification (Nguyen et al., 2012).

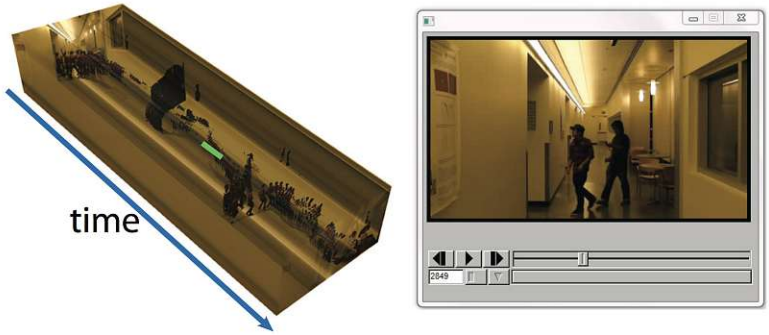


Figure 3.14: Video summagator enables a user to look into the video cube and navigate by selecting the corresponding area in the 3D summarization (Nguyen et al., 2012)

A video Summagator generates, based on the single frames in a sequence, a three dimensional representation ("cube") that can be rotated by a user in all three dimensions to reveal parts of the video that are occluded (Figure 3.14). Furthermore a scrollbar allows for browsing in the video along the time-axis. While this playful approach has obvious advantages such as a quick scene overview, it is hard to navigate on a frame by frame basis.

Swifter

Swifter addresses shortcomings especially prevalent when browsing online videos with limited bandwidth (Matejka et al., 2013). Instead of a single thumbnail Swifter utilizes the available screen size to display an array of thumbnails in the shape of a thumbnail grid (Figure 3.15).

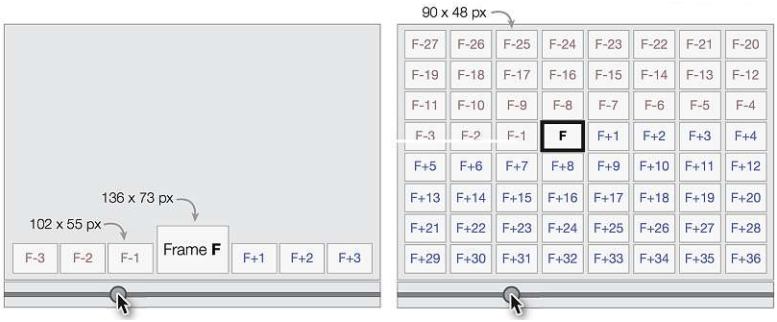


Figure 3.15: Swifter utilizes the whole screen for the preview thumbnails during navigation (Matejka et al., 2013).

Depending on the length of video stream (number of frames) and the width of the associated scrollbar (in pixels) every movement of the scrollbar's playhead is mapped to a calculated number of thumbnails. For example, a video stream has 9000 frames and the scrollbar has a width of 1000 pixels. Moving the playhead 1 pixel jumps 9 frames in the video stream. In this case, a Swifter-page has 9 thumbnails. However, when a

movie has a length of 1 hour the video stream consists of 108,000 frames. Then the movement of the playhead of 1 pixel is associated with 108 frames in the video stream. Consequently, the Swifter-page holds 108 much smaller thumbnails, as the screen estate does not change. This makes browsing more cumbersome, as it becomes progressively harder to identify the contents of ever smaller thumbnails as video length increases.

Swifter was initially proposed as a browsing interface for online content on desktop computers. However, online video content is also available for mobile devices and mobile devices are in general even more affected by slow and unreliable internet connections than desktops. When implementing thumbnails on mobile devices it is absolutely crucial to make sure that the content of the thumbnails is identifiable for a user. A study in 2010 investigated how the size of thumbnails correlate with the ability to identify their content correctly. Therefore (Hürst et. al. 2010) compare two groups of thumbnails, static thumbnails (showing one image typically from the mid of a video clip) and dynamic thumbnails (showing more images from a video clip in a loop). Figure 3.16 depicts thumbnail sizes in millimeter (mm) instead of pixel as in the original paper. This is a more appropriate measure as different devices have different pixel density and thus different sizes for the same "pixel length". However, the study clearly indicates that it is easier for a user to determine the content of a dynamic thumbnail than of a static thumbnail.

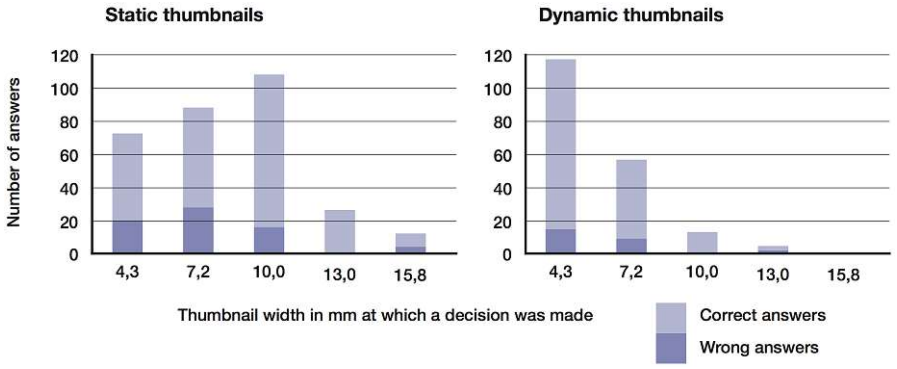


Figure 3.16: Perception of static and dynamic thumbnails. The depicted figures are adapted from Hürst et al. (2010) to display millimeter instead of pixels.

Regarding Swifter works better the more thumbnails are available and the screen estate of mobile devices are limited the thumbnails should be dynamic to maximize content identification rate. Panopticon does exactly this and adds another interesting feature.

Panopticon

Panopticon takes the term of dynamic thumbnails a step further. Not only does the content of the thumbnail move, the thumbnail itself moves (Jackson and Olivier, 2012). A thumbnail starts in the upper left corner, depicting the beginning of the video content. On its way to the lower right corner it moves through the screen in a row-like manner (Figure 3.17). Depending on the thumbnails relative position, on the screen the thumbnail

depicts the corresponding content of the video stream.

Figure 3.17: Panopticon fills the whole screen with thumbnails and moves both, the content of the thumbnail and the thumbnails itself (Jackson and Olivier, 2012).

A sequence at the beginning of the video stream will be closer to the top of the screen, a sequence at the end of the video stream closer to the bottom of the stream. This allows a user to quickly find a given sequence within the whole video stream by spotting it spatially on the screen as the whole content is laid out in a consistent spatio-temporal way.

To render a standard video stream in a Panopticon like visualization is a processing intensive task, however, after the transformation is done the replay of a Panopticon visualization is not different to a replay of a standard video stream. Therefore, the authors of Panopticon see a good area of application on resource constrained (read-only) technologies such as DVD or Blu-Ray players. Other fitting applications are reviewing large spans of video information such as 'life-logged' data from wearable cameras, surveillance footage, or providing video editors when cutting their initial raw cut from non-edited footage. However, it is open to discussion how this interface can be adapted for mobile devices with small screens, as the small screen estate limits the number of thumbnails and it is not clear for now how the usability scales with the number of thumbnails.

HiStory

HiStory is an interface approach that, like Panopticon, fills the whole screen with thumbnails and employs the thumbnails as navigation items. Again, like Panopticon, the first thumbnail in the top left corner represent the beginning of the video stream whereas the thumbnail in the bottom right represent the end of the video stream (Hürst and Darzentas, 2012). The application HiStory extract all thumbnails evenly from the video stream, i.e. every 3 minutes, and display them on the screen. A bar on the right indicates the zoom level. Even though it looks like a scrollbar it is not "scrollable", as it has only visualization purposes. When a user presses a thumbnail the interface zooms into the video stream, depicting a smaller time interval around the selected thumbnail.

This zooming in is also represented by the zoom level indicator on the right side of the screen (Figure 3.18).

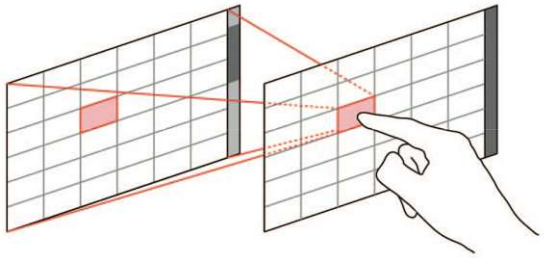


Figure 3.18: HiStory: At the beginning the thumbnails display the whole video sequence, the first frame at the top left position and the last frame at the bottom right position. A user can „zoom“ into the video sequence by pressing thumbnails (Hürst and Darzentas, 2012).

The more a user zooms into the video stream, the more details of the video are visible. Zooming all the way in, a user would end up with the finest granularity, depicting every single frame of the video. Zooming out is initiated with the device's back button. The authors of HiStory describe the interaction technique as similar to changing the scale of a map.

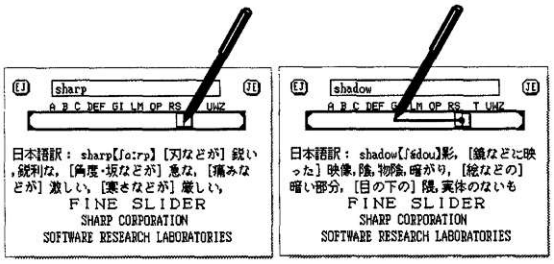
In an initial user study the paper's authors state that the participants understood the interface and could handle it quite well. However, all evaluation tasks were Known-Item-Searches (KIS), where the participants were asked to find a particular thumbnail in a video stream that was known to them beforehand. The results would be presumably different if the video stream was not known beforehand.

3.2.2 Browsing with Sliders

Sliders are an omnipresent tool for browsing documents, either for static documents that exceed the screen size or for time-based documents such as audio or video files. In the following various approaches are presented that extend and enhance basic slider widgets.

Fine Slider
An early idea on improving the scrollbar for browsing in longer documents was Fine Slider, utilizing the rubber band metaphor(Masui et al., 1995). The user can use the knob in the scrollbar by clicking on it and moving it around within the boundaries of the scrollbar widget and the document would depict the document at the indicated position (Figure 3.19). Furthermore, the user can click close next to the knob and move away from the knob as if stretching an imaginary rubber band. The length and direction of the rubber band indicates the speed and the direction the user browses in the document. The longer the rubber band is, the faster the user browses through the document. Even though the Fine Slider was presented and tested with long textual lists, the authors of the

Fine Slider hint to other applications where the rubber band metaphor can potentially be useful.



(a) Dragging the knob directly (b) Pulling the knob by a rubber-band

Figure 3.19: Fine Slider utilizes a rubber band metaphor (Masui et al., 1995).

As depicted, given the current position of the knob the imaginary rubber band can be longer to the left side than it can be to the right side. The closer the knob comes to the beginning or end of the scrollbar the shorter the rubber band can be, and consequently, the length of the rubber band becomes zero. Even though this is not considered a problem for a specific application it still is an inconsistency.

Ortho Zoom Scroller

The Ortho Zoom Scroller adds a second dimension to the traditional scrollbar element, allowing it to browse at various scales (Appert and Fekete, 2006). The scrollbar element behaves like a traditional scrollbar when the pointing device (mouse cursor) moves within the scrollbar's bounds, however, when dragging the mouse cursor outside of the scrollbar's bounds, it continuously changes the granularity of the slider movements. Figure 3.20 depicts the principle. On the left (a) the scrollbar itself fills the whole screen, indicating that the user has an overview over all elements in the list. In the middle (b) the mouse cursor is a little bit away from the scrollbar element and now the scrollbar fills about half of the scrollbar element, indicating that the user already zoomed into the list giving a more detailed view of the list. On the right (c) the mouse cursor is even further away from the scrollbar resulting in an even more detailed view of the list.

The zoom factor of the Ortho Zoom Scroller depends on the distance between scrollbar element and mouse cursor: the further the distance, the higher the zoom level. The design combines two dimensions in one interaction element, panning on the y-axis and zooming on the x-axis. While the original design of the Ortho Zoom Scroller was intended for textual lists it can be imagined that an adapted design could work for video browsing as well. An important feature for video browsing is fast forward/backward as well as slow forward/backward, which can be translated to zooming in and zooming out.

However, zooming out is not a needed feature for the Ortho Zoom Scroller as it always starts with a complete overview. Thus, a useful adaptation for video browsing must provide browsing at normal speed, fast forward browsing (zooming in) and slow browsing

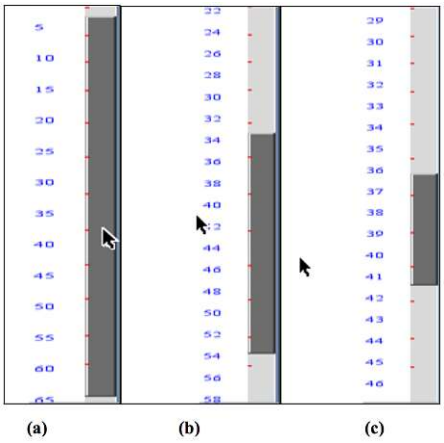


Figure 3.20: Ortho Zoom Scroller: (a) low precision (b) medium precision (c) high precision (Appert and Fekete, 2006)

(zooming out). Another important consideration is how to avoid too much occlusion when this interaction technique is transferred to a finger-based touch interface.

Elastic Panning

Researchers adapted the concept of the orthogonal slider for video browsing by extending the regular 1-dimensional video browsing scrollbar with a second dimension (Ramos and Balakrishnan, 2003). Elastic panning implemented 2-dimensional video browsing for stylus-based mobile devices. While the first dimension works like a regular video browsing scrollbar, the second dimension is similar to rubber band browsing (e.g. Fine Slider), however, elastic panning detaches the different browsing styles spatially. The Elastic panning (2nd dimension) is invoked by clicking directly on the window that holds the document's content, whereby the initial clicking position is set. After the initial position is set, the relative position of the initial position to the current cursor position determines the browsing speed (Figure 3.21).

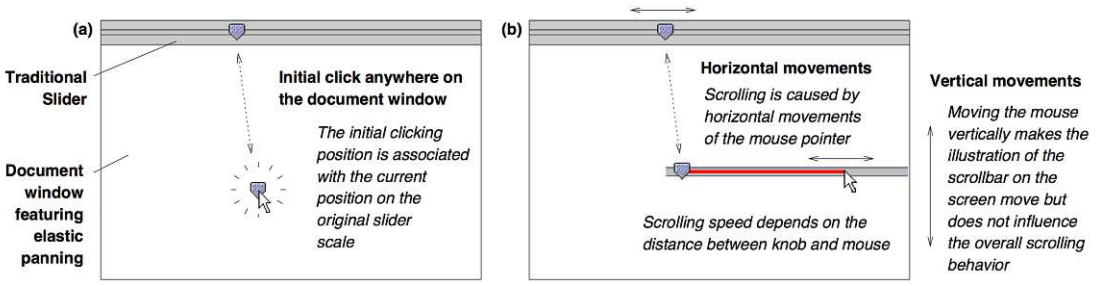


Figure 3.21: Elastic Panning (Hürst et al., 2004)

This approach lessens some of the Fine Slider's downsides. As Elastic panning spatially separates the regular browsing interaction element (scrollbar) from the special browsing

interaction element (Elastic panning) it is less error-prone to unintended false user input. Furthermore, Elastic panning's initial clicking position does not depend on the scrollbar's knob position and does not move after initiation. This allows a user to adjust and maintain a fixed browsing speed independent from any position in the video stream.

However, even with the initial clicking position placed at the border of the screen, the element occludes parts of the (underlying) content. Even though Elastic panning allows the vertical relocation of the initial clicking point through vertical mouse movement, the occlusion stays. Elastic panning was implemented for a stylus interface; consequently, finger-based interaction would occlude even more of the content than the interaction element itself.

Mobile Zoom Slider

Mobile Zoom Slider follows a similar approach as the Elastic panning interface (Hürst et al., 2007). However, instead of a rubber band like browsing tool, the Mobile Zoom Slider initiates virtual scrollbars with different granularity depending in the initial clicking position (Figure 3.22, left). The further away from the regular scrollbar the virtual scrollbar is initiated, the finer the granularity is. This allows users to easily adapt the browsing speed to their needs: slow browsing to find a specific frame in the video and fast browsing to reach any position in the video quickly. The authors of Mobile Zoom Slider refer to this as position-based browsing. The downside of position-based browsing is that the scrollbar should be longer than the whole screen when browsing at lower speed, which makes browsing through the whole video at a lower speed impractical.

To allow a constant browsing speed Mobile Zoom Slider offers two dedicated areas referred to as speed borders. The speed borders are located at the very left and the very right on the display and allow browsing at different speeds without the need to move the cursor. When pointing the cursor in the speed borders a user can browse through a video at a constant speed (Figure 3.22, right).

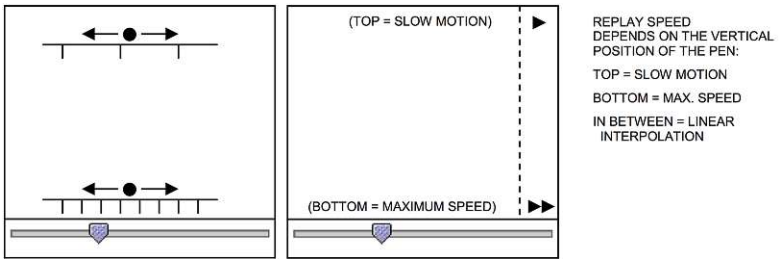


Figure 3.22: Mobile Zoom Slider: scrolling at the bottom allows for fast browing in a video sequence while scrolling at the top allows for precise browsing in a video sequence (Hürst et al., 2007).

In the study the authors of Mobile Zoom Slider conducted, users understood the novel interaction techniques and were able to use them sufficiently even though a few users needed some extra initial training. One observation was that the majority of the participants stuck to the position based browsing and, in general, ignored the speed based

browsing. It is not clear if the preference for the position based browsing was due to the tasks the participants were given, the initial training they received or some other motive. Participants also commented that the browsing elements of the Mobile Zoom Slider are hidden and only become visible after a user's input. Besides the controversial discussion about the visibility, users objected to the lack of immediate feedback about the browsing speed and suggested better feedback implementation. Finally, the participants argued that interaction elements that are very close to the border of the screen are hard to reach with the stylus. This last issue is probably even more likely to be true when using a finger-based interaction mechanism.

LG Patent

The electronics manufacturer LG Electronics patented an interaction methods for controlling the playback speed of video streams on mobile devices (Seon-Hwi, 2013). The direction of the the playback, forward or reverse, as well as the browsing speed is determined by a single moving gesture (Figure 3.23). Depending on the point of initiation and the point of release of an successive drag the proposed interaction method calculates speed and direction for the video.

If the point of release is right of the point of initiation the video plays forward and vice versa. The longer the distance between these two points are the faster the playback speed will be. A small popup serves as a feedback mechanism depicting the playback speed, however, the playback actually starts after the gesture is fully completed and the finger has lifted the surface.

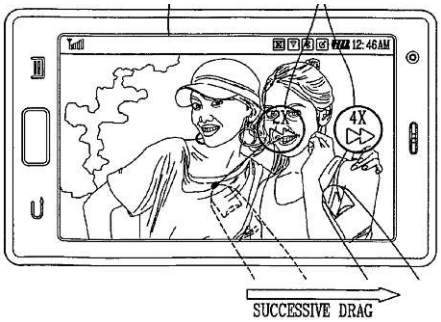


Figure 3.23: LG Patent: browsing is similiar to elastic panning (Seon-Hwi, 2013).

While this proposed method allows to adjust different browsing speeds and directions with a single gesture it does not support fine grained frame by frame browsing.

ZSlider

Another approach for adding a supplementary dimension was proposed by Ramos and Balakrishnan (2005). In their work the authors add the z-dimension to control the browsing speed, thus utilizing a pressure sensitive stylus and a stylus tablet. The authors put significant thought into avoiding potential shortcomings of their initial design, and built in interaction mechanisms to overcome them. One of the more obvious problems would be to maintain a given pressure on the stylus over a longer period of time to keep

the browsing speed stable. This problem was addressed by the authors with a locking mechanism. However, the conducted user study revealed that the participants wanted more and finer control over the locking mechanism.

Summary
A wide variety of different approaches to video browsing exist. Many of them extend the scrollbar element (slider) with an additional dimension. The advantage of these extended sliders (time compression) are their similarity with standard GUI slider, however, all presented implementations have their own drawbacks: some were intended for bigger screens or stylus interaction, and when implemented on a finger-based touch interface the interaction elements are too small or the finger occludes too much of the content. Furthermore, all presented interfaces are optimized for either normal browsing and fast browsing, or normal browsing and slow browsing. None are designed for all browsing modes: normal, fast and slow.

The second widely used approach is condensing the content of a video stream, often referred to as keyframes. Virtually every online video on demand platform uses keyframes, at least as an optional representation method. One can argue that even DVD covers are an artistic representation of the film and thus, a sort of analogue keyframe. However, keyframes have an inherent drawback, as a single keyframe does not reveal much of the content of the associated video it represents. And the longer a video is, the harder it becomes to pick the 'right' keyframe. Therefore, for a user a keyframe can be a reminder when a video is known to him or her, or, an eye-catcher when the video is unknown. Even though researchers have proposed several approaches to overcome this limitation and proliferate keyframes with more information, static keyframes often feel "odd" when utilized for dynamic browsing.

Thus, when designing a (time-compressing) browsing interface for touch-based mobile devices, the following points should be taken into consideration: avoid occlusion through fingers, allow different browsing speeds (slow, normal, fast), provide a locking mechanism for a given browsing speed, and give immediate feedback when changing browsing speed.

3.2.3 Browsing in a Collection of Assets

Beside browsing and trimming single video assets, another important aspect of video editing is bringing these single video snippets into the right order, thus allowing the final video to tell a story. Ordering media assets on a mobile device holds special challenges for interface and interaction design due to the limitations of such devices, such as small screen estate or imprecise touch-based interaction.

Thus, a designer has to balance a good overview over all media assets through small thumbnails of videos or images while using thumbnails that are big enough to provide sufficient visual information. Furthermore, an effective interaction technique is needed to scroll through all assets quickly while still allowing for precise ordering of these assets. In the following we revise the history and current interface and interaction design and discuss their applicability for mobile devices.

A straight-forward approach for browsing a collection of thumbnails is depicting all thumbnails at the same size in grid-like layout. A user can zoom out (making the thumbnails smaller) to get an overview and zoom in (making the thumbnail bigger) to scrutinize the thumbnails on a more detailed level. Repeated zooming can cause a feeling of being lost in bigger collections of thumbnails as such interfaces normally provide little to no information about the global structures of the collection itself. This puts a cognitive load on users who must mentally assimilate the overall structure of the information space and their location within it. Various techniques exist for visualizing information and helping a user to perceive this information. Three basic and widespread techniques are Zooming, Overview+Detail, and Focus+Context. We summarize them in the following, based on a review article by Cockburn et al. (2008).

Zooming
Zooming separates the overview and the detailed view temporally, and consequently a user can only see an overview or the details at a time. A zooming interaction is usually initiated by a user and is then carried out by a computer program. Normally the time span for the interaction is shorter than the time span the program needs for zooming, resulting in a brief period of automatic zooming. During this brief time span the user just observes the zooming animation. The animation establishes the relationship between the pre-zoom state and the post-zoom state. The better (more natural) an animation is, the easier it is for a user to perceive and interpret the transition from one state to the other.

The effects of different timing and animations can vividly be tested with online maps from different vendors as they generally implement divergent zooming strategies when considering the scroll wheel at the computer mouse as the preferred input device. When a user stops scrolling on the scroll wheel, one vendor has a slight scroll out animation lasting for half a second, while another vendor immediately stops scrolling. The first animation type can leave a user with the sensation of "smooth" control over the animation whereas the second animation type can leave a user with the impression of "stricter" control. However, preferences for animation types vary from user to user. In any case a zooming interface should provide a user with contextual information such as the zoom level (Figure 3.24). Furthermore, the zooming interaction should be accompanied with a panning interaction allowing the user to move on a given surface.

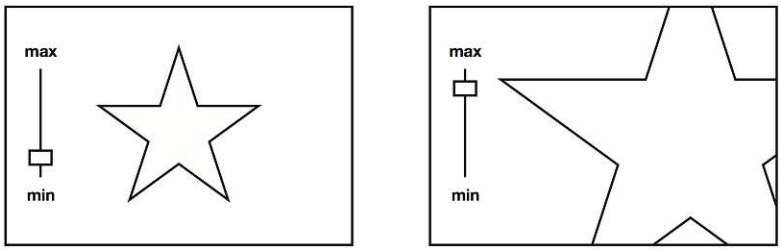


Figure 3.24: Left: The zoom level is low and the object can be seen as a whole. Right: The zoom level is high and only a small fraction of the object can be seen.

Overview+Detail
Overview+Detail spatially separates the overview and the detail. The overview not only can give away information about the zoom level it also transports additional contextual information helping a user to perceive the "bigger picture". In Figure 3.25 the upper right corner holds an overview while the main part of Figure 3.25 depicts a detailed view. The overview indicates the location of the detailed view by means of an arrow, a square or the like.

Figure 3.25: Overview+Detail: the main screen depicts the detials while the bottom right corner conveys an overview (Cockburn et al., 2009).

Depending on the implementation the overview can grow and lessen with the detailed view or stay the same. I.e., when zooming on street or district level within London the overview depicts whole London, whereas, when zooming at city level the overview depicts whole England, and so on.

An early implementation of Overview+Context is the video game Defender (Figure 3.26). It has on its top center an overview of the in-game world while the main screen holds the player's spacecraft, the planet's surface, the aliens and all other details. As the player moves his or her spacecraft left or right a) the main screen scrolls in the respective direction and b) the overview scroll in the respective direction as well. This keeps the game character centered in the overview, allowing for the best possible lookout in both direction at any time.

Focus+Context
While Zooming and Overview+Detail separate the states detail (zoomed in) and overview (zoomed out) temporarily or spatially, Focus+Context depicts both states within one visualization. To depict both focus and context on a single screen, some portion of the information is distorted. Furnas (1986) explains this approach with an instructive caricature named the "New Yorker's View of the United States". That said caricature shows the inner city of Manhattan/New York in great detail street by street. The neighbouring New Jersey is reduced to a colored patch and the rest of the U.S. is even more simplified to a few principal landmarks such as bigger cities or natural landmarks. That view allows a resident to find the closest mailbox as well as to guess the distance to the closest city or to a given natural landmark. Furnas asks in his paper if an analogous

Figure 3.26: The computer game „Defender“ has one of the first occurences of the overview+detail display.

view could be useful for computer interfaces as his fundamental motivation is balancing the local detail and the global context. Before applying and implementing the concept, Furnas clarified this concept formally. In his approach, he virtually assigns a number to an item which indicates how important this item is for a user when executing a specific task. This number is named “Degree of Interest” (DOI) and is composed of two components, the A Priori importance (API), which is determined by the current task and the Distance between an item and the item a user focusses on. This has two consequences, first, the DOI of an item can change during a task, and second, the DOI of an item can change from task to task. Figure 3.27 shows visualizations where items with lower importance are distorted (Cohen and Brodlie, 2004) with transitions between different DOI's being non-continuous or continuous. It should be noted that the continuous visualization Polyfocal Display has more than one point of interest.

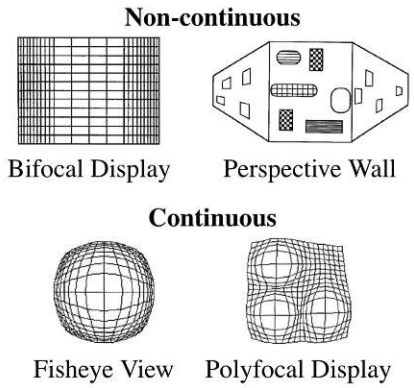


Figure 3.27: Degree of Interest can change non-continuously or continuously (Cohen and Brodlie, 2004)

However, Spence and Apperley (1982), who described the Bifocal Display, pointed to the importance of understanding the context of a task before implementing a visualization based on the idea of focus+context.

Summary
In their compilation of the various visualization techniques, Cockburn et al. (2009) investigate the performance of each technique and its preferred area of application. The low level evaluations target solely on acquisition time (finding an item) as time is easy to measure and easy to compare. The high level evaluations, in contrast, focus on the task domains. In their extensive discussion the authors conclude that every interaction technique has its advantages and disadvantages, and that for specific contexts an interaction technique will be suited better or worse. Furthermore, the authors conclude that combining the interaction techniques with domain specific information can lead to better user experience. The various interaction techniques can be categorized into overview+detail, zooming and focus+context.

Overview+Detail is an easy to understand interface and puts a low cognitive load on the user to perceive and interpret the shown information. On the other hand, the screen utilization is not optimal which is especially true for devices with an already small screen estate. Zooming can be a powerful interface when done right, however, it puts a cognitive load on a user and additional interface elements (zoom level indicator) are needed to lower the cognitive load. Focus+Context is suitable when a rapid overview over the data is needed for orientation and just a small subset of the information is relevant at any time. However, the distortion of a great portion of the data can easily give a wrong impression of the spatial distribution and thus, the amount of data in different segments of the visualization.

The presented interfaces and interaction techniques are quite fundamental and often part of existing interfaces and applications for manually sorting and ordering of multimedia assets, even though they are not always noticeable as such. As mentioned above, when designing a novel interface or application it is important to understand the context and (natural) limitations of a task in order to gain advantage of it building interfaces which better support a user.

3.2.4 Video Editing on Mobile Devices

Although the first films were static shots without any cinematic techniques (such as camera movements), filmmakers experimented and incorporated new modes of storytelling over time (Davenport et al., 1991). One such improvement for the medium film was the introduction of editing the film after the scenes were shot. Editing is the process of selecting, arranging, shortening and leaving out scenes. The considerations for editing can be manifold, like maintaining continuity, pacing of the story line or following the flow of a given script.

Browsing in a video asset and browsing in a collection of video assets are two repetitive but important tasks in video editing. Combining these tasks (interfaces) in one application

is challenging, as switching between the interfaces (tasks) can influence the interaction design of a single interface. Additionally, users can value aesthetics and enjoyment over efficiency (Norman, 2013), favouring a less productive tool due to its pleasing visual appearance. And finally, experienced users may have expectations how an interface for a particular task should appear. Thus, in order to produce a viable interface or application, a designer has to carefully balance learning curve, expectation, efficiency and aesthetics. One of the first, to the best of the author's knowledge, fully functional video editors was discussed by Jokela et al. (2007). Figure 3.28 outlines the navigational structure of the application.

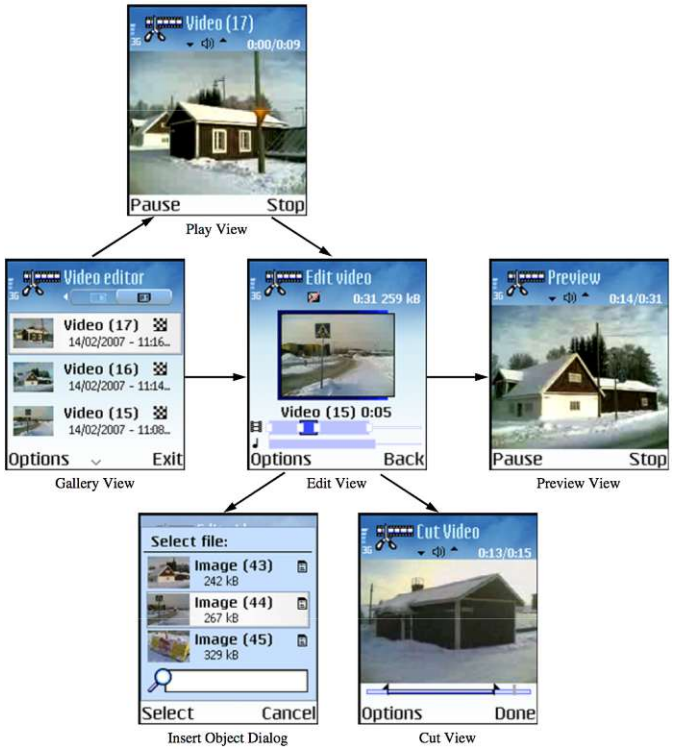


Figure 3.28: The mobile video editor by Jokela et al. (2007) resembles most of the features known from desktop video editing.

When comparing the Mobile Video Editor with a desktop video editing suite such as Final Cut Pro X (Figure 3.29) the obvious main difference is that the mobile version splits single tasks into single interfaces while the desktop version allows access to all tasks from one interface. Even though modern desktop monitors have relatively generous screen estate, each section often has a scrollbar as the amount of information exceeds the available space to display it at once.

Thus, it seems that there is never enough screen estate. However, when creating a special purpose application a designer or researcher can make assumptions about the

Figure 3.29: Final Cut Pro X, an professional video editing suite for desktop computer/laptop computer.

situations the application is used within. To get a better understanding of the differences between a general purpose video editing suite and a video editing suite for news reports several interviews with professional film video editors and video news reporters were conducted. The differences regarding the interface and interaction design are presented in the following:

Length of the video: While documentaries, series episodes or feature films (general film content) typically last somewhere between 30 and 90 minutes, a news report has a length between 1 and 3 minutes. Consequently, the raw footage used for a news report is only a fraction compared to the raw footage used for general film content.

Time of production: News reports are normally filmed and edited within one day while the production time for general film content can span for months or even years. Thus, while long term storage and retrieval of raw footage is not a primary concern during news report production, it is important store raw footage in a dedicated place for possible later retrieval (e.g. to find a reference for a future report). This should be taken care of during a production cycle.

Number of people involved: General film content involves a scriptwriter, a director, a director of photography, a cutter, sound engineer, actors, actresses and many more. News reporters work with a small team of up to three persons and are often on their own. This reduces the communication overhead and more importantly, it reduces the number of devices involved. As the idea is to

Technical Standards: Television broadcasters categorize their visual content into three distinct quality standards, high quality, mainstream and news, allowing news reports to be less restrictive concerning technical and aesthetic standards (Knör and Driesnack, 2009). Even though this work is not referring to the television news reports as its target

group, the lower standards for news reports make it easier to determine more important and less important features for the intended mobile video editing interface.

Gaining a better understanding of the context can help to design artefacts that better fulfill their purpose. Additionally, it is "important to understand the high level goals [and] profession-related goals and needs" (Väätäjä, 2010). Väätäjä and Männistö (2010) did a study with students of (video) journalism on creating mobile news and they summarized the mentioned video editing capabilities as follows: cut a video, merge two clips and add a title. Although this seems a bit oversimplified, it contains the key idea that a special purpose application can be designed by focusing on a small set of crucial functions. Thus, contextual information is not only important to understand the users and the work they do, it is also vital in identifying which specific features constrain function and should be removed to better address the core functionality of an interface and interaction design.

3.2.5 Video Compression

Digitizing videos in high quality asks for huge amounts of data storage. The following section lays out the basic ideas how to minimize this need with data compression and how to utilize them with modern mobile devices.

Film is a representation of continuous single pictures, also referred to as frames. When these frames are played back with sufficient tempo the audience perceives an optical illusion of continuous motion (Wertheimer, 1912). When looking at a physical filmstrip we can actually see the single frames. The first fully electronic television broadcast utilizing a cathode ray tube was made on 14 December 1930 by the German physicist Manfred von Ardenne (DRadio Wissen, 2014).

Over time different standards for analogue television broadcast has emerged in different countries such as NTSC (US), PAL (Germany) and SECAM (France). These three standards and varieties finally spread out over the world. In 1986 the Society of Motion Picture and Television Engineers (SMPTE) introduced D1, a digital recording standard bringing NTSC, PAL and SECAM to the digital domain. The digital format D1 is capable of digitizing and storing the analogue signals of NTSC, PAL and SECAM on magnetic tape (Baron and Wood, 2005). D1 as described in ITU-R 601 has a bitrate of around 150 Mbit/sec. With the introduction of high definition television (HDTV) the bitrate for digital video raised to approximately 1.5 Gbit/sec (Richer et al., 2006).

Due to the sheer amount of data video, streams often are compressed by utilizing a lossless or lossy compression method. While lossless compression can restore a data stream that is bit-identical to the original data stream, lossy compression cannot. However, the compression rates of lossy compressors are significantly higher than those of lossless compressors. Especially audiovisual data is suitable for lossy compression as the compressors can exploit shortcomings in human perception. Since a human will not able to tell the difference between the original and the lossy compression, the data rate can be significantly reduced. Advanced video compressors (codecs) do not only exploit spatial redundancy within a single frame, they also try to exploit temporal redundancy com-

paring consecutive frames. Modern lossy video codecs like MPEG-4 allow compressions rate between 20 and 200. High definition video sold on Blu-ray disks have a maximum data rate of 40 Mbit/s and thus a compression rate of approximately 40 (Blu-ray Disc Association, 2010), while the video encoding recommendations for the Android platform suggest 10 Mbit/s, a compression rate of 150 Android Developers (2015).

Besides its obvious advantages, high (lossy) compression rates come with downsides, such as high computing power especially for encoding but also for decoding, and the effect that with every generation (copy of a copy) the quality decreases. This has two implications for the design and the designer.

Due to advances of highly efficient video encoding and decoding microprocessors, high definition recording and playback on mobile devices is of lesser concern, especially since modern mobile operating systems utilize the capabilities of such microprocessors, e.g. Multimedia Framework (Google Android), Media Foundation (Windows Phone) or Quicktime Framework (Apple iOS). These multimedia architectures offer API's for audio and video playback. Depending on the architecture the API can offer advanced functionality such as audio and video filters, playback at various speed or reverse playback. However, when a novel interface/interaction design needs advanced functionality the API does not offer, designers and researcher are urged to work around these shortcomings. Depending on the fidelity of the interface/interaction design, this can become too high an obstacle to implement a viable prototype which offers meaningful insights.

To mitigate the degradation which is inherent to lossy compression, an obvious approach is to limit the number of generations (copy of a copy). Providing interface/interaction designs and applications for mobile devices that lower the need for multiple generations can help to circumvent the decline of quality. Non-linear video editing is one such approach, as it has the ability to define multiple editing steps (trim, re-order, etc.) before finally rendering the whole film in one pass.

3.3 Design and Evaluation Tools

Evaluating an interface or design proposal can be done by means of qualitative and quantitative measures. Points of critique are that a) qualitative gathering and analysis of data strongly depends on a researcher's skills and knowledge and b) quantitative interpretations are often based on a low number of participants and the difficulty of generalization. Critics often question the objectivity of quantitative data, as the data can hardly be gathered and collected completely without bias (Diekmann, 2007; Flick, 1995). However, when implemented and interpreted with care evaluation tools can be a powerful resource to assess new ideas and gain insights.

3.3.1 GOMS and Derivatives

GOMS is an acronym for Goals, Operators, Methods and Selection Rules and was proposed by Card et al. (1983). It is a formal method to describe both the knowledge of

a user to carry out a task and the ability of a system to accomplish an intended task (Han, 2011). The Goals describe a user's objectives or targets, the Operators name the tasks a user has to perform to accomplish the goals. The Methods deconstruct a user's goal into several sub-goals and attach a sequence of simple operators to them, whereas Selection Rules provide a set of condition clauses helping a user to achieve his or her goals. Gray et al. (1992) provide a good example of how well the GOMS model can predict a fall in productivity due to a change in workflow. Their article describes how GOMS was implemented, how the outcome hinted to the problems, and how important it is to understand the context of a given workflow.

KLM (Keystroke-Level Model) is a simplified version of GOMS leaving out all contextual information and focusing on six primitive operations needed to execute a task. These primitive operations are 1) pressing a key 2) pointing to a target utilizing a pointing device 3) moving a pointer in straight lines utilizing the pointing device 4) mental preparation 5) moving hands to an appropriate device (mostly between keyboard and pointing device) 6) waiting for the computer to execute a command. In KLM a user's task can be described as a sequence of these six primitive operations. A set of rules to simplify the sequence mathematically and an average time specifying each primitive operation allows the calculation of an average execution time for a given interaction. KLM was shaped with two concerns in mind: first, quick and easy usage during the design phase of interactive systems and second, high usability for computer system designers who are not trained in psychology or related matters (Card et al., 1980). The authors themselves mention the restrictions of KLM, such as KLM's applicability only for experienced users, error free performance and routine tasks, as GOMS and its subset KLM only predict completion time, leaving out all other aspects like ease of user, overall efficiency or user experience.

Trolltech, the company who started the cross-platform application framework Qt, proposed KLM-Qt, a mobile extension to KLM which is intended for standard keyboard usage (Schulz, 2008). The authors conclude both that KLM-Qt could only be a first step in defining a set of appropriate operations for mobile devices and that the Keystroke-Level Model can be a valid way to evaluate mobile user interfaces.

Rice and Lartigue (2014) focus more on the different interaction mechanisms by defining and compiling sets of interactions a user commonly faces when interacting with a touch-based interface. This also includes contextual parameters (Holleis et al., 2007) such as distractions. In reference to KLM, the authors named their compilation Touch-Level Model (TL-M), however, no real-world evaluations are provided yet.

El Batran and Dunlop (2014) proposed a set of timing information for touch-based gestures for mobile devices such as swipes, zooming and taps. Although the provided timing information is based on more than 3000 observations, the authors indicate the difficulty of applying these figures for different gesture amplitudes. It is apparent that various amplitudes correlate with different screen sizes and hence, measuring just execution times is not sufficient for a comprehensive evaluation.

GOMS and its derivatives can be used during the design phase to estimate the outcome of design decisions. The use of KLM is straightforward and can help to estimate the impact of design decisions. However, transferring KLM to the mobile domain requires not only adaptations in the operational tasks (swipe, tap and pinch instead of keyboard and pointing device), but also reconsideration of contextual changes such as concentrated work in the office on one hand and inattentive use in-situ on the other. Furthermore mobile devices with small screens often have to exchange the complete content of a screen during an interaction sequence, while desktop applications with more screen estate do not have to. Changing all the content of a screen during an interaction sequence puts an additional cognitive load on the user, as there is little or no visual feedback from the previous screen. While low interaction times can indicate a satisfied user experience, the figures KLM-Qt and TL-M deliver are not meant to measure user experience (UX), as UX involves a person's perception and anticipation of a system as well as the context where the system is used (Hassenzahl and Tractinsky, 2006; ISO, 1998). Both clearly are outside of the scope of all KLM-like methods. To address these more holistic variables other methods are needed that reach beyond (however advanced) timing.

3.3.2 Gesture Notations

Gesture notations are sketches expressing touch-based interactions and can be used to better understand and assess an interaction design before it is implemented. Thus, gesture notations can provide a more holistic impression, helping designers/researchers to critically convey and discuss their ideas in the first place, e.g. Buxton (2010) repeatedly indicates the importance of sketches as a tool for explaining and debating design ideas. However useful sketches are, they have one inherent constraint. Sketches are not defined, and every designer can sketch an interaction differently even though the underlying idea is the same. This can lead to verbose annotations explaining the sketch.

One way to overcome this problem is to stick to the guidelines of the device or operating system the manufacturer provides for their respective mobile operating system. Although these guidelines are useful, they have limitations. For example, platform specific gesture notations are based on the overall interaction approach of a platform and this varies from platform to platform. While Android favors a dedicated "back" button on the lower left of the device, iOS guidelines give preference to a software button on the upper left. Furthermore, guidelines do not anticipate novel ideas, such as Bezel swipe that was proposed by Roth and Turner (2009) allowing a user to make a swipe gesture over the edge of a device. Today, 2016, virtually every mobile touch-based device makes use of that gesture. And finally, every gesture notation is different, aggravating the design of platform independent designs. Especially with the rise of multi-platform developer tools such as Cordova/PhoneGap[1] or Xamarin[2] allowing a "code once - deploy everywhere" approach, a more general gesture notation is preferable. Thus, manufacturer specific

[1] cordova.apache.org
[2] xamarin.com

guidelines and notations are important, however, not always are they the appropriate tool to discuss the interaction mechanics of a novel application.

This very idea has been brought up and discussed by various designers and developers and some collections of touch-based gestures have been compiled (Wroblewski, 2010). However, developers often need to define new gestures that are not available right now (Khandkar, 2010). While some research has been done in providing programmers with programming frameworks to implement novel gestures that are not supported on a given platform, i.e. GDL - Gesture Definition Language (Khandkar, 2010), there is a lack of such a definition for an extensible gesture notation.

So far, a variety of tools exist for the different phases in a design process for mobile interfaces/interactions. These include sketch-like gesture notations for presenting and discussing design ideas as well as tools for assessing and evaluating these ideas. What is still missing is an extensible, platform agnostic gesture notation that is useful in every phase in a design process and a foundation for various approaches, existing and prospective.

CHAPTER 4

Methodology

To design interface and interaction mechanisms that are understood and accepted by a user, a researcher or designer can choose from a set of established methods. These methods either complement each other or try to achieve the same goals with different approaches. However, it is important to notice that methods are applied in a feedback-loop, allowing the designer and researcher to gradually improve the final product. The loop itself can be divided into three overarching stages:

- Understand: Understanding the context is important for developing artefacts that are useful and feasible. Furthermore the acceptance of such artefacts is higher when the needs of the users are understood.
- Make: Making artefacts is sometimes called designing artefacts. During this thesis the "making" process is based on a prototypical approach. Depending on the specific task and interaction the implementation can have many features with little detail (horizontal prototype) or can include models with few features but much detail (vertical prototype).
- Evaluate: Testing and evaluating the implemented interaction (made artefacts) is necessary to gain new insights and to check and critique chosen design decisions. The outcome of the evaluation is part of the input for the first step in the loop, "understand". This loop iterates several times until the evaluation determines the design offers a viable contribution to scientific discussion and user satisfaction

4.1 About Design

Before we introduce the various methods, a short reflection on design is necessary. What is design, and what is designing? Designing begins with understanding the problem and

ends with an artefact that is intended to solve the problem. Therefore design could be whatever is built to solve a given problem or whatever has a purpose and had gone through a design process before. Thinking about a given problem and implementing a prototype are mutual and complementary in the process of designing. "In actual reflection-in-action [...] each feeds the other, and each sets the boundaries for the other" (Schön, 1983, p. 280). Dorst and Cross (2001) also say that "problem space and the solution space co-evolve together", indicating a feedback loop when designing. Maher et al. (1996) also point in this direction when stating that the understanding of the problem and the solution for the problem change over time. Goldschmidt (1992) too mentions the mutual influence of problem space and solution space and suggests sketches as an effective instrument for interacting between these two spaces.

However, how designers draft a first vision of an artefact which bridges the gap between problem space and design space remains elusive. One approach is to describe designers as persons who just have the ability to carry out creative interventions (Buchanan, 1992). Other researchers take the opposing view, that there is always something which came before, something the designers build on (Jonas, 2012). This includes, for example, a designer's own experience (Buchanan, 1992). Others argue that experience cannot be the only explanation, as a designer would approach any new design challenge with the same naive approach Petruschat (2011). Others suggest that good artefacts are built on top of existing artefacts Atwood et al. (2002), and consequently, "[a]fter many generations of evolution the end product becomes a total response to the problem" (Lawson, 2006). An intuitive and creative approach is seen as vital by some researchers (Jones, 1992; van den Boom, 2011) even it is hard to reflect on them (Bartneck, 2009; Stolterman, 2008) as intuitivity can be described as a mixture of rules of thumb, simple heuristics (Gigerenzer and Todd, 1999) and developed skills (Damasio, 2014). Daley (1982) makes a more pragmatic statement on this topic: "The way designers work may be [...] literally indescribable in linguistic terms"

A designer can work alone and rely on his or her genius, or a designer can co-operate with others, whereby the partners do not have to be designers as well. Quite the contrary, it can be favorable for a project to incorporate diverse opinions and domain expertise and a designer's task is to distillate everything into one satisfying design. Rittel and Webber (1973) describe taming wicked problems by incorporating various stakeholders to paint a holistic picture of a given situation and to gather as much input as possible for a sensible proposal. This was seized on by among others Krippendorff (2007), who also favors participatory and human centered design processes for complex problems.

We can conclude that designing is a complex task that deals with various complex, sometimes opposing constraints at the same time. Therefore a designer has to analyze the current state and synthesize a future artefact which best fits this state. It does not matter which comes first as analyzing and synthesizing are intervened and carried out repeatedly (Schön, 1983; Dorst and Cross, 2001). To better identify weaknesses and spot potential improvements an analytical evaluation process can be weaved in the design process. Artefacts can be evaluated based on quantitative or qualitative methods (Heinrich and

Häntschel, 1999). The results of a decently applied evaluation can loosen the burden for a designer or a design team from having to decide solely on instincts or gut feeling. And an in-depth discussion of the evaluation results can help to systematically scrutinize a given artefact. However, not only the evaluation can be processed systematically as there exists a varied selection of different methods applicable for every step in the design process. And again, it is in the designer's hand to choose the right method in the right moment.

4.2 Research Methods

When designing and implementing novel interface and interaction designs a researcher has a multitude of methods at hand. The endeavor is applying the right method for the right task. And beside strict methods exist a plethora of pragmatic guidelines when implementing an application. A well-defined, well known and often criticized approach is the waterfall model. When presenting the waterfall model (not the original author's title) the author gave his personal views about managing large software developments. In the same publication he stated that his model „is risky and invites failure“ (Royce, 1970). Nevertheless, the waterfall model became very popular. The life cycle of the waterfall model starts with gathering the system and software requirements. The requirements are analyzed and the program is designed. Subsequently the system is implemented and tested, and finally the system is rolled out and into operation.

The waterfall model is very straightforward, with vivid descriptions for and clear distinctions between the single steps during a project. Nonetheless, the model has drawbacks. For example, user tests are at the end of the life cycle, making changes in the product complex and costly. This, among other reasons, led to official statements by US Department of Defense to „remove the waterfall bias“ (Larman and Basili, 2003) from their projects. Another reason is that designers do not always know all requirements beforehand, and often literally cannot know all requirements in advance.

4.3 User-centered Design

User-centered design is built on the assumption that a user often knows best what he or she needs and it is in a designer's hands to help a user to describe a user's future product. In this case the designer becomes the translator of the user needs and goals (Saffer, 2010). Similar approaches are participatory design and contextual design. The main contribution of user-centered design, participatory design or contextual design is to realize that the user and the context a user works in are vital input parameters for the design process. Furthermore, a user should actively attend a project from the beginning and not only as a sanctioning body at the very end.

User-centered design (UCD) follows an iterative pattern (Figure 4.1) with the main phases being identifying the people who will use the product and the circumstances under which they will use the product (Specify the Context of Use), identifying the

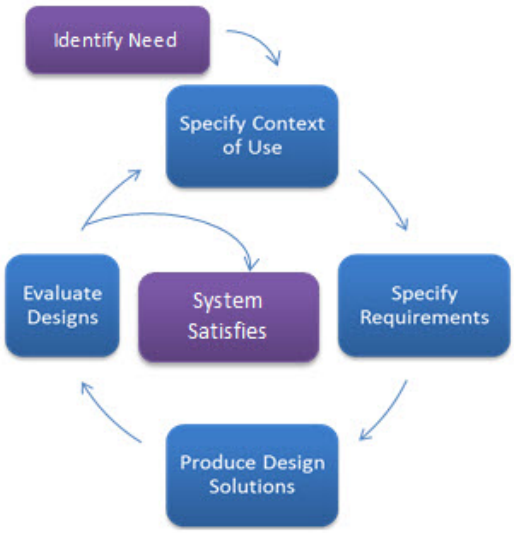


Figure 4.1: User-centered Design Model

user's goals (Specify Requirements), defining a concept and implementing a solution upon it (Produce Design Solutions) and testing and evaluating with actual users (Evaluate Designs). The methods used for the single phases of the UCD process are not strictly specified, and can be combined with other existing models and approaches such as agile software development or even the waterfall model (U.S. Department of Health & Human Services, 2016).

Even though the close involvement of users throughout a project can help to raise acceptance of the final product, the method also has drawbacks. First, it is not easy for untrained users to envision a new product, thus, a user is easily caught in the more of the same trap, i.e. demanding only a faster product even though the current product does not fit in general. Second, users are focused on their own tasks and often lack a broader perspective. This leads to requirements specialized on a very narrow application area, ignoring other parts of and participants in a project. Third, users can have different goals or personal (hidden) agendas a designer is not aware of. It is in the designer's hands to anticipate these shortcomings and actively level them for the best of the project.

System design and expert design are two variations on user-centered design, each with a different focus. Expert design builds on the knowledge of an experienced designer who has a great deal of knowledge and experience. However, even design experts do not necessarily have expertise in a given domain. Furthermore, it can be challenging for an expert to distinguish between his/her own experiences and knowledge and the experiences and knowledge a user has. Thus, a design that perfectly fits the needs and understanding of an expert does not necessarily fit the needs and the understanding of an untrained user. However, the pace with which design decisions can be made with expert design (Saffer,

2010) is a significant benefit. System design, in contrast, is a more engineering-based approach and complements the classical user-centered design process. System design is well suited for transferring often fuzzy requirements into definite technical solutions (Saffer, 2010). Regardless of the design approach, improving the safety and certainty of fuzzy requirements is a repetitive goal a designer faces. In order to do so he or she has a variety of methods such as observation, interviews, sketches or workflow analysis.

Observation is a method to gain real life data about users by monitoring what they do and how they do it. The collected data is context rich as the observed person acts and works in his or her accustomed environment. Depending on a designer's or researcher's activity and involvement in the user's routine observations can be distinguished in a number of categories (Saffer, 2010):

- Fly on the wall: unobtrusively observing.
- Shadowing: following user when they carry out their routines.
- Contextual inquiry: calling the routine of the user into question and asking the user why he/she is actually doing this in one way or another.
- Undercover agent: interacting with users while not letting them know that they are being observed.

Interviews "[are] a conversation between two or more people where questions are asked by the interviewer to elicit facts or statements from the interviewee" (Mirriam Webster, 2016). As a research method interviews can be distinct in various sub-methods.

- closed interview: : a range of predetermined answers are given, the interviewed person chooses an answer
- open interview: answer are not predetermined, the interviewed person answers in his or her own words
- structured interview: the procedure of the interview is rigid
- semi-structured interview: the procedure of the interview is flexible

Sketches are fast drawn drafts utilized as a way of eliciting ideas when reasoning about a design entity (Goldschmidt, 1992). The more complicated an artefact is, the harder it is to keep track of all combinations of internal and external states or all mutual and multilateral dependencies. Sketching as a method can help to easily leave out or integrate the level of detail, and thus, allowing a designer to continue the process of discussion. However, for a final interpretation all generalizations and simplifications must be considered and discussed (Schön, 1983). Sketches can be a useful tool when bridging the gap between users and experts (engineers) and the diverging points of view they

have on an artefact. An engineer has a deeper understanding of the inner processes and, consequently, can interpret the inner states. In contrast, for a normal user these inner states are normally without a deeper meaning. Knowing little or nothing about the internal mechanism, users demand information that is useful for them to complete their tasks. Thus, an interface that is based on the task is often more appropriate than an interface based on the system mechanism (Gentner and Grudin, 1990). Sketches can be a useful tool when discussing the conversion from internal processes to useful interface elements.

While engineers have knowledge about the internal functioning of an artefact, users have knowledge about their workplace and their daily workflow. Due to deep familiarity with a working environment, it can be sometimes difficult for a user to correctly describe the single sequences of a daily routine. However, work sequences carried out by a user over a period of time inevitably lead to both a broader understanding of a work environment and groom the skills needed for the work. Thus tapping into this knowledge can be quite beneficial for the designer. Personas are a design method to describe the behavior, motivations and expectations of an (imaginary) person in detail. This (imaginary) person can act as an archetypical person who interacts with a product, service or system (Saffer, 2010). Personas visualize and describe an amalgamated user the future and can help to tailor a system in favor of this (archetypical) user. Robert Reimann and Kim Goodwin developed personas further by describing three distinct goals for a persona: experience goals, end goals and life goals. The experience goals deal with the feeling of a persona when interacting with the system, the end goals focus on what a user actually wants when accomplishing a task, and the life goals describe the context in which the system interacts with the user (Saffer, 2010).

The data acquired by user-centered design methods are not by default meaningful. Turning the data into information allows a designer or researcher to make reasonable judgments. To do this, data can be structured in several ways, such as clustering similar pieces of data, or juxtaposing related pieces of data and naming the resulting clusters (Saffer, 2010). In a second step the data can be analyzed, summed, extrapolated and abstracted (Saffer, 2010). The analysis breaks data down into smaller chunks allowing these chunks to be grouped according to determined categories such as activities or objects. While the analysis works at a micro level, the summation lifts the information to a macro level. The extrapolation in turn derives conclusions from the analyzed and summed information and tries to anticipate (a small glimpse of) the future. Finally, the abstraction of the data is a last and important task helping to understand the conceptual models users have. The methods implemented when researching and designing novel interfaces address three distinctive categories. First, pointing out and describing areas in need of improvement, second, depicting the opportunities for improvement in the current design, and third, guiding the designer and researcher to these areas and opportunities (Saffer, 2010).

4.4 Guidelines

In contrast to user-centered design methods, which are mainly utilized to gain insight within a novel research domain, guidelines are insights already consolidated, synthesized and ready for use. The process of analyzing, summarizing, extrapolating and abstracting is completed and generalized. It is in a researcher's and designer's hands to apply the right guideline for the right problem, and even more importantly, it is in a researcher's and designer's hands to break and overrule a guideline. Deciding when to use or ignore a guideline, always for the user's best interest, requires experience on the researcher's or designer's side. Ultimately, the variables that have to be leveled are the same for virtually every practical artefact: usability, feasibility/suitability and affordability.

4.4.1 Usability

The International Organisation for Standardisation (ISO) 9241-11:1998 defines usability as "The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use". Furthermore, the standard characterizes three important dimensions for usability:

- Effectiveness: the accuracy and completeness with which users achieve goals
- Efficiency: the resources expended in relation to the accuracy and completeness
- Satisfaction: the comfort and acceptability of use

Effectiveness and efficiency can be measured as time and resources needed to accomplish a given task. In contrast, satisfaction is a subtle emotional condition which eludes objectification and is difficult to measure.

However, there exists reasonable research how satisfaction can be supported, how dissatisfaction can be evaded, and what approaches can be implemented to achieve these goals. Abraham Maslow states in his hierarchy of needs that unless an individual's basic needs have been met, higher levels in the pyramid are of no relevance (Benson and Dundis, 2003), i.e. before reaching out for self-esteem an individual will need to fulfill more elementary needs such as food and shelter. Equally, in human computer interaction a design must serve low-level needs first such as functionality and reliability before high level needs such as usability can be served (Lidwell et al., 2007). Another important observation is the degree of immersion a user has or feels when interacting with a technical artefact. Whenever people feel underchallenged they become bored or apathetic and whenever they feel over-challenged they become stressed or frustrated (Lidwell et al., 2007). Thus, neither over- nor under-challenging users seems a promising strategy to keep users in balance, but this is easier said than done. The problem begins with defining a user's proficiency and experience with a system, as this has a direct impact on the complexity a user can handle. However, over time a user learns to interact

with a system and become more confident in using it, so this definition is dynamic. And finally, due to a lack of concrete knowledge about the internal processes of a product, a user often imagine these processes on his or her own, which also must be accounted for. Research suggests that this translation of events into a consistent model allows a user to comprehend a system and anticipate its behavior (Staggers and Norcio, 1993). Thus, as designers and researchers we are facing conflicting and/or varying positions that must be balanced:

- A trade-off between learning and ease of use (satisfaction) on one hand, and flexibility and efficiency on the other (Gentner and Nielsen, 1996).
- Cognitive Load is the amount of mental activity that is required to accomplish a goal (Lidwell et al., 2007) and must be kept within a range of tolerance. While reducing cognitive load is a generally favorable proposition, when a task becomes stupidly repetitive with limited cognitive load users will disengage.
- A tradeoff between flexibility and ease of use is unavoidable (Odlyzko, 1999) as computers became general-purpose devices for virtually every field of application, and consequently, can't be optimized for any individual task (Norman, 1981). It is in a designer's hands to find the balance between flexibility and usability; to know that "everything is possible" and at the same time concentrate on the main aspects of a design.

As satisfaction is linked to users' state of mind and self-esteem, Csikszentmihalyi (2008) suggests immediate feedback and control over their actions as fundamental elements in feasible interface and interaction design.

4.4.2 Feasibility

Feasibility defines whether an artefact is capable of supporting a user in accomplishing a task. Depending on the task and the artefact a variety of evidence based guidelines are in use to provide designers and researchers. Nilsson (2009), Park et al. (2011) among others have identified and combined various factors to develop user interface guidelines. However, as technology and user proficiency is ever changing and evolving guidelines should be assessed critically whenever applied.

The quote "form follows function" is credited to an ancient Roman architect (Sullivan, 1956) and describes the idea that a building itself and its very purpose should be the guideline for its form. Jackson (1993), a proponent of form follows function, argues against design that simply shrouds its technical internals. (Norman, 2013) addresses the same ideas when stating that artefacts should have unambiguous input / output mechanisms.

In contrast, Kurosu and Kashimura (1995) indicated in their paper that the aesthetic usability is strongly affected by the aesthetic aspects rather than the inherent usability.

Thus, for a user an appealing design can be perceived as having a superior usability even though it has not been compared to another usability design which is not equally aesthetically pleasing. Aesthetic design promotes positive thinking, fosters positive relationships and consequently makes people more tolerant of design flaws (Lidwell et al., 2007).

No matter how mature an artefact is, there is always the chance for an error, and a feasible interface and interaction design can help to minimize false input or misunderstandings in the way users perceive and interact with an error. Norman distinguishes between two main categories for errors, slips and mistakes. While a mistake is an error of intention a slip is an error of execution (Norman, 1981). Thus, an error is intentionally planned by the user due to a misunderstanding of underlying concepts. For example, a user only clicks on buttons that are visible not knowing that more buttons were available when the user scrolls up or down. Slips, on the contrary, normally occur due to a lack of concentration given to a certain task. For example, a well known daily procedure (i.e. driving home) is changed (i.e. stop at the supermarket before to get some milk) and the users keep to the procedure they are used to (i.e. driving home without going to the supermarket before), even though they fall within the process. Slips and mistakes are not completely avoidable, however, interface and interaction design can help to reduce them by providing clear and distinctive feedback, minimizing information noise and supplying standardization (Lidwell et al., 2007).

Other approaches to minimizing the consequences of a slip or a mistake are action reversibility, action confirmation and consequence warning. Reversibility, often titled undo, enjoys several advantages over both confirmations and warnings. First, the user does not have to identify all possible consequences beforehand since every interaction can be undone. Second, unlike a warning or a confirmation, action reversibility does not necessarily interrupt workflow. In contrast, reversibility encourages users to explore the possibilities of an artefact and naturally learn through trial and error.

4.4.3 Affordability

By affordability we refer to both the cost of an artefact for the user and the total time and money spent to design and implement a prototypical artefact by the producer. While usability and feasibility are focused on ensuring high user acceptance, guidelines for affordability aim to produce a working prototype within a reasonable time span and budget. Three widely discussed and often applied guidelines (rules of thumb) are the 80/20 rule, Occam's razor and an iterative development cycle.

The 80/20 rule argues that 20 percent of the functions in a software product fulfill 80 percent of a user's needs. This relations of 80 to 20 can be observed in variations in different fields of application and was first described by Italian economist Vilfredo Pareto when studying the distribution of land in Italy in the late 19th century where 20 percent of the population owned 80 percent of the land (Teich and Faddoul, 2013). This 80/20 distribution can be found in various fields and contexts including economics, software

engineering and design, to name a few. In economics 20 percent of the customers generate 80 percent of the revenue (Weinstein, 2002), in software engineering 80 percent of errors are caused by 20 percent of the code (Gittens et al., 2005) and in design, as previously mentioned, 80 percent of a product's usage involves 20 percent of its features (Lidwell et al., 2007). When designing according to the 80/20 rule it is on the designer and researcher to identify 20 percent functions that make a product 80 percent usable and focus on them. Experience and self-assurance are needed to assess different and often opposite requirements and statements from customers and users.

Occam's razor is an approach that can support a designer or researcher in decision making by favoring simplicity over complexity. It is attributed to William Occam's (1287–1347) and is based on the idea that simple hypotheses' can be assessed and evaluated more easily than complex ones. Applied to interface design, Occam's idea suggests that whenever there is a decision between two design alternatives, all other variables being equal, choose the simpler one. While this goes well with other design recommendations such as decluttering interfaces, it is still necessary for the researcher or designer to determine which one is the simpler. This may be obvious at times and no so obvious at other times. And, of course, this does not guarantee success. Courtney and Courtney (2008) stated "there are many examples where Occam's razor would have picked the wrong theory given the available data". Therefore, even though Occam's razor is a simple yet plausible and powerful guideline, the actual task, the alternatives and the context determine its applicability. And, a designer or researcher should always re-question and re-evaluate a design decision as the development goes on. This can be addressed and supported with appropriate development and process models.

An iterative development cycle allows and supports the re-evaluation of design decisions by its very nature. Iterations in the process of developing interface and interaction designs can encourage the reconsideration and evaluation of design decisions, as the design is not statically determined. A common problem when iterating can be the absence of a defined endpoint as each iteration refines a given design and reveals new starting points for further refinement (Lidwell et al., 2007). Establishing clear criteria and design goals at the beginning of the process can help to decide when a last iteration has to be made.

All the mentioned and briefly described methods just cover a small fraction of all existing approaches in design and research, whereby it is often hard to distinguish methods as they can be very similar and differ just in small details. However, they represent the methods that were mainly used when designing and implementing the various interface and interaction designs which are summarized in the next chapter.

CHAPTER 5

Summary of the Scientific Papers

This chapter sums up the following papers of which the author is the sole contributor or the prime contributor. All papers are full papers, peer reviewed, presented at international conferences and published under the authority of the Association for Computing Machinery (ACM) or the Institute of Electrical and Electronics Engineers (IEEE).

5.1 Paper 1 - ProPane: Fast and Precise Video Browsing on Mobile Phones

ProPane: fast and precise video browsing on mobile phones. In Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia (Ulm / Germany, 2012). ACM.

Introduction: Roman Ganhör

Related Work: Roman Ganhör

Implementation: Roman Ganhör

User Study and Evaluation: Roman Ganhör

Conclusion and Future Work: Roman Ganhör

This work proposes a novel browsing mechanism for video clips on mobile devices that is suitable for professional needs, i.e. video editing. Professional video editors need precise control when browsing footage, however, they also need the possibility to quickly skim through the footage. Initial interviews and studying existing video editing software for desktop use revealed following browsing speeds: frame by frame, very slow, slow, normal, fast, very fast. The paper addresses the requirements for mentioned precise and fast browsing on one hand and the constraints of limited screen estate for interaction elements

on the other. To overcome the need for a multitude of interaction elements, each assigned to a predetermined speed, a temporal/spatial approach is proposed. Instead of defining single interaction elements for specific browsing speeds, the paper determines a single interaction area for all browsing speeds, and the browsing speed is calculated depending on the starting position and the distance of the thumb from the starting position.

Figure 5.1 depicts the different areas of the proposed interface. Area A is for browsing backwards in the video clip, area B is for browsing forward in the video clip, area C shows the actual video clip and area D gives away additional metadata such as time and position in the video clip. Figure 5.2 illustrates how different starting points allow for different browsing speeds (slow, normal, fast). If the finger taps on the top (the starting point is on the top) the possible browsing speeds are 30fps at the starting point, 15fps when moving the finger to the center or 6fps when moving the finger to the bottom. If the finger taps on the center (the starting point is on the middle) the possible browsing speeds are 30fps at the starting point, 60fps when moving the finger to the top or 15fps when moving the finger to the bottom. If the finger taps on the bottom (the starting point is on the bottom) the possible browsing speeds are 30fps at the starting point, 60fps when moving the finger to the center or 120fps when moving the finger to the top.

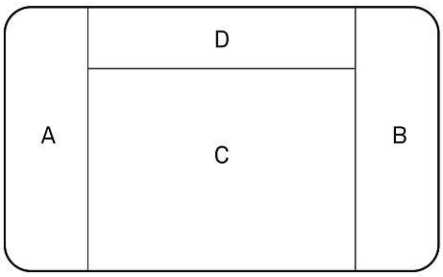


Figure 5.1: ProPane browsing panes

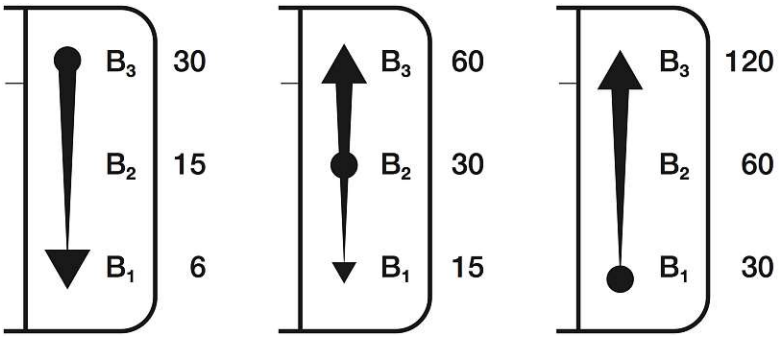


Figure 5.2: Different browsing speed depending on the starting point (left: slow; center: normal; right: fast)

This approach allows smooth changes of browsing speeds and consumes less screen estate compared to existing browsing mechanisms. The evaluation was conducted with regular

smartphone users and professional video editors. While the professional users intuitively understood the advantage of the combined browsing area, regular users had first to understand where all the different browsing speeds can be applied in a meaningful way. However, all participants easily understood the interaction mechanics and all of the professionals and most of the regular users were in favor to add such browsing capability to their favorite mobile video player.

5.2 Paper 2 - Athmos: Focus+Context for Browsing in Mobile Thumbnail Collections

Athmos: Focus+ Context for Browsing in Mobile Thumbnail Collections. In Proceedings of International Conference on Multimedia Retrieval (Glasgow / Scotland, 2014). ACM.

Introduction: Roman Ganhör

Background and Related Work: Roman Ganhör

Implementation

- Layout: Roman Ganhör
- Thumbnail Sizes: Roman Ganhör
- Gestures: Roman Ganhör
- Prototype: Roman Ganhör, Jakob Frohnwieser (during the academic course From Design To Software under the supervision of Roman Ganhör)

Evaluation: Roman Ganhör

Conclusion and Future Work: Roman Ganhör

The paper deals with browsing in media collections where single media assets are represented by thumbnails. While there is sufficient academic research on the general topic of browsing and searching in media-assets collections, little work is done in the area of touch based mobile devices with small screens.

However, the paper describes an interface that combines several existing approaches for browsing large data sets, particularly focus+context, and adapts them to the mobile domain. The interface aims for three goals. First, at any given time all thumbnails of a collection shall be visible on the screen to serve as context. Second, the user is aware of the actual position within the collection. Third, at least a small set of thumbnails should be big enough for identifying their content without the need for magnification. Additionally, the interaction design is built on intuitive gestures allowing for fast and precise browsing and searching.

Figure 5.3 depicts how the basic idea of a film strip with the focus on the current range and the context being the starting range and the ending range is transferred to a mobile

device. The three ranges are split and re-ordered to fit on the limited screen estate with the image on the top left being the first image in the collection and the image on the bottom right being the last image in the collection. The applied interaction mechanism allows for browsing on a picture by picture basis when swiping left to right or right to left at the center of the screen. However, a user can also bring any image into focus quickly by performing a swipe gesture from any picture to the center. The thumbnail sizes in the current range (focus) are predetermined and the thumbnail sizes in the starting range and ending range are calculated depending on the number of thumbnails in the respective range. Even the thumbnails get distorted as there is no other way to squeeze them in the given screen estate the whole impression gives away an appropriate overview of all thumbnails in the media collection.

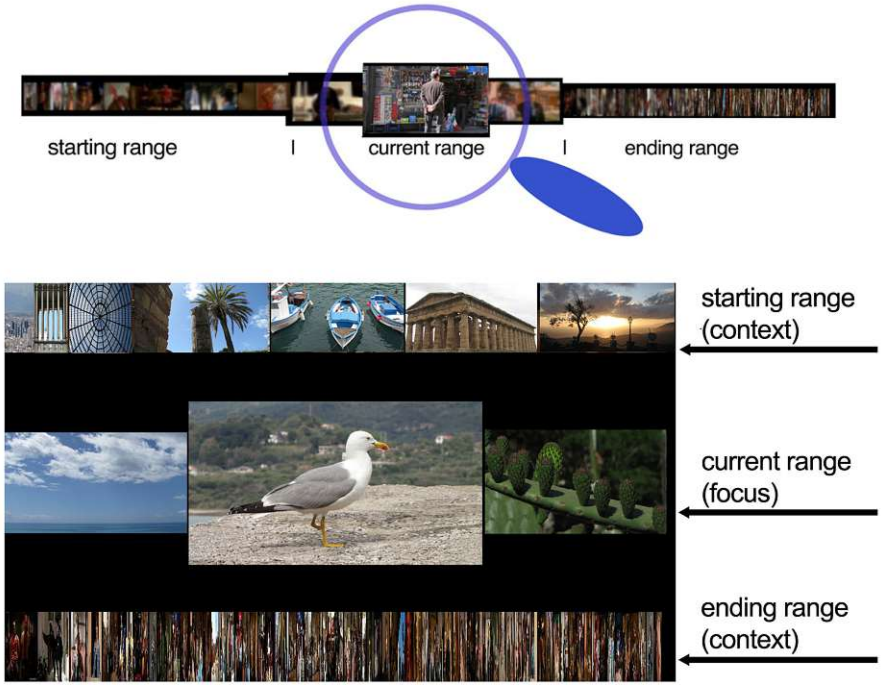


Figure 5.3: Focus+context transferred from the film strip metaphore (top) to the mobile screen (bottom)

During a conducted evaluation a group of participants with various professional backgrounds carried out predetermined tasks with both, a standard browsing application and the proposed application. Despite its early prototype-status users did not complain about the proposed interface, on the contrary, every user suggested additional gestures and provided ideas to improve the interface. The amount and quality of suggestion given by the participants indicated that the idea of the interface/interaction was well understood. Considering the general positive feedback it seems promising to add more advanced browsing capabilities.

5.3 Paper 3 - Muvee - An Alternative Approach to Mobile Video Trimming

Paper 3: Muvee: An Alternative Approach to Mobile Video Trimming. In Proceedings of International Symposium on Multimedia (TaiChung / Taiwan, 2014) IEEE.

Introduction: Roman Ganhör

Background and Related Work: Roman Ganhör

Implementation: Roman Ganhör

Evaluation: Roman Ganhör

Conclusion and Future Work: Roman Ganhör

The demand for effective and efficient interfaces for mobile video editing rises with the opportunity to record videos on mobile devices, i.e. news agencies like CNN and BBC already turn their audience into video based news-generators. However, little research exists on novel or alternative interfaces for trimming or editing video clips that elaborate on the possibilities of mobile touch based devices.

To define a suitable interface and interaction design for mobile video editing a participatory design process with professional video editors was applied on top of extensive literature research and market analysis. After this initial design phase a set of must-have functions were specified, and in a second step the necessary interface/interaction elements had to be arranged to be unambiguous, easy and efficient to use. In contrast to existing applications and interfaces for video editing, the final design proposal extends the timeline metaphor known from most existing video editing applications.

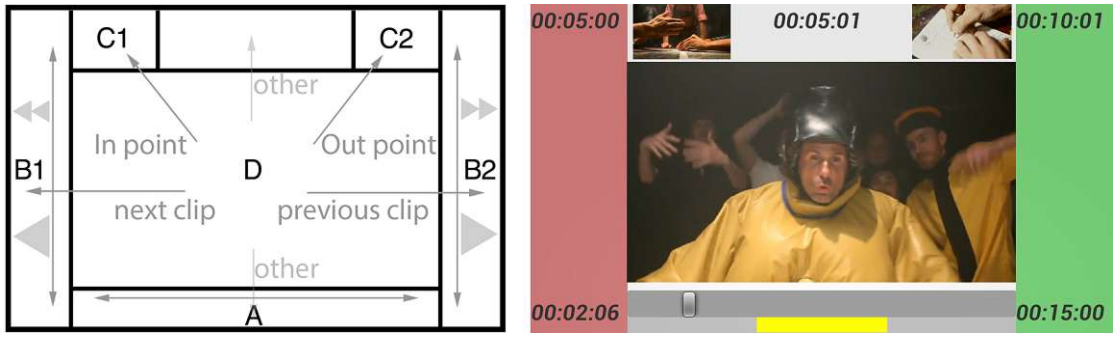


Figure 5.4: On the left the interaction mechanic is depicted and on the right the implementation can be seen

The left half of Figure 5.4 shows the various interaction elements schematically while the right half is a screenshot of the actual implementation: B1 and B2 are for browsing within a video clip at various speeds, area D shows the content of a video clip, A is a

minimal timeline and C1/C2 are needed for trimming. A swipe gesture from the current frame (area D) to C1 sets the In point for the video clip and replaces the thumbnail in C1 with a thumbnail of the new In point. A swipe gesture from the current frame (area D) to C2 sets the Out point for the video clip and replaces the thumbnail in C2 with a thumbnail of the new Out point. The yellow bar on the bottom of area A shows the length and position of the trimmed video clip within the whole video clip.

Utilizing the timeline (area A) mainly information purposes and not for interaction purposes allows for more spacious interaction mechanics as the interaction has not to concentrate on the timeline. The proposed interface and interaction design was qualitatively and quantitatively evaluated against one of the most used video editing applications for mobile devices using a timeline metaphor,iMovie by Apple Inc. The quantitative evaluation measured ten typical trimming tasks carried out during video editing. The statistics shows that the proposed interface is significantly faster in eight out of ten tasks. Additionally the qualitative evaluation showed a strong indication for the usefulness of the proposed interface. The participants, some being professional video editors, felt more comfortable with the proposed interface as they felt with iMovie, even iMovie utilized the well known and established timeline metaphor.

5.4 Paper 4 - INSERT: Efficient Sorting of Images on Mobile Devices

INSERT: Efficient Sorting of Images on Mobile Devices. In Proceedings of the Annual Meeting of the Australian Special Interest Group for Computer Human Interaction (Melbourne/Australia, 2015). ACM.

Introduction: Roman Ganhör, Florian Güldenpfennig

Background: Roman Ganhör

Implementation of Insert:

Interface: Roman Ganhör

Interaction: Roman Ganhör

Prototype: Roman Ganhör, Sophie Weiss (during the academic course From Design To Software under the supervision of Roman Ganhör)

Discussion: Roman Ganhör, Florian Güldenpfennig

Future Work: Roman Ganhör, Florian Güldenpfennig

Even though automatized grouping can be a powerful and convenient feature, it often does and can not satisfy the users' intentions or needs. All too often the discrepancy between user expectation and automated results lead to frustrating user experiences.

Thus, it comes as little surprise that there are also less 'mechanical' and much more playful approaches to exploring and ordering photo collections,

Therefore, the paper proposes a novel mobile phone application for supporting manual sorting of photo collections in an efficient fashion. The application divides the screen estate into two areas, browsing and selecting, and draws on different concepts such as overview+detail.

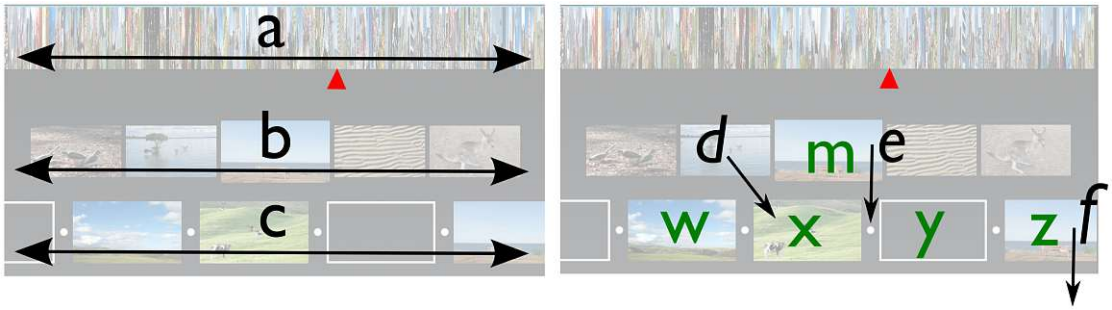


Figure 5.5: Depiction on the browsing mechanics (left) and the selection mechanics (right)

The left half of Figure 5.5 depicts the interaction mechanics needed for browsing a collection, both coarse and fine while the right half schematically outlines the interaction mechanics for selecting and ordering single assets. In the following we address the interaction gestures and interaction elements accordingly to Figure 5.5 with a letter (a, b, c, d, e, f, m, w, x, y, z). Swiping along the top area (a) allows for coarse browsing in the media collection. The red triangle indicates the current position within the collection. The thumbnails in the the top area are distorted allowing them to be squeezed in the limited space, however, the middle area (b) allows for fine browsing attaching a swipe gesture to jumping to the next or previous thumbnail. The thumbnails in the middle area (b) are not distorted and are sufficiently big to determine their content. While area (a) and area (b) represent the current collection of thumbnails area (c) holds all selected thumbnails. To move a thumbnail from area (b) to area (c) a user executes a swipe gesture from area (b) to area (c). This swipes gestures are indicated as (d) and (e) in Figure 5.5. If a gesture's end point is an existing thumbnail in area (c) the existing thumbnail will be overwritten with a the new thumbnail, i.e. gesture (d) will overwrite the existing thumbnail (x). Adding a thumbnail to the collection in area (c) instead of overwriting an existing one can be done by either moving a thumbnail to an empty placeholder (y) or moving a thumbnail to a bullet that generates a new placeholder and fills the placeholder with the thumbnail, i.e. the gesture (e) generates a new placeholder between (x) and (y) and fills the placeholder with the thumbnail (m). Removing a thumbnail from the selected thumbnail area (c) can be carried out with gesture (f) applied to thumbnail (z). After the gesture is carried out the thumbnail is deleted and an empty placeholder remains.

A conducted user study participants showed that the proposed interaction mechanisms were well perceived and that there is yet much research to be conducted aiming at the management of image collections on mobile device, in particular with small screens. Even though, the majority of the participants liked the user experience, there were also strong indications that, the more photos a participants regularly shoots the more he/she is interested in an application that allows organizing media assets directly on the smartphone.

5.5 Paper 5 - Monox: Extensible Gesture Notation for Mobile Devices

Monox: extensible gesture notation for mobile devices. In Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services (Toronto / Canada, 2014). ACM.

Introduction: Roman Ganhör

Background and Related Work: Roman Ganhör

Monox - Gesture Graphics:

- Tapping Gesture: Roman Ganhör
- Moving Gesture: Roman Ganhör
- Multi-Touch Gesture: Wolfgang Spreicer
- Screen Change: Roman Ganhör
- Rare Gestures: Wolfgang Spreicer

Evaluation:

- First Iteration: Roman Ganhör
- Second Iteration: Roman Ganhör, Wolfgang Spreicer
- Third Iteration: Roman Ganhör, Wolfgang Spreicer

Discussion and Conclusion: Roman Ganhör, Wolfgang Spreicer

Sketching is a useful tool when proposing, refining and discussing novel interaction mechanisms and application designs. However, when reviewing existing research on touch based interface and interaction mechanism it became apparent that almost every designer, researcher or system vendor has their own notation to describe an idea. While these notations are often ad-hoc and optimized for a given problem, this fragmentation has downsides. First, it complicates discussions as a same interface element or touch based interaction often have different notations and the notations can vary notably. Second, it aggravates literature review as the diverse notations must be aligned. Third, it accelerates the numbers throwaway notations as single purpose ad-hoc notations often lack of a set

of rules of how they can be extended. This paper presents a concept of an extensible sketching notation for mobile gestures, that can provide a common basis for collaborative design and analysis of mobile interactions. The concept is platform independent and enables general discussions and negotiations on topics of mobile gestures.

Figure 5.6 explains with an example how the different parts of the gesture notation can be used to define a gesture move. On the left side the various arrow illustrations show (a) a move gesture, (b) a swipe gesture, (c) a bidirectional swipe gesture and (d) a move gesture that should be carried out with two fingers. On the right side of Figure 5.6 6NEU a smartphone is depicted that is used in portrait mode, the bevelled edge on the sketch indicating the device's lower right corner. The move gesture is initiated in the upper right corner and ends in the lower half of the smartphones interactive area.

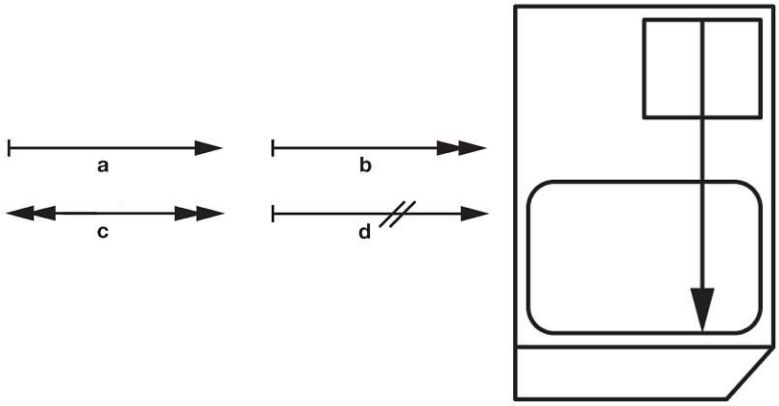


Figure 5.6: Monox gesture notation: examples of basic swipe and move gestures (left), one example of a complete defined move gestures on a smartphone in portrait mode (right)

Beside the moving gestures the notation consists of tapping gestures and multi touch gestures. These basic gesture can be combined to define well known gestures such as drag and drop. Among others the notation describes hardware buttons, sensor input, actuator output and the device's form factor. All single notations are envisioned to be unambiguous and open to extensions and mixing.

An extensive three-stage evaluation showed the practicability and ability to serve as a common denominator for discussion and communication within interdisciplinary groups of researchers, designers and developers.

CHAPTER 6

Scientific Contribution to the Field

This work contributes to the research area of multimedia interface and interaction design and the more general research area of human computer interaction in three ways. First, it investigates the work practices of modern television journalism and compares them with online video journalism. Second, it introduces, demonstrates and evaluates novel interaction techniques that allow a video editor to edit a video on a touch based mobile device. Third, drawing on the experience it presents an extensible gesture notation for touch based devices that can be used as a tool for discussing, implementing and evaluating interface and interaction designs.

6.1 Online Video Journalism

Existing research in CSCW and HCI covers work practices in the area of television journalism and online journalism quite sufficiently, however, there is little work done in the emerging field of online video journalism. A circumstance that is grounded more in the fact that there is just a small number of people working in that specific area so far than due to little relevance for the research community. During this thesis one person in television news broadcasting was shadowed during a regular working day to generate a basis to confirm the interviews upon. Additionally, eight people were interviewed working in the area of video news making, four in the „traditional“ area of television news broadcasting and four in the area of online video journalism. While the interviewees who work in television broadcasting agreed with the observations made during the shadowing in general, the interviewees working in online video journalism reported on noticeable differences (see also Chapter 2). The main findings are discussed in the following.

The most apparent difference was in the team size with television teams mainly working in groups of three to four persons which allows them to divide responsibilities within the

group. This allows every team member to concentrate on their task(s): directing; filming; sound recording; presenting; video editing. In contrast, online video journalists mainly work alone being director, director of photography, sound engineer, host and video editor in one person.

Another difference was seen in the workflow and the persons who are involved in the workflow. Television broadcaster schedule their TV program in advance which has two consequences. Team members (and possible replacements) must be organized beforehand and deadlines are predetermined due to the program schedule. Online video journalists differ from that as they do not need to organize team members and therefore they are comparably flexible. Furthermore, online media usually do not follow a time driven release schedule and can publish a news report whenever it is completed. This organizational freedom allows online video journalists to decide spontaneously, without prior coordination, whether or not a topic is relevant for coverage. As online video journalists work mostly alone they lack intra-group communication and intra-group feedback loops. Feedback from the online audience cannot replace professional feedback from colleagues.

While newspapers are traditionally associated with written text and television broadcasters to moving pictures, these boundaries begin to fade as online media provides all needed distribution channels and market players do not need expensive hardware such as a printing press or broadcast transmission stations. Thus, the media market faces a new situation as all market player equally rely on external service providers to publish or broadcast content. This makes the initial situation more equally as none of the market player has an advantage over the other due to experience in crucial infrastructure. Online video journalists said that they see their advantages in locality and time to market and, as their production overhead and costs are minimal they can even cover local stories that are too small to be considered by television broadcasters.

Television news journalists and online news journalists assume that the way news are produced and consumed will change over the next years in two ways. First, as mobile video consumption is rising they think that mobile video production will also rise over the next years and adapt to a mobile audience. This change has aesthetical implications as the screen of a mobile device is considerably smaller than a regular stationary television set, however, the change also has technical implications. While television journalists refer to truck-sized video studios for mobile news production, online news journalists mention laptops as their favorite tool for the same purpose. Online news journalists also were in favor of turning a smartphone from a consumption device into a production device allowing them to be even more spontaneous and to deliver their reports from the scene. Thus, it became evident during the interviews that feasible interface and interaction mechanisms for mobile video editing can turn out to be an useful tool for online video journalists.

The genre of online video journalists is a fertile area for research as little literature exists so far. Future research has various strands to follow. It can scrutinize the workflow and cooperative work practices focusing on the tools for inter-personal work. Especially interesting can be the lack of group dynamic during the production phase and the

consequences thereof. Another strand can be studying and comparing the content and aesthetical quality of online news reports and television news reports. However, a first step in supporting online video journalists can be in providing appropriate tools for mobile video editing, which is described in the following section.

6.2 Interactions for Mobile Video Editing

As stated above mobile video editing solutions promise to be an interesting tool for online video journalists. However, when searching in academic and patent databases for mobile video editing it became evident that current research focuses on automatic and collaborative video editing. In the following the contribution to contemporary research in the area of manual video editing on mobile devices are described.

Existing approaches in the field of mobile video editing on mobile devices with small screens focus mainly on algorithms to automate or semi-automate film compilation. These attempts aim to make filmmaking easier with the drawback of taking away some of the creativity an editor can contribute to a well drafted film narrative. This drawback may be of little concern when compiling personal or family video memories. However, if a film is intended to uphold some level of artistic or professional standard an editor needs control over the editing process. This thesis investigates the area of efficient and effective interaction mechanisms for advanced manual mobile video editing. It puts the mobile device and the users in the center of an user-centered design process trying to achieve the best possible outcome. Within the scope of the studies it contributed to three tasks vital for video editing: a) browsing media assets, b) trimming media assets and c) ordering media assets. The proposed work has shown that novel approaches for small touch-based mobile devices can help implement viable and useful interfaces and interaction techniques for browsing, trimming and ordering media assets. While existing interfaces for mobile video editing address either professionals or amateurs the work has shown that it is possible to implement complex interfaces and interaction techniques that are useable for both professionals and amateurs.

a) Browsing media assets: The proposed solution for fast and precise video browsing for mobile devices with small screens combines spatial and temporal components in a single interface and interaction design. In contrast to existing solutions it takes the *initial starting point* of a browsing gestures into account to determine the browsing speed. As the browsing speed is perpetually calculated depending on the *initial starting point* and the *current touch point* the proposed interface allows for a great variety of different browsing speeds within a limited interaction area. Due to the intuitive nature of the interface and the positive feedback during the evaluation, it is arguable that it is suitable for both professional video editors and casual video consumers. As all gestures can be carried out solely with the thumbs the device can be held steadily with both hands during browsing tasks, which was also pointed out positively during the evaluation (see Chapter 5, Paper 1).

b) Trimming media assets: While there is reasonable literature on the use of mobile

video editing interfaces and automated mobile video trimming, little research exists on actual implementations of novel mobile interfaces for trimming media assets manually. Therefore, this thesis proposes and contributes an alternative interface and interaction design for manual video trimming on mobile devices that does not solely rely on the timeline metaphor which is known from desktop video editing applications. The final layout consists of *long distance gestures* to *avoid ambiguity*. In a quantitative study, that simulated real world trimming tasks, the interface and interaction design proved to be a viable proposal as it was significantly faster than Apple's iMovie in six out of eight trimming tasks (see Chapter 5, Paper 3).

c) Ordering media assets: Ordering media assets consists of two independent steps: first, browsing a collection of media assets and second, re-ordering the sequence of the assets in the collection. There is a history of research in both areas whereby most of it targets desktop computers. Little research exists for mobile devices for small screens, however, interface and interaction design for smartphone like devices is hardly available. This thesis proposes two interfaces and interaction designs especially designed for smartphone devices targeting *browsing in a collection* (see Chapter 5, Paper 2) and *ordering assets in a collection* (see Chapter 5, Paper 4). Both proposals builds up on existing research and adapts established metaphors, such as focus+context and overview+detail, and both papers contribute to the research in the field by *adapting and adjusting* them for mobile gesture based interaction. The proposed interfaces allows an user to examine a thumbnail in detail (focus) while the rest of the thumbnails provide additional information (context) such as position in the thumbnail collection. The conducted evaluations state a satisfactory user experience also due to the utilization of familiar touch gestures.

In this thesis the vital tasks for video editing (browsing, trimming, ordering) were implemented and evaluated individually. Every task, and therefore every interface, is intended to work full screen and exploit every bit of available screen estate. Thus, when implementing an entire video editing suite, a user has to change between screens constantly. Further research should investigate the benefits and drawbacks of single screen and multiple screen applications for complex mobile applications. This includes cognitive load, completion time and overall user experience.

All proposed interfaces and interaction concepts presented in this thesis are designed as proofs-of-concept, even though they have been implemented and evaluated as closely as possible to conceivable real-world applications and tasks. Further work is required to foster real world applicability. The work explores the possibilities to map desktop interactions to touch based equivalents and while providing an enjoyable user experience. Even though the interfaces and interaction mechanics are novel for the mobile domain, the work mainly focuses on established features and tasks known from desktop video editing. Reconsidering the entire workflow of video editing for the mobile domain could offer a fruitful ground for further research.

In general, a lack of scientific research was discovered in the area of complex mobile interaction as current research mainly focuses on easy and intuitive interfaces and interaction for basic tasks. However, as mobile devices are steadily getting more powerful

this branch of interaction research will attract more attention from both industry and academic research.

6.3 Extensible Gesture Notations

Sketching is a favorable method for researchers and designers when developing and discussing novel ideas or proposals. In computer science various notations exist, for example the widely applied standardized unified markup language (UML). Such markup languages and notations are intended to provide a standard way to visualize the design of an idea and to provide a common ground for discussions. However, in the area of touch based interaction design such a notation is still missing. Regardless of whether researchers scrutinizing new approaches for mobile interactions or a team of developers exploring different variations of an interface design, researchers and designers use their very own notation or adapt one of the vendor provided notations. The main drawback of these approaches is the ambiguity that comes with ad-hoc purpose made notations.

When starting this thesis the author experienced a lack of suitable gesture notation to discuss interface designs and interaction mechanics as vendor specific notation systems are tied to specific platforms and tend to focus on graphical interface elements rather than on interaction. To fill this gap, interaction designers and researchers started to make their own compilations focusing on interaction rather than interface elements. However, the sketches were not precise in their meaning, were tricky to draw as they tended to be more artistic than accurate and, finally, were not extensible. Drawing on the experiences from discussing and designing the interface and interaction mechanics this work proposes an extensible gesture notation for touch based devices that focuses on unambiguity, interoperability, accuracy, and extensibility to mitigate the aforementioned detriments. The notation assists designers and researchers during the design phase when discussing interface and interaction ideas. In contrast to vendor specific notations and general collections of touch gestures, the notation is intended as an open alternative supporting the design process of multi-platform applications. Furthermore, the notation has a simplified depiction allowing it to easily be incorporated in paper sketches and interactive UI builders. Finally, the notations extensibility prepares it for future extensions and novel interactions ideas without altering the foundations of the notation. The proposed notation can help to discuss and pinpoint strengths and weaknesses in an interaction design for mobile applications. The findings show that the proposed notation can provide a common ground for discussing and exchanging ideas during all phases in a project: design; implementation; evaluation (see Chapter 5, Paper 5).

The basic gesture set and its extension rules proposed in this thesis aims to solve these shortcomings. The design and evaluation process of the proposed extensible gesture set involved three iteration cycles and incorporates eclectic input from various users. However, there is still place for further research. First, merging the interaction oriented gesture notation with an equally open and extensible interface notation is needed. As interaction mechanics often become clearer when presented combined with interface elements, this

would be a reasonable first enhancement to focus on. Second, dismantle and decompose the current gesture notation and find underlying basic gestures. Even though this has been done in the original work, an additional decomposition could turn out to be beneficial when integrating the notation in software products for rapid prototyping, such as in integrated development environments (IDEs). Furthermore, providing a software based framework of the notation for different IDE on diverse platforms could help to minimize the burden of cross-platform development.

List of Figures

Bibliography

Android Developers (2015). Supported media formats - video encoding recommendations. http://developer.android.com/guide/appendix/media-formats.html/. [Online; accessed 2-October-2016].

Appert, C. and Fekete, J.-D. (2006). Orthozoom scroller: 1d multi-scale navigation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '06, pages 21–30, New York, NY, USA. ACM.

Astell, A. (2008). The world's 50 most powerful blogs. https://www.theguardian.com/technology/2008/mar/09/blogs/. [Online; accessed 2-October-2016].

Atwood, M. E., McCain, K. W., and Williams, J. C. (2002). How does the design community think about design? In *Proceedings of the 4th Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques*, DIS '02, pages 125–132, New York, NY, USA. ACM.

Barnes, C., Goldman, D. B., Shechtman, E., and Finkelstein, A. (2010). Video tapestries with continuous temporal zoom. In *ACM SIGGRAPH 2010 Papers*, SIGGRAPH '10, pages 89:1–89:9, New York, NY, USA. ACM.

Barnes, S. B. (1997). Douglas Carl Engelbart: developing the underlying concepts for contemporary computing. *IEEE Annals of the History of Computing*, 19(3):16–26.

Baron, S. and Wood, D. (2005). Rec. 601 - the origins of the 4:2:2 DTV standard. ITU dimension. EBU Technical Review. https://tech.ebu.ch/docs/techreview/trev_304-rec601_wood.pdf/. [Online; accessed 2-October-2016].

Bartneck, C. (2009). Notes on Design and Science in the HCI Community. *Design Issues*, 25(2):46–61.

Benson, S. G. and Dundis, S. P. (2003). Understanding and motivating health care employees: integrating Maslow's hierarchy of needs, training and technology. *Journal of nursing management*, 11(5):315–320.

Bläsi, B. (2004). Peace journalism and the news production process. *Conflict & communication online*, 3(1/2):1–12.

Blu-ray Disc Association (2010). White paper Blu-ray Disc Read-Only Format - 2.B Audio Visual Application Format Specifications for BD-ROM Version 2.4. http://www.blu-raydisc.com/assets/Downloadablefile/BD-ROM_Audio_Visual_Application_Format_Specifications-18780.pdf/. [Online; accessed 2-October-2016].

Bowman, S. and Willis, C. (2003). We Media - How audiences are shaping the future of news and information. http://www.hypergene.net/wemedia/download/we_media.pdf/. [Online; accessed 2-October-2016].

Browne, S. (1998). *Nonlinear Editing Basics: A Primer on Electronic Film and Video Editing*. Focal Press.

Bruns, A. and Highfield, T. (2015). *18. From news blogs to news on Twitter: gatewatching and collaborative news curation*. Edward Elgar Publishing.

Buchanan, R. (1992). Wicked problems in design thinking. *Design Issues*, 8(2):5–21.

Buck, J. (2011). *Timeline 2 - A History of Editing*. BookBaby.

Burger, C. (2014). Die Raute als Kennzeichen für User Generated Content. http://derstandard.at/2000002823997/Die-Raute-als-Kennzeichen-fuer-User-Generated-Content/. [Online; accessed 2-October-2016].

Bush, V. (1945). As We May Think. *Life Magazine*, (10):112–124.

Butter, A. and Pogue, D. (2002). *Piloting Palm: The inside story of Palm, Handspring, and the birth of the billion-dollar handheld industry*. John Wiley & Sons.

Buxton, B. (2010). *Sketching user experiences: getting the design right and the right design*. Morgan Kaufmann.

Card, S. K., Moran, T. P., and Newell, A. (1980). The Keystroke-level Model for User Performance Time with Interactive Systems. *Commun. ACM*, 23(7):396–410.

Card, S. K., Newell, A., and Moran, T. P. (1983). *The psychology of human-computer interaction*. L. Erlbaum Associates Inc.

Chaffey, D. (2016). Mobile Marketing Statistics compilation. http://www.smartinsights.com/mobile-marketing/mobile-marketing-analytics/mobile-marketing-statistics/. [Online; accessed 2-October-2016].

Cisco Corporation (2016). Cisco Visual Networking Index: Forecast and Methodology, 2015–2020, White Paper. http://www.cisco.com/c/dam/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.pdf/. [Online; accessed 2-October-2016].

Cockburn, A., Karlson, A., and Bederson, B. B. (2009). A Review of Overview+Detail, Zooming, and Focus+Context Interfaces. *ACM Comput. Surv.*, 41(1):2:1–2:31.

Cohen, M. and Brodlie, K. (2004). Focus and context for volume visualization. In *Theory and Practice of Computer Graphics, 2004. Proceedings*, pages 32–39.

Courtney, A. and Courtney, M. (2008). Comments regarding "on the nature of science". *arXiv preprint arXiv:0812.4932*.

Csikszentmihalyi, M. (2008). *FLOW - The Psychology of Optimal Experience*. Harper Perennial Modern Classics.

Daley, J. (1982). Design creativity and the understanding of objects. *Design Studies*, 3(3):133–137.

Damasio, A. R. (2014). *Descartes' Irrtum: Fühlen, Denken und das menschliche Gehirn*. Ullstein eBooks.

Davenport, G., Smith, T. A., and Pincever, N. (1991). Cinematic primitives for multimedia. *IEEE Computer Graphics and Applications*, 11(4):67–74.

Diekmann, A. (2007). Empirische Sozialforschung: Grundlagen, Methoden, Anwendungen (18. Aufl.). *Reinbek: Rowohlt*.

Dorst, K. and Cross, N. (2001). Creativity in the design process: co-evolution of problem–solution. *Design Studies*, 22(5):425–437.

DRadio Wissen (2014). Endlich Fernsehen. `http://dradiowissen.de/beitrag/zeitmaschine-stichtag-14-dezember/`. [Online; accessed 2-October-2016].

Dubberly, H. (2007). The Making of Knowledge Navigator. `http://www.dubberly.com/articles/the-making-of-knowledge-navigator.html/`. [Online; accessed 2-October-2016].

El Batran, K. and Dunlop, M. D. (2014). Enhancing KLM (Keystroke-level Model) to Fit Touch Screen Mobile Devices. In *Proceedings of the 16th International Conference on Human-computer Interaction with Mobile Devices & Services*, MobileHCI '14, pages 283–286, New York, NY, USA. ACM.

Flick, U. (1995). *Qualitative Forschung: Theorie, Methoden, Anwendung in Psychologie und Sozialwissenschaften*. Rowohlt.

Furnas, G. W. (1986). Generalized Fisheye Views. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '86, pages 16–23, New York, NY, USA. ACM.

Gentner, D. and Nielsen, J. (1996). The Anti-Mac Interface. *Commun. ACM*, 39(8):70–82.

Gentner, D. R. and Grudin, J. (1990). Why Good Engineers (Sometimes) Create Bad Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '90, pages 277–282, New York, NY, USA. ACM.

Gigerenzer, G. and Todd, P. M. (1999). *Simple heuristics that make us smart*. Oxford University Press, USA.

Gittens, M., Kim, Y., and Godwin, D. (2005). The vital few versus the trivial many: examining the pareto principle for software. In *29th Annual International Computer Software and Applications Conference (COMPSAC'05)*, volume 1, pages 179–185 Vol. 2.

Goldman, D. (2007). A framework for video annotation, visualization, and interaction. Doctoral dissertation, University of Washington.

Goldschmidt, G. (1992). Serial sketching: visual problem solving in designing. *Cybernetics and System*, 23(2):191–219.

Gray, W. D., John, B. E., and Atwood, M. E. (1992). The precis of project ernestine or an overview of a validation of goms. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '92, pages 307–312, New York, NY, USA. ACM.

Han, J. (2011). A goms-based granular computing model for human-computer interaction design. In *Service Operations, Logistics, and Informatics (SOLI), 2011 IEEE International Conference on*, pages 243–248.

Hardman, L. (2005). Canonical processes of media production. In *Proceedings of the ACM Workshop on Multimedia for Human Communication: From Capture to Convey*, MHC '05, pages 1–6, New York, NY, USA. ACM.

Hassenzahl, M. and Tractinsky, N. (2006). User experience - a research agenda. *Behaviour & Information Technology*, 25(2):91–97.

Heinrich, L. J., Heinzl, A., and Roithmayr, F. (2007). *Wirtschaftsinformatik: Einführung und Grundlegung*. Oldenbourg.

Heinrich, L. J. and Häntschel, I. (1999). *Evaluation und Evaluationsforschung in der Wirtschaftsinformatik: Handbuch für Praxis, Lehre und Forschung*. Oldenbourg.

Hermida, A. (2010). From TV to Twitter: How ambient news became ambient journalism. *Media/Culture Journal*, 13(2).

Holleis, P., Otto, F., Hussmann, H., and Schmidt, A. (2007). Keystroke-level Model for Advanced Mobile Phone Interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '07, pages 1505–1514, New York, NY, USA. ACM.

Hürst, W. and Darzentas, D. (2012). History: A hierarchical storyboard interface design for video browsing on mobile devices. In *Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia*, MUM '12, pages 17:1–17:4, New York, NY, USA. ACM.

Hürst, W., Götz, G., and Welte, M. (2007). Interactive video browsing on mobile devices. In *Proceedings of the 15th ACM International Conference on Multimedia*, MM '07, pages 247–256, New York, NY, USA. ACM.

Hürst, W., Snoek, C. G., Spoel, W.-J., and Tomin, M. (2010). Keep Moving!: Revisiting Thumbnails for Mobile Video Retrieval. In *Proceedings of the 18th ACM International Conference on Multimedia*, MM '10, pages 963–966, New York, NY, USA. ACM.

Hürst, W., Götz, G., and Lauer, T. (2004). New methods for visual information seeking through video browsing. In *Proceedings of the 8th International Conference on Information Visualisation*, pages 450–455.

ISO (1998). International Organisation for Standardisation 9241-11:1998.

Jackson, D. and Olivier, P. (2012). Panopticon: A Parallel Video Overview Technique.

Jackson, F. (1993). Design for the real world: Human ecology and social change. *Journal of design history*, 6(4):307–310.

Johnson, J., Roberts, T. L., Verplank, W., Smith, D. C., Irby, C. H., Beard, M., and Mackey, K. (1989). The Xerox Star: A Retrospective. *Computer*, 22(9):11–26.

Jokela, T., Karukka, M., and Mäkelä, K. (2007). Mobile video editor: design and evaluation. In *International Conference on Human-Computer Interaction*, pages 344–353. Springer.

Jonas, W. (2012). Zufälle, Marotten, Eigentum - die anderen Gründe von Designtheorie. Oder: Alles Gesagte wird von jemanden gesagt. Fiasco - ma non troppo! In *21. Designtheoretisches Symposion der Burg Giebichenstein*. Kunsthochschule Halle.

Jones, J. C. (1992). *Design methods: seeds of human futures*. John Wiley & Sons.

Kay, A. C. (1972). A personal computer for children of all ages. In *Proceedings of the ACM Annual Conference - Volume 1*, ACM '72, New York, NY, USA. ACM.

Khandkar, S. H. (2010). *A domain-specific language for multi-touch gestures*. PhD thesis, University of Calgary.

Kirsner, S. (2008). *Inventing the Movies: Hollywood's Epic Battle between Innovation and the Status Quo, from Thomas Edison to Steve Jobs*. Scott Kirsner.

Knör, R. and Driesnack, D. (2009). Untersuchungen zu Kompression und Konvertierung bei HDTV. `http://www.irt.de/webarchiv/showdoc.php?z=MzM5MCMxMDA2MDE2MTAjcGRm/`. [Online; accessed 2-October-2016].

Kraus, D. (2014). Auf Augenhöhe: Wie US-Broadcaster sich mit ihrem Publikum verbinden. http://derstandard.at/2000001750462/Auf-Augenhoehe-Wie-US-Broadcaster-sich-mit-ihrem-Publikum-verbinden/. [Online; accessed 2-October-2016].

Krippendorff, K. (2007). Design research, an oxymoron? *Design research now*, pages 67–80.

Kurosu, M. and Kashimura, K. (1995). Apparent usability vs. inherent usability: Experimental analysis on the determinants of the apparent usability. In *Conference Companion on Human Factors in Computing Systems*, CHI '95, pages 292–293, New York, NY, USA. ACM.

Larman, C. and Basili, V. R. (2003). Iterative and Incremental Developments. A Brief History. *Computer*, 36(6):47–56.

Lawson, B. (2006). *How designers think: The design process demystified*. Routledge.

Lee, P., Stewart, D., and Calugar-Pop, C. (2016). Technology, Media & Telecommunications Predictions 2016. http://www2.deloitte.com/content/dam/Deloitte/global/Documents/Technology-Media-Telecommunications/gx-tmt-prediction-2016-full-report.pdf/. [Online; accessed 2-October-2016].

Lewis, P. (1993). Of Zoomers, Newtons and Real Life: So Far, Promise Exceeds Usefulness. http://www.nytimes.com/1993/11/09/science/personal-computers-zoomers-newtons-real-life-so-far-promise-exceeds-usefulness.html/. [Online; accessed 2-October-2016].

Lidwell, W., Holden, K., and Jill, B. (2007). *Universal principles of design: 125 ways to enhance usability, influence perception, increase appeal, make better design decisions, and teach through design*. Rockport.

Lindstedt, I., Löwgren, J., Reimer, B., and Topgaard, R. (2009). Nonlinear News Production and Consumption: A Collaborative Approach. *Comput. Entertain.*, 7(3):42:1–42:17.

Luff, J. (2007). Musings of a Consultant. *Society of Broadcast Engineers. Pittsburgh Chapter*, 15(4):3.

Maher, M. L., Poon, J., and Boulanger, S. (1996). Formalising design exploration as co-evolution. In *Advances in formal design methods for CAD*, pages 3–30. Springer.

Marr, B. (2015). Big Data: 20 Mind-Boggling Facts Everyone Must Read. *Forbes Magazine*.

Masui, T., Kashiwagi, K., and Borden, IV, G. R. (1995). Elastic graphical interfaces to precise data manipulation. In *Conference Companion on Human Factors in Computing Systems*, CHI '95, pages 143–144, New York, NY, USA. ACM.

Matejka, J., Grossman, T., and Fitzmaurice, G. (2013). Swifter: Improved online video scrubbing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 1159–1168, New York, NY, USA. ACM.

Mirriam Webster (2016). `https://tech.ebu.ch/docs/techreview/trev_304-rec601_wood.pdf/`. [Online; accessed 2-October-2016].

Myers, B. A. (1998). A brief history of human-computer interaction technology. *interactions*, 5(2):44–54.

Nguyen, C., Niu, Y., and Liu, F. (2012). Video summagator: An interface for video summarization and navigation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 647–650, New York, NY, USA. ACM.

Nilsson, E. G. (2009). Design patterns for user interface for mobile applications. *Advances in Engineering Software*, 40(12):1318–1328.

Norman, D. A. (1981). Categorization of action slips. *Psychological review*, 88(1):1.

Norman, D. A. (2013). *The design of everyday things: Revised and expanded edition*. Basic books.

Odlyzko, A. (1999). The visible problems of the invisible computer: A skeptical look at information appliances. *First Monday*, 4(9).

Park, W., Han, S. H., Kang, S., Park, Y. S., and Chun, J. (2011). A factor combination approach to developing style guides for mobile phone user interface. *International Journal of Industrial Ergonomics*, 41(5):536–545.

Petruschat, J. (2011). Wicked Problems. `http://www.petruschat.dlab-dd.de/Petruschat/Wicked_Problems.html/`. [Online; accessed 2-October-2016].

Ramos, G. and Balakrishnan, R. (2003). Fluid interaction techniques for the control and annotation of digital video. In *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology*, UIST '03, pages 105–114, New York, NY, USA. ACM.

Ramos, G. and Balakrishnan, R. (2005). Zliding: Fluid zooming and sliding for high precision parameter manipulation. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology*, UIST '05, pages 143–152, New York, NY, USA. ACM.

Rice, A. D. and Lartigue, J. W. (2014). Touch-level model (tlm): Evolving klm-goms for touchscreen and mobile devices. In *Proceedings of the 2014 ACM Southeast Regional Conference*, ACM SE '14, pages 53:1–53:6, New York, NY, USA. ACM.

Richards, M. (2008). Why the iphone makes 2008 seem like 1968 all over again. *The Open University, Facuty of Mathematics, Computing and Technology, Computing Department*.

Richer, M. S., Reitmeier, G., Gurley, T., Jones, G. A., Whitaker, J., and Rast, R. (2006). The atsc digital television system. *Proceedings of the IEEE*, 94(1):37–43.

Rittel, H. W. and Webber, M. M. (1973). Dilemmas in a general theory of planning. *Policy sciences*, 4(2):155–169.

Roeder, J., Brecht, R., and Kunert, T. (2006). Effective production of television content for mobile devices. In *2006 IEEE International Symposium on Consumer Electronics*, pages 1–5.

Roth, V. and Turner, T. (2009). Bezel swipe: Conflict-free scrolling and multiple selection on mobile touch screen devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, pages 1523–1526, New York, NY, USA. ACM.

Royce, W. W. (1970). Managing the development of large software systems. In *proceedings of IEEE WESCON*, volume 26, pages 328–338. Los Angeles.

Rubin, M. (2000). *Nonlinear-A Field Guide to Digital Video and Film Editing*. Triad Publishing Company.

Saffer, D. (2010). *Designing for interaction: creating innovative applications and devices*. New Riders.

Sager, I. (2012). Before IPhone and Android Came Simon, the First Smartphone. `http://www.bloomberg.com/news/articles/2012-06-29/before-iphone-and-android-came-simon-the-first-smartphone/`. [Online; accessed 2-October-2016].

Schön, D. A. (1983). *The reflective practitioner: How professionals think in action*, volume 5126. Basic books.

Schulz, T. (2008). Using the keystroke-level model to evaluate mobile phones. *Proceedings of the 31st Information Systems Research Seminaria-IRIS*, 31.

Seon-Hwi, C. (2013). Mobile terminal using proximity touch and wallpaper controlling method thereof. US Patent 8,576,181.

Spence, R. and Apperley, M. (1982). Data base navigation: an office environment for the professional. *Behaviour & Information Technology*, 1(1):43–54.

Staggers, N. and Norcio, A. F. (1993). Mental models: concepts for human-computer interaction research. *International Journal of Man-machine studies*, 38(4):587–605.

Stolterman, E. (2008). The nature of design practice and implications for interaction design research. *International Journal of Design*, 2(1).

Styria Media (2015). Bezirksblatt / Mein Bezirk. http://www.meinbezirk.at/registrieren//.

Sullivan, L. H. (1956). *The autobiography of an idea*, volume 281. Courier Corporation.

Teich, S. T. and Faddoul, F. F. (2013). Lean management—the journey from toyota to healthcare. *Rambam Maimonides Medical Journal*, 4(2).

The Open University (2014). 2014 Text Messaging Usage Statistics. http://www.openuniversity.edu/news/news/2014-text-messaging-usage-statistics/. [Online; accessed 2-October-2016].

Ursu, M. F., Thomas, M., Kegel, I., Williams, D., Tuomola, M., Lindstedt, I., Wright, T., Leurdijk, A., Zsombori, V., Sussner, J., Myrestam, U., and Hall, N. (2008). Interactive tv narratives: Opportunities, progress, and challenges. *ACM Trans. Multimedia Comput. Commun. Appl.*, 4(4):25:1–25:39.

U.S. Department of Health & Human Services (2016). User-Centered Design Basics. https://www.usability.gov/what-and-why/user-centered-design.html/. [Online; accessed 2-October-2016].

Väätäjä, H. (2010). User experience evaluation criteria for mobile news making technology: Findings from a case study. In *Proceedings of the 22Nd Conference of the Computer-Human Interaction Special Interest Group of Australia on Computer-Human Interaction*, OZCHI '10, pages 152–159, New York, NY, USA. ACM.

Väätäjä, H. and Männistö, A. A. (2010). Bottlenecks, usability issues and development needs in creating and delivering news videos with smart phones. In *Proceedings of the 3rd Workshop on Mobile Video Delivery*, MoViD '10, pages 45–50, New York, NY, USA. ACM.

van den Boom, H. (2011). *Das Designprinzip: warum wir in der Ära des Designs leben*. kassel university press GmbH.

Van Every, S. (2004). Interactive Tele-journalism: Low Cost, Live, Interactive Television News Production. In *Proceedings of the 12th Annual ACM International Conference on Multimedia*, MULTIMEDIA '04, pages 170–171, New York, NY, USA. ACM.

Vihavainen, S., Mate, S., Seppälä, L., Cricri, F., and Curcio, I. D. (2011). We want more: Human-computer collaboration in mobile social video remixing of music concerts. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 287–296, New York, NY, USA. ACM.

Warner, W. (1988). Business Plan July 1988. https://archive.org/details/AvidBusinessPlanJuly1988WOCR/. [Online; accessed 2-October-2016].

Weinstein, A. (2002). Customer retention: A usage segmentation and customer value approach. *Journal of Targeting, Measurement and Analysis for Marketing*, 10(3):259–268.

Wertheimer, M. (1912). *Experimentelle Studien über das Sehen von Bewegung*. JA Barth.

Wiggins, R. H. (2004). Personal digital assistants. *Journal of Digital Imaging*, 17(1):5–17.

Wikipedia (2016). VisualEditor. `https://www.mediawiki.org/wiki/VisualEditor/`. [Online; accessed 2-October-2016].

Woyke, E. (2014). *The Smartphone: Anatomy of an industry*. The New Press.

Wroblewski, L. (2010). Touch Gesture Reference Guide. `http://www.lukew.com/ff/entry.asp?1071/`. [Online; accessed 2-October-2016].

Appendix - Papers

ProPane: Fast and Precise Video Browsing on Mobile Phones

Roman Ganhör
Vienna University of Technology
Favoritenstrasse 9-11/187
1040 Vienna, Austria
roman.ganhoer@tuwien.ac.at

ABSTRACT

Studies show that every fourth smartphone user watches videos on their device. However, because of increasing camera and encoding quality more and more smartphones are providing an attractive tool for creating and editing videos. The demand for smooth video browsing interfaces is challenged by the limited input and output capabilities that such mobile devices offer. This paper discusses a novel interface for fast and precise video browsing suitable for watching and editing videos. The browsing mechanism offers a simple but powerful interface for browsing videos at different levels of granularity. All interactions can be carried out with no modal changes at all. The interface is easy to understand and efficient to use. A first evaluation proves the suitability of the presented design for casual users as well as for creative professionals such as video editors.

Categories and Subject Descriptors

H.5.2 [**Information Interfaces and Presentation (e.g. HCI)**]: User Interfaces – *Graphical user interfaces (GUI), input devices and strategies, interaction styles, screen design*

General Terms

Design, Experimentations, Human Factors

Keywords

handheld devices, mobile video, video browsing, interface design, multimedia, navigation, precise, touch, Android

1. INTRODUCTION

Recent studies by Cisco about the mobile internet traffic and video consumption predict an increase of the worldwide mobile internet traffic from 1,2 PB/Month in 2012 up to 10,8 PB/Month in 2016 [4]. The study states that by the end of 2012 more than half of the internet traffic generated by consumers will be video based. It is apparent that video already is playing a key role and it will keep this key role.

MUM '12, December 04 - 06 2012, Ulm, Germany.

Today one out of four smartphone users consumes videos on their devices. Younger users tend to watch more than older users [28]. This passive usage is complemented by an increasing group of people utilizing their mobile phones for recording videos for leisure purposes. Teenage users in particular tend to take a very lightweight approach to mobile video. They often just capture short video clips with their mobile phone or other multifunction devices, and share these clips momentarily on the mobile screen or via Bluetooth, or later on websites, such as YouTube [27]. Capturing video has become very easy and happens spontaneously with mobile phones, but the resulting clips are rarely edited, mostly because this would imply a different physical and social context [27] or because it is simply too difficult [16,33].

On a professional level an increasing number of news media is incorporating the audience into their news creation processes and utilizing multimedia material created by the audience as news content, such as CNN and CBS. Furthermore, some news agencies, such as Adresseavisen in Norway and Reuters, have experimented with, or adopted, smartphones as tools of professionals for news making. Väätäjä et.al. also mentions the usefulness of mobile video for online publications. In a professional context such as video-journalism a mobile phone is the device of choice for quick from-the-scene-reportage [29,32].

Regardless of background - amateur or professional - recorded video footage normally runs through a process called post-production where video editing is applied [5]. Väätäjä et.al. stated that one of the minimum requirements of a mobile video editor should be the ability to cut a video from beginning and end [30]. Shortening clips and removing irrelevant material from clips is important to make a clip fulfill filmography-standards or just to make a clip which is too long shorter [16,19].

Despite usage (passive watching or active editing) or the background (amateur or professional) a recurrent task is browsing through a video forward and backward, with different speed to reach a specific frame or to watch a given sequence again [21].

This would suggest that if we were to design tools to better support such users and tasks, then design goals would include the following issues:

- the interface can be used for watching a video and editing a video
- all interaction is done with/on the touchscreen - no physical buttons are needed
- short response times
- tight holding - the interface works in the mobile context like a car-ride or train-ride

According to the design goals the research questions are the following:

Is it possible to browse videos on a mobile device fast and frame-accurate? What does a fast, stable, productive and easy to use interface for frame-accurate video browsing on mobile phones look like? Is such an interface feasible for both amateurs and professional videographers? Would users like to incorporate such an interface to their favorite video player or video-editing suite?

In this paper *ProPane* is presented as a new video-browsing interface for touchscreen-based handheld mobile devices. The next section gives a brief overview of the basics of time-based media in general and of relevant work for video browsing in the mobile context especially, followed by a discussion of the design of the system and a user evaluation.

2. BACKGROUND AND RELATED WORK

One of the main challenging issues concerning video browsing is the temporal nature of the media. Browsing mechanism suitable for static content like pictures or written documents are not always transferable to time based media like video [3,10]. Furthermore, browsing mechanisms and interfaces that are working satisfactorily on desktop applications do not automatically work equally well on small screens offered by mobile phones.

The following subsections discuss the basics of video and video browsing in the mobile domain.

2.1 Frames, Frame-rate, Browsing

Watching videos on mobile phones normally means looking at 25 or 30 pictures (frames) every second [18]. The number of pictures is called frame-rate and is notated in frames per seconds (fps). A quick query on gsmarena.com searching for mobile phones capable of shooting high definition video footage found 249 phones using 30fps and 14 phones using 25fps. A similar query on amazon.com found 119 phones using 30fps and 50 phones using 25fps. To keep it simple we stay with 30fps. Hence a video clip of 1 minute length has - 60 seconds with 30 frames a seconds - 1800 single frames.

Browsing through the clip can be done at different speed levels: real-time (30 fps), slow motion (<30 fps) or fast forward (>30fps). Fast forward at double speed has a frame rate of 60fps, slow motion at half speed has 15fps accordingly.

Different strategies can be used for navigation through clips. These includes fast forward to move close to an arbitrary position within the clip. Slow motion or even frame-by-frame navigation is used when looking for a specific frame. It has to be noted that real-time, fast forward and slow motion can be used in both directions forward and backward.

2.2 Slider

In contrast to tape based video, modern file based video allows the user to jump to any position in the video without spooling through the whole video. A slider is one of the easiest metaphors for such video browsing (Figure 1).

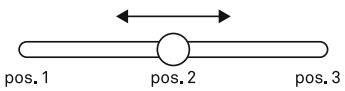


Figure 1: Slider

The slider in its length represents the whole video clip. Typing/Padding on the very left side of the slider (marked as pos.1) lets the video jump to the beginning of the video. Typing/Padding on the very right of the slider (pos.3) lets the video jump to the end of the video. Typing/Padding somewhere on the slider lets the video jump to the respective frame in the video (pos. 2).

By contrast, exact positioning and navigation on a frame-by-frame basis is hard and most of the time not possible with a slider. For example a given video has a length of 1 minute (1800 frames) and the width of a given mobile phone is 800 pixels. Moving the slider one pixel - which is hard enough to achieve - moves the position within the video by 2 frames. Technically speaking, the size of the slider is restricted to the screen size and therefore cannot scale to long video clips. Multi-scale timeline slider was one approach to solve that problem [25].

2.3 Mobile Zoom Slider

Hürst et.al introduced another version of such a multi-scaled slider for the mobile domain [12,13,14]. Pointing anywhere on the screen followed by a left or right movement results in a backward or forward navigation in the video clip. Doing so on top of the screen, the navigation in the video is on a frame-by-frame basis; doing so on the bottom of the screen, the navigation in the video is like with an ordinary slider (Figure 2: Mobile Zoom Slider).

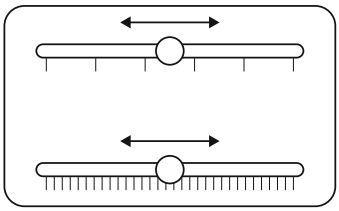


Figure 2: Mobile Zoom Slider

In between is a linear interpolation. The main advantages of the Mobile Zoom Slider are that a) different granularity levels are available and b) an interaction can start anywhere.

On the down side, the Mobile Zoom Slider is optimized for pen interaction. Using the Mobile Zoom Slider with your thumbs occludes big portions of the content and is therefore not practical. It is also hard to stay on a given granularity and speed. To overcome the last limitation Hürst et.al added a speed based area to their browsing concept [12]. Such a speed based area is located on the very right and very left of the screen and allows the user to skim forward and backward on different speed levels constantly and continuously. This makes it easy to stay on a given speed level (Figure 3).

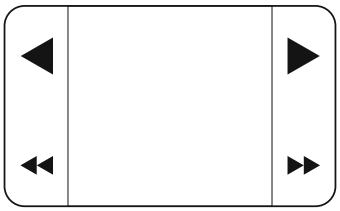


Figure 3: Speed Based Area

ProPane, the navigation interface presented in this paper, is based on the idea of the speed based area and extends it with a concept known from wheel based navigation.

2.4 The Wheel

A well known implementation of a wheel based navigation is the touch wheel used by the iPod (Figure 4: right side). Browsing forward and backward can be done by moving the finger clockwise and counter-clockwise on a circular touch sensitive

interface. The longer the gesture is carried out the faster the browsing will be.

Another widely used implementation of a wheel based navigation is the jog/shuttle (Figure 4: left side). The jog/shuttle wheel allows users to control the browsing interaction with two elements: the inner jog ring for precise browsing and the outer shuttle ring for fast browsing. The inner jog works similar to the touch wheel presented above. The outer shuttle ring is a rate control consisting of a spring-loaded ring located around the jog. The play rate increases in the forward direction as it is rotated clockwise, and backward when rotated counter-clockwise. When released, the ring "snaps" back to its original position [20].

Figure 4: Jog/Shuttle and Touch Wheel [20]

Touch wheel and jog/shuttle are good examples for the following design guidelines a) self-centering mechanisms are an important characteristic for rate controls and b) incorporating an acceleration function into a position control significantly improves scrolling performance [22,7].

ProPane incorporates these browsing design guidelines not on a circular but linear basis.

2.5 Guidelines for Mobile Interfaces

In contrast to traditional desktop computers mobile devices differ significantly in [12]:

- Performance. Despite recent improvements, performance remains a big issue for mobile devices: Low battery life prohibits high clock rates; design and size of the devices make heat transmission difficult; limitations exist, e.g., in the achievable frame-rate due to restricted processor power and memory.
- Input devices. Keyboard, mouse, and touchpad are predominant input devices on PCs and laptops. In contrast to this, handheld devices are usually operated by much fewer mechanisms; buttons (e.g. cell-phones), a pen (e.g. PDAs) or fingers (e.g. smartphones).
- Screen size. The terms "mobile" and "handheld" imply that there is (and always will be) a natural limit for the screen size, i.e. the area in which to represent the content and additional interface elements for direct interaction.

Guidelines exist for every major mobile platform [1,24]. The main statements from these guidlines are to 1) focus on the user's content 2) reduce complexity without diluting capability 3) provide shortcuts that empower and delight 4) adapt familiar hallmarks of the desktop experience and 5) set the viewport appropriately for the device.

Jokela et.al. also states avoiding modal-changes (screen changes) on mobile devices [17]. Huber et.al. explored the design space and the characteristics of interaction concepts for mobile video browsing and concluded the importance of the thumb for mobile interaction interfaces [11].

All browsing interfaces and design guidelines presented in this section should be considered when implementing *ProPane*.

3. IMPLEMENTATION OF PROPANE

In this section we introduce the idea of *ProPane* and how we implemented *ProPane*. Furthermore the methodology used during implementation is presented.

The motivation for *ProPane* was to develop an interface for browsing video streams using just a small set of gestures. The interface allows users to start, stop and browse to any arbitrary point in the video clip in a fast and precise way. Work done by other researchers, plus common and browsing specific guidelines, should be incorporated.

3.1 Premises and Methods

One design goal was to maximize the usability of the interaction design "for the wild". This implies that users hold the mobile phone in their hands during browsing and do not use a table or some other "fixation". While holding a smartphone the thumbs are limited to some portion of the screen. The transition between coarse and fine browsing should be seamless. The browsing interface should be effective and efficient to use.

The development is based on the ideas of participatory design and contextual design [2,8]. At the beginning interviews and ethnographic observation was done with professional videographers to determine their goals and needs. Various design visions were implemented as low-fi prototypes and evaluated by a small group of four people. The evaluation sorted out ideas and introduced new ones. This short cycle of low-fi prototype and evaluation was done several times. After the fourth iteration the number of problematic and unclear interaction tasks went to zero. The gathered ideas were implemented as a high-fi prototype. There were another two evaluation cycles with high-fi prototypes to finish the design of *ProPane*.

3.2 Basic Layout

The basic layout of the screen includes several distinctive areas (Figure 5). Pane A on the left is the interaction area for browsing backward in the video stream. Pane B on the right is the interaction area for browsing forward in the video stream. Pane C (video pane) shows the video. Pane D holds a regular slider and an information bar.

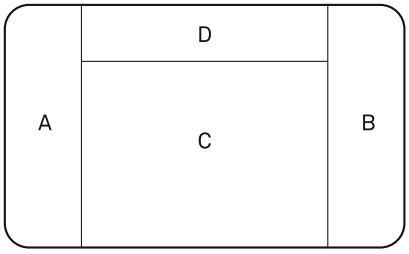


Figure 5 : Interface Panes

However, while we acknowledge that the potential exists to make smaller the interaction elements, this was not the focus of this paper.

3.3 Standard Browsing

Standard browsing implements fast forward and fast backward capabilities to the presented basic layout.

Figure 6 gives a detailed picture of the standard browsing scheme. If B1 is clicked once, the video jumps forth one frame. If B1 is pressed longer, the video jumps forward one frame by one frame. After one second continuously pressing B1, the video starts

playing forward in real-time (30fps). The moment the thumb is lifted from B1, the video stops.

If B1 is pressed and the thumb moves to B2, the playing speed is doubled to (60fps). At B3 the playing speed is quadrupled (120fps). Holding the thumb at B3 increases the frame-rate to 20x real-time (600fps) within 3 seconds. Moving the thumb back to B1 lets the video play at normal speed immediately. Lifting the thumb at any time stops playing the video.

Pane A with A1, A2 and A3 implements the same functionality but backwards.

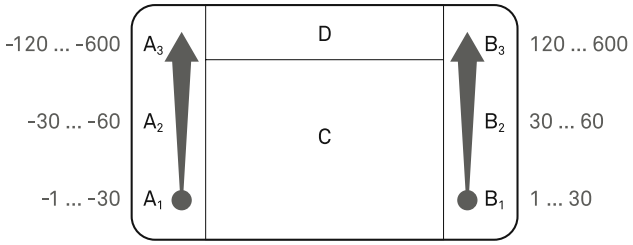


Figure 6 : Standard Browsing with Frame-rate

Area D (slider area) shows a slider giving feedback about which position in the clip the user is at.

This interaction schema is called standard browsing and was implemented as described. Standard browsing is feasible for most browsing tasks. Users can browse forward and backward at various granularity levels. Speed changes can be accomplished seamlessly. And it is possible to browse to a specific frame using frame-by-frame navigation.

However, standard browsing misses one important browsing concept - continuous slow motion. Slow motion will be added in the following scheme, the advanced browsing.

3.4 Advanced Browsing

Advanced browsing is very similar to standard browsing and adds slow motion capabilities to standard browsing/basic layout. The interaction scheme of standard browsing remains. Instead of one starting point on the bottom of pane A and B the browsing is initiated at the top of pane A and B as well (Figure 7).

Initially clicking on B3 jumps forward one frame in the video clip. Holding B3 for one second lets the video play forward in real-time (30fps). Pressing B3 and moving the thumb to B2 slows the playing speed down to the half (15fps) immediately. Pressing B3 and moving the thumb to B1 slows the playing speed down to a fifth (6fps) immediately. Moving the thumb back to the starting point B3 increases the playback speed to real-time (30fps). Lifting the thumb at any time immediately stops the playback of the video without further consequences. Pane A with A1, A2 and A3 implements the same functionality but backwards.

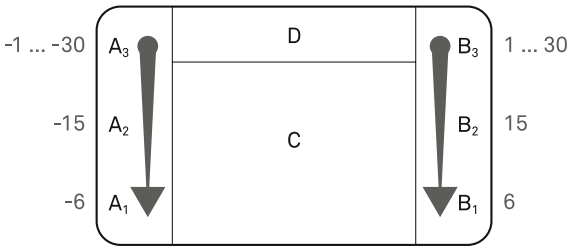


Figure 7: Advanced Browsing - initial start on top

3.5 Progressive Browsing

Progressive browsing combines standard browsing (fast forward) and advanced browsing (slow motion) in one gesture. This allows the user to change between fast forward and slow motion with one gesture. Instead of starting the gesture on the bottom (fast forward) or on the top (slow motion) of pane A and B progressive browsing gesture is initiated in the middle of pane A and B (Figure 8).

Initially clicking on B2 jumps forward one frame in the video clip. Holding B2 for one second lets the video play forward in real-time (30fps). Pressing B2 and moving the thumb to B3 speeds up the playback to the double (60fps). Moving the thumb to B1 slows the playing speed down to the half (15fps). Lifting the thumb at any time stops the playback of the video immediately. Pane A with A1, A2 and A3 implements the same functionality but backwards.

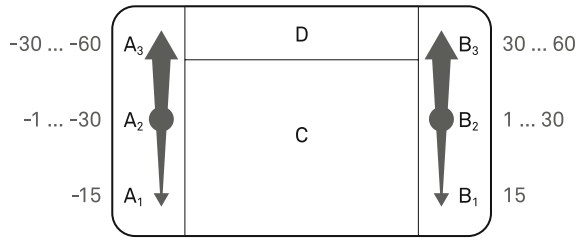


Figure 8: advanced browsing - initial start in the middle

Together, standard browsing, advanced browsing and progressive browsing allow fast and accurate video browsing in both directions. The browsing speed and so the frame-rate can be changed easily and quickly within a wide range. Various frame-rates (6fps, 15fps, 30fps, 60fps, 600fps) can be locked. No browsing scheme interferes with the other.

The interaction concept presented was implemented by exploiting/incorporating three components:

- spatial (different initial starting points)
- temporal (the longer the button is pressed the higher the frame-rate)
- differential (the further away from the starting point the higher/lower the frame-rate)

Very fine positioning tasks can be done as well as skimming through a long video on high speed without modal changes.

3.6 Play / Pause / Go-to

Usually a video player consists of a play button and a pause or stop button. Most file based video players also have a slider representation of the video to jump quickly to any position in the video.

ProPane as well has a play/pause button. Tipping on pane C (video pane) starts the playback of the video in real-time. Tipping pane C again pauses the video. *ProPane* also has a go-to function. Dragging the thumb on the slider (pane D) lets the video jump to the corresponding position in the clip.

3.7 Implementation

The implementation was carried out on a HTC Sensation smartphone equipped with the Android 2.3.3 operation system. This smartphone has a 4,3" (10,9 cm) screen with a resolution of 960x540 pixels an ARM derived Qualcomm MSM8260 Snapdragon dual-core 1.2 GHz Scorpion processor and 768 MB of internal RAM.

Android was chosen due to its open architecture and the availability of such devices in the lab. This specific device was chosen because it was the most powerful one.

The main concerns considering a fluid interaction experience were immediate visual feedback from the video and a smooth playback. It has to be noted the playback includes slow motion, fast forward and frame-by-frame browsing. To give the users the desired fluid interaction experience, preparation had to be done in advance. The most important one will be discussed now. All video streams used in this study had a frame-rate of 30 frames per seconds. To give the user a smooth playback the application must be capable to refresh the picture shown at the screen at least 30 times a second (refresh rate). Normal playback with 30fps is no problem since the Android platform offers low level functions to playback a video stream at real-time. However, there are no ready made, responsive functions provided for slow motion and fast forward. Even worse the Android platform does not provide direct access to arbitrary frames in the video stream. This has to do with the complex internal structure of modern video codecs [6,9]. But such a function is vital for a quick response. Another problem was that changing the speed of the video (say from slow motion with 15fps to real-time with 30fps) could not be achieved without short pauses.

To overcome these limitations, every video clip was decomposed to its single frames and stored as an image sequence on the smartphone in advance. For a video clip with the length of 2 minutes 3600 single frames were stored. This made it possible to access every single frame of the video just by loading the corresponding image from the file system.

User interaction must have a short response time and at best no delay at all. To achieve this the a) application were separated in multiple threads with different levels of priority, b) the images were prepared to minimize processing power on the smartphone and c) Java-objects were heavily re-used to avoid unreferenced objects and therefore unexpected garbage collection.

All mentioned precaution were taken into consideration during the implementation of *ProPane* to achieve the desired fluid interaction. The implementation was evaluated in a user study described in the next chapter.

4. USER STUDY AND EVALUATION

The user study included 18 participants (14 male, 4 female) from the ages of 16 to 47. Nine are professional video editors or have a formal training in video editing. All of them possess and use a mobile device on a regular basis. Nielsen states that for a qualitative usability study 5 participants are enough [23]. Since we wanted to compare professional video editors with non-professional we needed at least 5 persons of each group.

The main questions to be answered were: Is *ProPane* suitable for browsing videos on a mobile device fast and frame-accurate? Is the interface feasible for both amateurs and professionals videographer? Do participants like it and what would they like to see changed? Would participants add *ProPane* a their favorite browser?

4.1 Evaluation Setup

All evaluations were made using the HTC Sensation smartphone on which *ProPane* was implemented. To force the participants to use the novel parts of the interface instead of already known features, the play/pause button (pane C) and the slider (pane D) were disabled.

In order to evaluate the presented concepts the users were given different exercises close to real life tasks. After that an open discussion took place about the interface. A questionnaire was prepared and every participant was asked all questions during that discussion [15]. If the questions of the questionnaire were not answered during the open discussion they were asked explicitly afterwards. The interview was noted on paper as keywords. Interesting, unusual und unexpected quotes were noted as a whole sentence.

To make the attendees comfortable with the standard browsing interface they were asked at the beginning to choose their favorite video among four different topics (skate, mountain bike, football, playing kittens). The idea was to give the participants an emotional tie to the video and to motivate them to play around with the standard browsing scheme. Every video was about two minutes in length. After a few minutes they were interviewed and they were introduced to the advanced and progressive browsing scheme. They were asked to use all features. For example the users were asked to find their favorite frame or their favorite sequence in the video.

As a last exercise the users were asked to find a specific frame within the video (e.g: “find the first frame where the second biker can be seen” or “find the first frame where the red cat tries to jump on the tree”).

4.2 Questionnaire

Gathering quantitative feedback, the participants were asked numerous questions which the users could rate on a 3-point Likert Scale. Every question could be answered with yes/I would like to (equals 2 on the chart), maybe/I do not care (equals 1 on the chart) and no/I would not like to (equals 0 on the chart). The resulting chart for amateurs is plotted on the left side, the resulting chart for professionals is plotted on the right side.

Question 1: Would you like to add the features of the tested interface to your favorite video player?

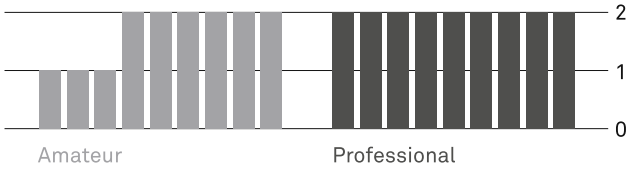


Figure 9: Question 1

Question 2: Do you like the fast forward functionality?

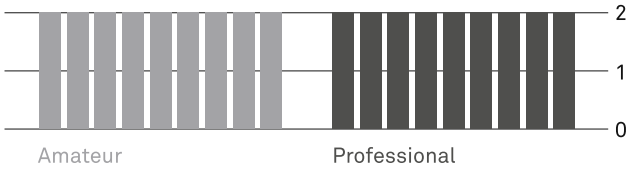


Figure 10: Question 2

Question 3: Do you like the slow motion functionality?

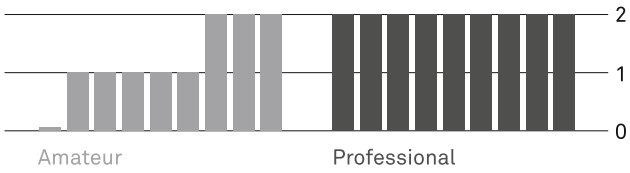


Figure 11: Question 3

Question 4: Do you like the frame-by-frame functionality?

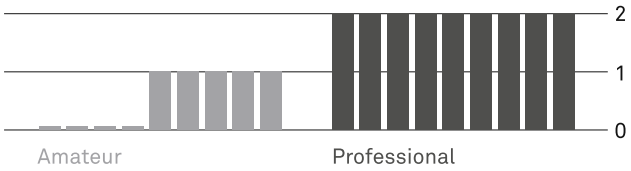


Figure 12: Question 4

Question 5: Do you like the advanced browsing scheme?

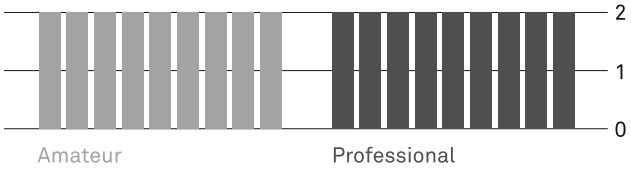


Figure 13: Question 5

Question 6: Do you like the progressive browsing scheme?

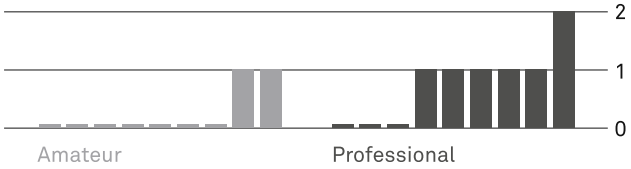


Figure 14: Question 6

Question 7: The professionals were asked if they would like to use this browsing interface in a video-editing suite. All participants strongly agreed.

4.3 User Comments

A discussion was held with every participant about video browsing in general and the tested interface in particular. The statements below are exemplary:

Example 1: *"Some feedback would be nice to see how fast I am at the moment."*

This was stated by one amateur participant directly. Other participants stated such a functionality after the (open) discussion led the topic in such a direction.

Example 2: *"Given such an interface I would start video editing on the mobile."*

This was stated by a few professionals. Amateurs were not strongly interested in video editing on the mobile phones.

Example 3: *"I like the tight handling. I can use it during a bus ride as well."*

This was stated by the same amount of professionals as amateurs.

Example 4: *"Well, that is easy."*

This or something similar was stated by all participants during the first phase of testing the interface.

Example 5: *"Why should I use this? This twists my brain."*

This or something similar was stated by most participants commenting on the progressive browsing scheme.

Example 6: *"I would like to see some sort of a slider."*

This was stated by a majority of the participants.

Example 7: *"Is there a play button? This play button should be in the middle of the screen."*

This statement were made by all participants in one way or another.

The slider and the play button were left out on purpose. So these statements are obvious. However, the comments show how familiar such navigation items already are.

4.4 Results

The outcome of the questionnaire showed that participants liked *ProPane* and would like to use such an interface instead of their actual video player or as an extension to their favorite video player (Q1 and Q7). Most of the participants found the basic browsing mechanism (play, fast forward, fast rewind) useful (Q2).

Advanced features like frame-by-frame browsing were more attractive for professionals; amateurs tend to ignore such functionality (Q4). Other advanced features, such as slow motion, were adopted by amateurs as well as professionals (Q3). Despite the varying use of such features both amateurs and professionals liked the advanced browsing scheme (Q5).

Both amateurs and professionals agreed on the usefulness of advanced browsing, though they both questioned the practicality of progressive browsing (Q6).

Summing up the user statements, amateurs and professionals agreed on the basic ideas introduced with *ProPane*. Given the research questions we can state that the presented interface is capable of browsing videos on a mobile device in a fast and frame-accurate way. The interface is feasible and useful for both amateurs and professional videographers.

This interaction schema seems to be very powerful in terms of fast and accurate browsing. Both browsing areas can be controlled just with the left and right thumb. This allows an ergonomic use and a firm grip of the device similar to mobile game devices or game controllers which was mentioned by the participants.

The varying browsing schemes (basic, advanced, progressive) are independent of each other and do not influence one another. So users can stay with their favorite browsing scheme and leave out the ones they do not like.

A big surprise was to see that users obviously have very diverse mental models of video browsing. One participant wanted to browse forward with the left pane A and browse backward with the right pane B.

Despite the suggestions and remarks all participants liked the smooth interaction design exploiting spatial, temporal and differential parameters.

5. CONCLUSION AND FUTURE WORK

ProPane, a novel interface for fast and precise video browsing for mobile devices with small screens was presented and evaluated. The similarities, differences and novelty compared to existing interfaces were discussed. The presented interface makes it easy to browse at different granularity/speed levels in order to navigate to arbitrary positions in a video clip. Due to the intuitive nature of the interface and the results of the evaluation, we can argue that it is suitable for professional video editors and casual video consumers.

The positive feedback of the user study showed that not only casual users prefer the interface and would add it to their favorite mobile video player, but even professional video editors were enthusiastic about the precision and smoothness. They said with such an interface they would like to cut videos on their mobile phones even if they do not do so now. Professional video editors were excited about the navigation and mentioned a lot of

advanced features useful for video editing. One mentioned feature even useful for amateurs would be adding audio playback.

All gesture are solely done with the thumbs which was highly approved by the participants. The device could be held steadily during browsing tasks, which was positively pointed out during the evaluation.

Despite the positive feedback a few questions arose. It seems that every user has his or her own mental model about navigation in a video. However all suggestions do not question the basic idea of the interface and could easily be implemented. Further implementations should provide a possibility for the mentioned adaptations.

6. ACKNOWLEDGMENTS

We thank all the volunteers who wrote and provided helpful comments on previous versions of this document.

7. REFERENCES

[1] Apple Inc. 2012. *iOS Human Interface Guidelines*. https://developer.apple.com/library/ios/#DOCUMENTATION/UserExperience/Conceptual/MobileHIG/Introduction/Introduction.html

[2] Bodker, K., Kensing, F. and Simonsen, J. 2009. *Participatory IT Design: Designing for Business and Workplace Realities*. The MIT Press, USA

[3] Church, K.,Smyth, B. and Keane, M.T. 2006. Evaluating interfaces for intelligent mobile search. In *Proceedings of the 2006 international cross-disciplinary workshop on Web accessibility* (W4A '06). ACM, New York, NY, USA, 69-78. DOI=http://doi.acm.org/10.1145/1133219.1133232

[4] Cisco Inc. 2012. *Entering the Zettabyte Era*. (August 2012). http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/VNI_Hyperconnectivity_WP.html

[5] M. Davis. 2003. Editing out video editing. In *IEEE Multimedia Magazine, spec. ed. Computational Media Aesthetics* (IEEE Computer Society, vol.10, iss.2, April-June 2003). 54–64

[6] Gao, B., Jansen, J., Cesar, P. and Bulterman, D.C.A. 2011. Accurate and low-delay seeking within and across mash-ups of highly-compressed videos. In *Proceedings of the 21st International Workshop on Network and operating systems support for digital audio and video* (Vancouver, Canada, June 01 – 03, 2011). NOSSDAV '11. ACM, New York, NY, 105-110. DOI=http://doi.acm.org/10.1145/1989240.1989266

[7] Hinckley, K., Cutrell, E., Bathiche, S. and Muss, T. 2002. Quantitative analysis of scrolling techniques. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Minneapolis, USA, April 20-25, 2002). CHI'02. ACM, New York, NY, USA, 65-72. DOI=http://doi.acm.org/10.1145/503376.503389

[8] Holtzblatt, K. and Burns, J. 2004. *Rapid Contextual Design: A How-to Guide to Key Techniques for User-Centered Design (Interactive Technologies)*. Morgan Kaufmann, USA

[9] Hourunranta, A., Islam, A. and Chebil, F. 2006. In *Proceedings of the International Conference on Multimedia and Expo* (Toronto, Canada, July 09-12, 2006). 1305-1308. DOI=http://dx.doi.org/10.1109/ICME.2006.262778

[10] Houten, Y.v., Oltmans, E. and van Setten, M. 2001. *Video Browsing and Summarization*. Enschede, Telematica Insitute (August 2012). Available at https://doc.novay.nl/dsweb/Get/Document-19342

[11] Huber, J., Steimle, J. and Mühlhäuser, M. 2010. Toward more efficient user interfaces for mobile video browsing: an in-depth exploration of the design space. In *Proceedings of the international conference on Multimedia* (Firenze, Italy, October 25–29, 2010). MM'10. ACM, New York, NY, USA, 341-350. DOI=http://doi.acm.org/10.1145/1873951.1873999

[12] Hürst, W., Götz, G. and Welte, W. 2007. Interactive video browsing on mobile devices. In *Proceedings of the 15th international conference on Multimedia* (Augsburg, Germany, September 23–28, 2007). MM'07. ACM, New York, NY, USA, 247-256. DOI=http://doi.acm.org/10.1145/1291233.1291284

[13] Hürst, W. and Meier, K. 2008. Interfaces for timeline-based mobile video browsing. In *Proceedings of the 16th ACM international conference on Multimedia* (Vancouver, Canada, October 26-31, 2008). MM'08. ACM, New York, NY, USA, 469-478. DOI=http://doi.acm.org/10.1145/1459359.1459422

[14] Hürst, W., Meier, K. and Götz, G. 2008. Timeline-based video browsing on handheld devices. In *Proceedings of the 16th ACM International Conference on Multimedia* (Vancouver, Canada, October 26–31, 2008). MM'08. ACM, New York, NY, USA, 993-994. DOI=http://doi.acm.org/10.1145/1459359.1459545

[15] International Organization for Standardization ISO/IEC. 1998. *Ergonomic requirements for office work with visual display terminals (VDT)s—Part 11 Guidance on usability*. ISO/IEC 9241-11:1998

[16] Jokela, H., Karukka, H. and Mäkelä; K. 2007. Empirical observations on video editing in the mobile context. In *Proceedings of the 4th international conference on mobile technology, applications, and systems and the 1st international symposium on Computer human interaction in mobile technology* (Singapore, September 10-12, 2007). MC'07. ACM, New York, NY, USA, 482-489. DOI=http://doi.acm.org/10.1145/1378063.1378140

[17] Jokela, H., Karukka, H. and Mäkelä; K. 2007. Mobile video editor: design and evaluation. In *Proceedings of the 12th international conference on Human-computer interaction: interaction platforms and techniques* (HCI'07), Julie A. Jacko (Ed.). Springer-Verlag, Berlin, Heidelberg, 344-353

[18] Keith, J. 2007. *Video Demystified: A Handbook for the Digital Engineer, Chapter 8, Butterworth Heinemann*. 5th ed., Elsevier Inc,.USA

[19] Laurier, E., Strebel, I. and Brown, B. 2008. Video Analysis: Lessons from Professional Video Editing Practice . In *Forum Qualitative Sozialforschung / Forum: Qualitative Social Research*, 9(3), Art. 37, http://nbn-resolving.de/urn:nbn:de:0114-fqs0803378

[20] Lee, E. 2007. Towards a quantitative analysis of audio scrolling interfaces. In *CHI '07 Extended Abstracts on Human Factors in Computing Systems* (San Jose, USA, April 28 – May 03, 2007). CHI 2007. ACM, New York, NY, USA, 2213-2218. DOI=http://doi.acm.org/10.1145/1240866.1240982

[21] Lee, H. 2001. *User Interface Design for Keyframe-Based Content Browsing of Digital Video*. Dublin City University,

School of Computer Applications. Dissertation. http://doras.dcu.ie/2086/

[22] Moscovich, T. and Hughes, J.F. 2004. Navigating documents with the virtual scroll ring. In *Proceedings of the 17th annual ACM symposium on User interface software and technology* (Santa Fe, USA, October 24–27, 2004). UIST '04. ACM, New York, NY, USA, 57-60. DOI= http://doi.acm.org/10.1145/1029632.1029642

[23] Nielsen, J. 2012. *How Many Test Users in a Usability Study?* http://www.useit.com/alertbox/number-of-test-users.html

[24] Open Handset Alliance. 2012. *User Interface Guidelines.* http://developer.android.com/guide/practices/ui_guidelines/index.html

[25] Richter, H, Brotherton, J., Abowd, G.D. and Truong, K. A. 1999. *Multi-Scale Timeline Slider for Stream Visualization and Control*, Technical Report GIT-GVU-99-30, GVU Center, Georgia Institute of Technology

[26] Schoeffmann, K., Taschwer, M. and Boeszoermenyi, L. 2010. *The video explorer: a tool for navigation and searching within a single video based on fast content analysis. In Proceedings of the first annual ACM SIGMM conference on Multimedia systems* (Phoenix, USA, February 22–23, 2010). MMSys'10. ACM, New York, NY, USA, 247-258. DOI=http://doi.acm.org/10.1145/1730836.1730867

[27] Terrenghi, L., Fritsche, T. and Butz, A. 2008. Designing Enviroments for Collaborative Video Editing. *International Conference on Intelligent Environments* (Seattle, USA, July 21 - 22, 2008). IET'08. DOI=http://dx.doi.org/10.1049/cp:20081137

[28] TNS Emnid. 2012. *Bewegtbild wird inzwischen von jedem Fünften mobil genutzt.* (Köln / Bielefeld, Germany, August 1, 2012). http://www.tns-emnid.com/presse/presseinformation.asp?prID=863

[29] Väätäjä, H. 2010. User experience evaluation criteria for mobile news making technology: findings from a case study. In *Proceedings of the 22nd Conference of the Computer-Human Interaction Special Interest Group of Australia on Computer-Human Interaction* (Brisbane, Australia, November 22-26, 2010). OZCHI 2010. ACM, New York, NY, USA, 152-159. DOI=http://doi.acm.org/10.1145/1952222.1952252

[30] Väätäjä, H. and Männistö, A.A. 2010. Bottlenecks, usability issues and development needs in creating and delivering news videos with smart phones. In *Proceedings of the 3rd workshop on Mobile video delivery* (Firenze, Italy, October 25, 2010). MoViD'10. ACM, New York, NY, USA, 45-50. DOI=http://doi.acm.org/10.1145/1878022.1878034

[31] Viaud, ML., Buisson, O., Saulnier, A. and Guenais, C. 2010. Video exploration: from multimedia content analysis to interactive visualization. In *Proceedings of the international conference on Multimedia* (Firenze, Italy, October 25–29, 2010). MM'10. ACM, New York, NY, USA, 1311-1314. DOI=http://doi.acm.org/10.1145/1873951.1874209

[32] Vihavainen, S., Mate, S., Seppälä L., Cricri, F. and Curcio, I. 2011. We want more: human-computer collaboration in mobile social video remixing of music concerts. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Vancouver, Canada, May 07–12, 2011). CHI 2011. ACM, New York, NY, USA, 287-296. DOI=http://doi.acm.org/10.1145/1978942.1978983

[33] Zsombori, V., Frantzis, M., Guimaraes, R. L., Ursu, M.F., Cesar, C., Kegel, I., Craigie R. and Bulterman, D.C.A. 2011. Automatic generation of video narratives from shared UGC. In *Proceedings of the 22nd ACM conference on Hypertext and hypermedia* (Eindhoven, The Netherlands, June 06–09, 2011). HT'11. ACM, New York, NY, USA, 325-334. DOI=http://doi.acm.org/10.1145/1995966.1996009

Athmos: Focus+Context for Browsing in Mobile Thumbnail Collections

Roman Ganhör
Vienna University of Technology
Favoritenstrasse 9-11/187
1040 Vienna, Austria
roman.ganhoer@tuwien.ac.at

ABSTRACT
Smartphones are already common tools for presenting photos to family, friends and colleagues. However, browsing and searching through collections of photos can be a tedious repetitive task. In order to facilitate fast and convenient browsing we propose a novel interface that is based on the focus+context approach and tries to eliminate the need for scrolling. Furthermore the possibilities of touch-based devices are taken into account to provide a simple and productive interaction design. In general, the interface minimizes the need for zooming into photos to reveal details and offers a contextual overview at any given time. A first small evaluation proves the suitability of the presented design and brought up interesting suggestions for future work.

Categories and Subject Descriptors
H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

General Terms
Video Browsing, Image Browsing, Mobile devices, Touchscreens

Keywords
focus, context, handheld devices, mobile, thumbnail, search, browsing, multimedia, navigation, precise, touch

1. INTRODUCTION
Smartphones are ubiquitous devices in our daily life. Taking photos with such devices is a common task and, consequently, browsing and organizing collections of photos has become commonplace [10].

The default applications for browsing image galleries on mobile devices typically provide a list or grid of thumbnails (Figure 1). For example, the pre-installed photo browser on Apple iPhone 4 displays 20 photos at once, all in the same size. Thus, searching for a specific image in a longer list can lead to extensive scrolling back and forth [1]. Even more cumbersome is the small size of the thumbnails which requires zooming into the thumbnail and out again.

With these limitations in mind, researchers proposed alternatives or extensions to the simple scrollable thumbnail list. Sorting thumbnails according to embedded metadata like date or location is a common method already implemented in several picture browsers [2]. While this approach helps organizing the picture collection, the tedious tasks of scrolling and zooming remain. As a result it is easy to get lost in bigger collections of pictures since the interfaces provide little to no information about the global structures of the collection itself. This can cause a cognitive load for users who must mentally assimilate the overall structure of the information space and their location within it [6].

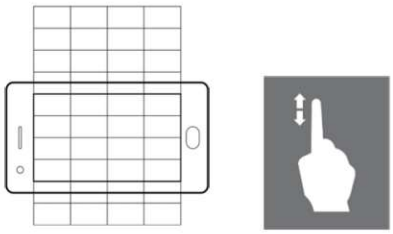


Figure 1: Scrollable grid layout for browsing larger quantities of thumbnails [12]

One obvious way to minimize scrolling is to reduce the size of the thumbnails and therefore the length of the thumbnail list. This has two downsides. First, it is harder to identify the content of an individual thumbnail. Second, a touch gesture can be assigned more easily to the wrong thumbnail due to the size of a finger that can span over more than one thumbnail.

To overcome these limitations of existing interfaces for picture browsing new approaches must be found to minimize or even eliminate scrolling and zooming. The aim would be an interface that allows examining a picture in detail while all remaining pictures serve as contextual information. This approach has already been discussed in several papers and is known as focus+context [6] or Fisheye View [7]. However, little work has been done so far scrutinizing focus+context on picture viewing in the mobile domain. While the limited screen size challenges the design objectives, gesture interaction offers a wide variety of possibilities.

In this paper, we present *Athmos*, a new way of browsing thumbnails by combining existing approaches and adapting them for the mobile domain. The main motivation for our approach is to combine both, detailed information about the thumbnail itself and contextual information such as the position of the thumbnail in the collection or the size of the entire thumbnail collection. Additionally, we want to reduce the demand for scrolling and magnifying thumbnails.

ICMR '14 April 01 - 04 2014, Glasgow, United Kingdom

2. BACKGROUND & RELATED WORK

Pictures and videos are generally represented as static thumbnails, miniature images created from a bigger picture or from a representative still frame of a video. Laying out all thumbnails in a list or grid defines the standard interface for browsing through collections of such thumbnails. Tapping on an image usually enlarges it to the size of the screen. A tap on the enlarged image reduces it back down to the thumbnail size. This is a typical exponent of a zooming interface.

Zooming interfaces imply a separation of focus (zoomed in) and context (zoomed out) both spatial and temporal. The approach to eliminate the spatial and temporal separation by displaying the focus (zoomed in) within the context (zoomed out) in a single view is called focus+context. Focus+context can help to reduce the mental load for the user [16] and lead to more efficient and enjoyable interfaces even on devices with small screens [9]. Multifaceted work has been done in the area of focus+context at small screen [2][17][22] though little work has been done in the area of displaying photos and pictures [13][15]. Patel et.al [15] conducted a thorough investigation of how people search their photo collections on small screens in general, whereas Khella and Bederson [13] describe an image browser for the pocket pc that employs quantum strip Treemaps. One rare example for focus+context photo search on mobile devices is presented at the end of this section (*3D Sphere*).

Fisheye View and DOI

Fisheye view is an approach based on the idea of representing content in various sizes. The items of interest are displayed in their original sizes or at least large enough to make their content easily identifiable. The remaining content is displayed smaller and provides contextual information. Some earlier thoughts on this idea were presented by Spence and Apperley in 1982 [21] and Furnas in 1986 [7]. Both articles try to achieve a balance of local detail with global context on one screen [3]. In the formalization of the Fisheye View Furnas introduces an index called "Degree of Interest" (DOI). The DOI assigns a number to each node (item) in a structure estimating the users interest in seeing that node, given the current task [7]. Generally speaking, the further away from the focus the lower the DOI. If the DOI falls under a specific value the item is not presented to the user. However, the DOI can be combined with geometric distortion such as downscaling a thumbnail, i.e. the lower the DOI the higher the distortion of an item [5].

Practical Approaches

Holmquist describes a practical focus+context approach to visualize large data sets known as *Flip Zooming* [8]. As an example the authors take 31 pictures of American street signs. The picture of interest is in focus of the user and, thus, is presented at its normal size. All remaining pictures are visible at the same time as a thumbnail sketch to serve as context. In Figure 2, the street sign in focus is surrounded by several other street signs in thumbnail size. Given the western style of reading (top left to bottom right) there are 13 street signs before and 17 street signs after the focused page.

Figure 2: Flip Zooming

Pointing and clicking on a thumbnail changes the focus to that thumbnail. However, that particular approach was not intended for small screen sizes and it is not described how to deal with larger collections of pictures.

Another practical approach and maybe the first large-scale deployment of fisheye style effects is the Dock icon-panel of the Mac OS X (Figure 3).

Figure 3: Mac OS X Dock

Similar to the *Perspective Wall* [14] proposed in 1991, the dock provides a smooth transition between detail and context. Figure 4 illustrates how the size of an item and its DOI are connected. Given that the grey square is in the focus of the user it is therefore shown as the biggest square. The surrounding squares (context) become smaller according to their distance from the item in focus. The further away from the focus the less interesting the item is for the user. Speaking in terms of DOI: the size of an item represents its DOI.

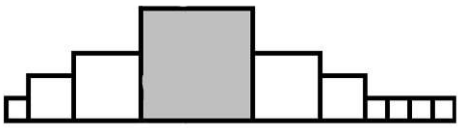


Figure 4: Size of Item and DOI

In order to transfer the aforementioned approaches to gesture based mobile devices, some considerations must be taken in advance. For instance, due to the bounded space of mobile devices the arrangement of items should be more space effective in comparison to *Flip Zooming*. Moreover, a combination of Mac OS X Dock and finger gestures would lead to hidden items of interest (finger on focused item) and fully visible surrounding items (context).

Cover Flow

Cover Flow is well known as part of iTunes and various other Apple software products. It uses the metaphor of flipping through paper cards within a bar jukebox. *Cover Flow's* approach does not reduce the need for scrolling. However, it almost supersedes the need to zoom into a picture. Furthermore, it provides some context such as the predecessors and successors (Figure 5).

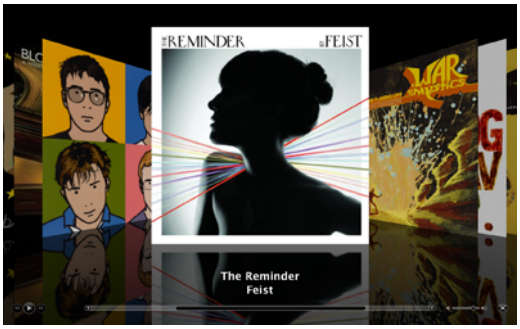


Figure 5: Cover Flow

Cover Flow displays three thumbnails prominently but conceals most parts of all other thumbnails, even on large screens. Thus the number of thumbnails a user can directly select is limited especially on smaller screens. Furthermore, an extra slider is needed for orientation in the collection. Despite the shortcomings it is a viable example of a focus+context approach on mobile phones.

3D Sphere

Another implementation of the focus+context approach consists of the use of 3D graphics to present thumbnail collections [18][19]. Projected on cylindrical or spherical arrangements peripheral thumbnails are displayed smaller in size than thumbnails in the center of the screen. The interface shown in Figure 6 is implemented on an iPad. Due to the spherical projection thumbnails on the outer bounds are not only distorted by size but also by aspect ratio.

Figure 6: Spherical Projection

However, the number of pictures that can be displayed simultaneously is a limitation for such projections. Moreover, the globe does not make very good use of the screen, leaving a lot of unused space.

The Dominant Color

All the aforementioned examples deal with a low number of pictures. Facing the problem of presenting thousands of pictures on one screen, Schoeffmann et.al. [20] have elaborated the *d-Dominant Color* method. In this method the dominant color in every picture is calculated and represented as a colored vertical line of fixed height. When horizontally visualizing the group of vertical lines a colored diagram of a fixed height is developed (Figure 7). Such diagrams obviously do not provide detailed information but can be elaborated to convey some meaning. For instance, assuming that pictures made at night are rather blue and pictures made during daylight are rather brown it is not hard to pin point the position in the diagram where the sun began to rise.

Figure 7: Dominant Color Diagram

Providing that such an approach results in a rectangular outline it is relatively easy to fill up a rectangular screen without leaving too much unused space. Even though the design was intended for use in a different professional field and was not intended to run on mobile devices [20], the basic idea seems interesting and valuable.

Pros and Cons

As previously mentioned, screen utilization is an important topic in mobile interfaces. Any visualization should try to use as much space as possible to maximize the amount of information exhibited on the small screen. At the same time visualizations should be informative, pleasing and not cluttered.

All the approaches presented so far have their strengths and weaknesses. *Flip Zooming* provides a good overview about items located before and after the item in focus. At the same time the available space is not used up completely. In addition to that, it only provides two different sizes for items: a large size for the focal and a small size for all contextual items. The Mac OS X Dock in contrast provides more item sizes and the context can therefore transport more information: the bigger the item, the closer it is located to the focus. However, the smaller the items, the more space is wasted. This is especially exasperating on small screens. *Cover Flow* is intuitive but hides most of the context whereas 3D Sphere is not efficient in screen usage. Finally, the dominant color has a good chance for screen efficiency due to its basic rectangular layout. However, it is not suitable for detailed navigation and must be combined with other approaches in order to work satisfactorily.

In the next sections we introduce *Athmos (Advanced THumbnail browsing on MObile Screens)*, our approach for focus+context thumbnail browsing on small screens. We attempt to overcome the limitations of the mobile context that we identified in past approaches by combining and adapting the presented ideas in a novel and feasible interface.

3. IMPLEMENTATION OF ATHMOS

The proposed interface, *Athmos*, meets three requirements we find beneficial for browsing thumbnail collections on mobile devices. First, at any given time all thumbnails of a collection shall be visible on the screen to serve as the context. Second, the user is aware of the actual position within the collection. Third, at least a small set of thumbnails should be big enough for identifying their content without the need for magnification. In order to meet all three requirements at the same time we worked on finding a balance between the opposed goals in thumbnail size. Thumbnails within the focus should be as big as possible. This forces them in focus to use a bigger portion of the screen leaving little space for the context. At the same time thumbnails that composes the context also demand for as much as possible space to provide their contextual information. However, a balance must be found to achieve an efficient interface.

We also considered enjoyable user experience and ease of perception as an important issue for our proposed interface. To support perception the interface should incorporate natural

metaphors, e.g. the transition from focus to context should be seamless. As the Degree of Interest (DOI) lowers, the smaller a thumbnail gets. This indicates a smooth transition from the biggest to the smallest thumbnail size. Furthermore, the interaction should also support common gestures for mobile devices such as *swipes* and *flings*. In addition, potentially unusual gestures can be added but should not interfere with the standard gestures. Smartphones provide powerful API's for standard gestures and standard graphical effects. Introducing new gestures and graphical effects can result in erratic gesture recognition and slow graphic rendering. Again, a balance between user experience and smartphones capabilities must be made.

Initial interviews with potential users and interaction designers confirmed these requirements as reasonable. One suggestion interviewees made several times was to evaluate the interface with thumbnails taken from a movie. Due to the temporal nature of a movie a thumbnail's position in the collection indicates its corresponding clip position in the movie. Furthermore, using screenshots from a DVD or similar media guarantees that every thumbnail has the same aspect ratio - 16:9. This is especially helpful when it comes to implementing a first prototype of the interface without caring too much about different aspect ratios (e.g. 4:3, 3:2, 16:9) and image orientations (landscape, portrait). Despite the previous preoccupation, the interface in question is not intended to be limited to video thumbnails.

Layout

The basic idea of *Athmos* consists of a long strip of thumbnails which is slid under a magnifying lens. The thumbnail in the center of the lens gets magnified the most. The further away from the lens the smaller the thumbnails are (Figure 8). While the magnifying lens is in a fixed position the strip can be moved to the right (moving closer to the beginning of the strip) or to the left (moving closer to the end of the strip) with a simple move-gesture.

Figure 8: Conceptual Sketch: Thumbnails Close to the Lens get magnified

Since a single thumbnail queue does not make the best use of the screen, considering the limited size and the form-factor of a smartphone, *Athmos* splits the thumbnails into three ranges (Figure 9); the *current range* can be seen as the focus and populates the center of the screen, the *starting range* is located in the upper area of the screen and the *ending range* is moved to the lower area of the screen. Both, *starting range* and *ending range* serve as context.

All thumbnails in the context range have the same height (y-axis) while their widths (x-axis) vary. The more thumbnails in the context the smaller they get. This unbalanced pinching clearly distorts the image and can lead to patterns similar to Dominant Color Diagram (Figure 7). This drawback of distorted images was taken into account to make good use of the screen.

Figure 9 demonstrates this idea with 90 thumbnails. Whatever the number of thumbnails is, the comparison of the first row and the last row always indicates the position of the focus within the thumbnail collection. Through the size and number of thumbnails in the first and last row we can roughly estimate our position as pretty much at the beginning of the collection. This is indicated by the ratio the thumbnails aspect ratios are squeezed in the *starting range* and in the *ending range*. Taken Figure 9 as a random example: in the *starting range* we are able to estimate or even count the number of thumbnails (nine thumbnails). In the *ending range* we can only refer to them as "a lot". However, in this example the user can easily identify the approximate position in the thumbnail collection as "rather in the beginning".

When comparing the *starting range* and the *ending range* we see that the thumbnails are always displayed over the whole width of the screen, independently of their number. Due to the small amount of thumbnails in the *starting range* at least a few thumbnails (three thumbnails) are virtually not distorted in terms of aspect ratio. On the left side of the *starting range* the thumbnails are already visibly distorted in terms of aspect ratio, indicating their bigger distance to the focus (*current range*). The *ending range* appears crowded and indicates the greater number of thumbnails compared to the *starting range*. As a result, comparing *starting range* and *ending range* provides the contextual information of the position of the focus.

Figure 9: Layout and Naming of the Athmos-Interface

While Figure 9 shows an arbitrary position in the collection, Figures 10 and 11 depict special cases, such as being at the very beginning and at the very end of a thumbnail collection. Whereas these special cases poorly exploit the screen the relatively low number of such cases means it is generally not a problem.

Figure 10: Picture at the beginning of a collection

Figure 11: Picture at the end of a collection

Presenting thumbnail collections in the way described above might have several benefits. In most cases the screen is well utilized and the context improves orientation. It provides a smooth transition of thumbnail sizes whenever possible. Due to the heavy distortion of thumbnails even bigger collections of pictures (thumbnails) can be displayed. On the other hand, special cases like seen in Figure 10 and Figure 11 can end up with unused space. Considering the small number of special cases in comparison to the expected overall benefit we did not address this issue in this paper.

Thumbnail Sizes

According to recent studies the human brain has a remarkable capability to perceive details even on very small thumbnails [11]. Even heavily distorted and small thumbnails can communicate information about their content. At least they can serve as a context for the collection of thumbnails they are located within. Since we decided (for evaluation purposes) to take thumbnails from a movie all thumbnails have an aspect ratio of 16 to 9. While the focus (*current range*) supports two thumbnail sizes the context (beginning range and the *ending range*) can contain thumbnails with seven different sizes.

The focus of the thumbnail collection is always in the center of the screen. Two slightly smaller thumbnails accompany a prominent thumbnail in the center of the screen. In contrast, the number of thumbnails in the context is variable. As the number of thumbnails varies the sizes of the thumbnails also vary to better fit into the *starting range*/*ending range*s. To lower the calculation power seven thumbnail sizes for the context thumbnails were pre-defined. At startup every picture in the collection is resized in all 9 possible sizes (2 sizes for focus, 7 sizes for context). During interaction the application only needs to calculate the position of a thumbnail and load the correctly sized thumbnail and display it. The different sizes in Figure 12 are labeled with letters beginning with A and B for the thumbnails in the focus and C to I for the thumbnails in the context. Thumbnails in the focus, category A and B, always maintain the correct aspect ratio. Thumbnails in the context can either maintain their aspect ratio, category C, or distort the aspect ratio, category D to I. The more thumbnails in the context, the more distorted they get in order to fit into the limited available space.

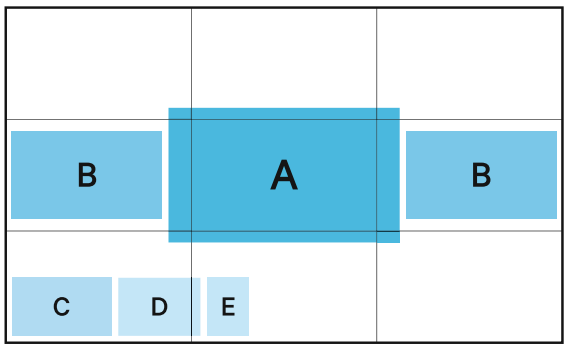


Figure 12: Names and positions of the thumbnails

Athmos is implemented for an Android based smartphone (Samsung Galaxy Nexus) with an effective screen size of 1196x720 pixels. Thumbnails sizes in the *current range* (focus area, size A and B) never change. The sizes for A and B were defined with two prerequisites in mind: first, the bigger thumbnail (A) should be approximately half the width of the screen; second, all thumbnails in the focus must support 16:9 aspect ratio. Therefore, thumbnails of category A have 524x294 pixel and thumbnails of category B have 325x182.

In contrast the thumbnails in the context can vary in their sizes depending on the number of thumbnails in the particular range. However, the algorithm of *Athmos* tries to a) display thumbnails as big as possible and b) assure a constant decline in width. The height of a thumbnail in context is always 131 pixels. Thus, thumbnails in the context (category C to I) do not necessarily fulfill the correct aspect ratio. Thumbnails of category C come close to the aspect ratio of 16:9 and the distortion is almost not visible. Thumbnails of category D are visibly distorted but still disclose a good portion of their content. The more the thumbnails get distorted the more they serve just as contextual information (Table 1).

Category	C	D	E	F	G	H	I
Y	233	115	57	27	13	6	4
X	131	131	131	131	131	131	131
items per row	5	10	20	41	92	170	299
margin x	6	4	2	2	1	1	0

Table 1: Category and size of thumbnails in the starting and ending range (context)

To calculate the maximum number of thumbnails visible simultaneously on the screen we consider only thumbnails of category I in the context (*starting range* and *ending range* with 299 items per row, Table 1). The number of thumbnails in the focus (*current range*) is fixed to three thumbnails. Adding this up we calculate 299 + 3 + 299 = 601. To calculate the maximum number of thumbnails the interface can handle we have to consider special cases as shown in Figure 10 and Figure 11. For these special cases we add 299 (context) and 2 (focus) adding up to 301 thumbnails. The maximum number of non-distorted thumbnails - considering thumbnails of category C as non-distorted - visible simultaneously on the screen is 5 plus 5 (context) plus 3 (focus) equals 13.

Gestures

The basic gestures are built around the three main thumbnails in the center of the screen (focus). Expanding this trisection to the upper and lower area leads to splitting the screen into nine tiles of the same size (Figure 13). Any gesture starts within the boundaries of a tile and ends in the boundaries of another. A gesture is notated as “Start Tile to End Tile”, e.g. “5 to 4”. All implemented gestures are depicted in Figure 14.

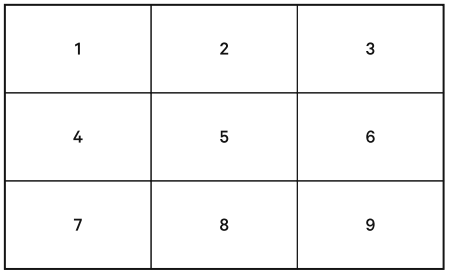


Figure 13: The screen divided into 9 tiles

Moving 1 position (e.g. “5 to 4”) moves the virtual strip under the magnifying glass (tile 5) to the left; hence the collection moves one picture closer to the end. Carrying out this gesture repeatedly leads ultimately to the situation shown in Figure 11. The same effect has the gesture “6 to 5”. In contrast “4 to 5” and “5 to 6” eventually lead to a setting as the one illustrated in Figure 10.

Moving 2 positions (e.g. “6 to 4”) basically has the same effect as two repetitions of the gesture “5 to 4”. “4 to 6” has the same effect as two times the gesture “5 to 6”.

Moving 3 positions (e.g. “5 to 2”) has the same effect as three times the gesture “5 to 4”. “5 to 8” has the same effect as three times the gesture “5 to 6”.

The gestures described so far are used for precise browsing. Every gesture is predefined and, therefore, has a precise amount of movement. In contrast, the following gestures allow fast but not always exact browsing.

When *moving arbitrary positions* (e.g. “9 to 5”) an algorithm calculates which thumbnail is selected dependent on the position of the finger. When moved to tile 5 the selected thumbnail will be dropped there. This algorithm is called whenever gesture “1 to 5”, “2 to 5”, “3 to 5”, “7 to 5”, “8 to 5”, ”9 to 5” is performed.

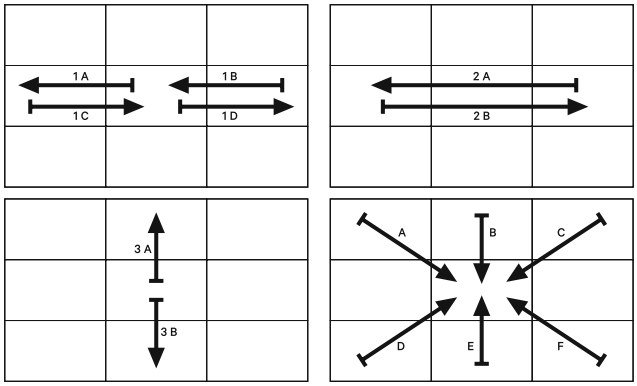


Figure 14: Top left: moving 1 position; top right: moving 2 positions; bottom left: moving 3 positions; bottom right: moving arbitrary positions

4. EVALUATION

To gain some insights in how *Athmos* perfoms in the users' hands, we conducted a qualitative user study. The overall goals were an estimation of user acceptance and gathering user feedback as well as suggestions for improvements. The user study included 11 participants (7 male, 4 female) from 16 to 47 years. Six were professional media artists or had a formal education in media art. All of them possessed and used a mobile device on a regularly basis.

At the beginning, a short oral tutorial was given to each user where they were introduced to the basic ideas of *Athmos* and its operating gestures. Understanding and learning the interaction concept turned out to be intuitive and easy with minimal oral instructions. All participants understood the concept of simultaneously displaying all thumbnails on one screen and how the gestures work.

To estimate the practicability of *Athmos* two categories of user tasks were conducted. A low level mechanical task, such as target acquisition and a high level cognitive task such as the users' ability to search and comprehend the information space [6]. After these tasks were completed a small questionnaire was conducted. Finally, an in-depth interview was made with every participant to gain information about the usability of the interface and to gather ideas for further improvements. The open interview was designed to encourage the participants to criticize the presented interface and to propose new concepts.

The users were given a set of 90 prepared screenshots sampled from a movie and were asked to run two exercises which were similar to tasks normally carried out in such thumbnail galleries. Exercises included (1) finding specific thumbnails and (2) going to a certain position. Both task were also carried out with a conventional thumbnail-based interface.

(1) *Search task*: the users were shown a picture and were then required to find a similar thumbnail. Pictures, and hence thumbnails, that stood out in terms of color were easily found on both interfaces. With *Athmos*, users could find them at first glance several times whereas with the conventional interface further interaction was commonly required (scrolling and zooming). This was observed during the evaluation. Some remarks made by participants were: *"This interface gives me a good idea where different sequences are located even though they are highly compressed.", "So I can browse the gallery and look at pictures at the same time? That is cool."* or *"There is no need for switching back and forth which makes it easier for me"*. Stating from the comments *Athmos* offered a better user experience due to its automatic built-in magnification feature. In contrast, conventional interfaces have to enlarge every thumbnail separately, which can be tedious. However, the comments could also be an expression about the novel interface in general.

(2) *Position task*: the users were asked to browse to defined positions in the thumbnail collection. This included a) the first thumbnail b) the last thumbnail c) the central thumbnail d) ten thumbnails closer to the end from the current position e) the tenth thumbnail from the beginning. Both interfaces did equally well. Some tasks (e.g. task a) and task b)) were equally easy on both interfaces, due to the fact that the tasks did not require contextual information. Other tasks, such as c) were easier to complete with *Athmos* as this interface conveys information about the user's position through the arrangement of the visible thumbnails. Conventional thumbnail interfaces with scrollbars that fade out after a few seconds adds to the users own doubt as to his or her correct position. Several participants repeatedly tapped on the screen just to make the scrollbar visible to check and confirm their own position. Some remarks made by participants on *Athmos* were: *"This makes perfect sense for me.", "With this magnified picture in the center I feel in better control over the whole process."* or *"I would like to try it with my own pictures"*. Stating from the comments and the observation *Athmos* offered a better general view over the current position due to the ability to display all thumbnails at the same time. Again, the comments made could be an expression about the novel interface in general.

After the tasks were carried out a small questionnaire was made to gather a first impression of the user experience. The answers were assigned according to a 4-point Likert Scale. The answer set was no (equals 1 on the Likert Scale), rather no (equals 2), rather yes (equals 3) and yes (equals 4). A neutral answer was left out on purpose to force the participant forming an opinion and making a statement. Figure 15 to Figure 17 show a summary of the distribution of the answers.

Question 1: Considering the tasks you did just before. Do you think the concepts presented with *Athmos* were beneficial?

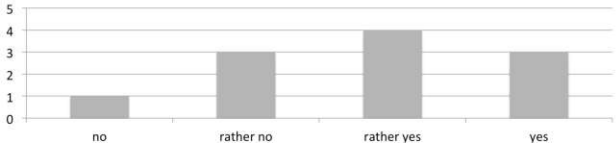


Figure 15: Question 1

Question 2: Do you think the idea of *Athmos* could be useful for you personally?

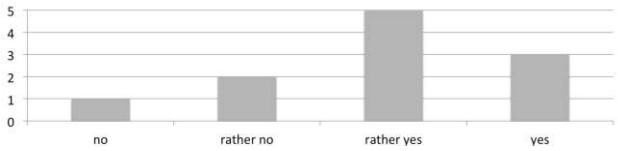


Figure 16: Question 2

Question 3: Given the chance you could add new features to *Athmos*. Do you think such an advanced *Athmos* could be useful for you?

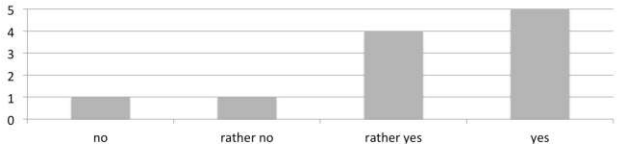


Figure 17: Question 3

Given the widespread adoption of the vertical scrolling metaphor the feedback was surprisingly positive and encouraging. The participants noted the attraction of the novel approach even though they thought it is not always superior to the conventional scrolling list.

Despite the general positive feedback, nearly every participant had suggestions for improvements. Most frequently, additional gestures such as a slow *swipe* from tile 1 to tile 3 were mentioned. This gesture would push the thumbnails right of the finger through the focus. This gesture would be a convenient shortcut for a repeatedly carried out "4 to 5" gesture.

Picking a thumbnail out of a crowded area (e.g. lower right area in Figure 9) tends to be imprecise due to the size of the thumb is bigger than the one distorted thumbnail and the thumbnails do not reveal a lot more then the average color of the represented image. Participants suggested integrating mode change. Whenever the interface senses the beginning of the gesture "move arbitrary positions" the mode of the *current range* would change. This mode change can be made visible through different colors. Instead of displaying the thumbnails in the focus, the *current range* would display the thumbnails under the finger of the participant. Thus, it would be easily possible to identify even heavily distorted thumbnails.

Extending the browsing-only concept with an ability to reorganize thumbnails was suggested several times by media artists. This feature seems especially useful for longer thumbnail list. Dragging a thumbnail over a longer standard list can be tedious and time consuming. This idea was maybe triggered by the thumbnail collection that was used for this evaluation since reorganizing movie clips (thumbnails) is a common task in video editing. However, participants also mentioned that they would reorganize their media items more frequently if it were not for the lack of convenient interfaces. Finally a few participants expressed their wish to use the interface on their own mobile device with their own media items.

5. CONCLUSION AND FUTURE WORK

In this paper, *Athmos*, a novel interface for fast and precise thumbnail browsing for mobile devices with small screens was presented. The interface combines several approaches for browsing large data sets, particularly focus+context. Users can examine a thumbnail in detail (focus) while the rest of the thumbnails provide additional information (context) such as position in the thumbnail collection. Common touch gestures were adopted and novel interaction elements were introduced to provide a satisfactory user experience.

The interface was implemented on a regular smartphone and tested with a small group of users with various professional backgrounds. The users have carried out tasks with *Athmos* and a standard browsing application. Subsequently a questionnaire was given and an in-depth interview was performed. Despite the fact that the interface elements had to be programmed from scratch the users were pleased with the speed and fluidity of animations the interface offered. Interestingly, no user complained about the proposed interface despite its early prototype-status. On the contrary, every user suggested additional gestures and provided ideas to improve the interface. The amount and quality of suggestion given by the participants indicated that the idea of *Athmos* was well understood. Considering the general positive feedback *Athmos* gathered it seems promising to develop it further to a more advanced browsing tool.

Nonetheless, touch-based interfaces are no longer novel and people are used to this kind of interaction. Consequently, even on novel interfaces users expect familiar gesture to work. Extending the interface with the most often proposed features will be the next step in the development of *Athmos*. It will also be important to discuss the interface and interaction mechanism when displaying more than 601 pictures. Moreover, we will strive to explore two fields. First, can an interface such as *Athmos* be an alternative to existing grid based browsing interfaces? To answer this we are planning to carry out a long-term study encouraging users to use *Athmos* in-situ with their own content. Second, we are interested in finding out for which working domains and collection sizes *Athmos* is suitable. We can use the findings of the long-term study and evaluate the interface with users with various professional backgrounds to answer the second question. However, in spite of the question we plan to address, we acknowledge the fact that we should first think of the various sizes and orientation of the pictures we will be working on.

6. ACKNOWLEDGMENTS

We thank all the volunteers who wrote and provided helpful comments on previous versions of this document.

7. REFERENCES

[1] Ahlström, D., Hudelist, M.A., Schoeffmann, K. and Schaefer, G. 2012. A user study on image browsing on touchscreens. In *Proceedings of the 20th ACM international conference on Multimedia* (MM '12). ACM, New York, NY, USA, 925-928. DOI=http://doi.acm.org/10.1145/2393347.2396348

[2] Alensw.com. 2013. QuickPic - Quick Browsing Tons of Pictures. https://play.google.com/store/apps/details?id=com.alensw.PicFolder&hl=en

[3] Bartram, L., Ho, A., Dill, J. and Henigman, F. 1995. The continuous zoom: a constrained fisheye technique for viewing and navigating large information spaces. In *Proceedings of the 8th annual ACM symposium on User interface and software technology* (UIST '95). ACM, New York, NY, USA, 207-215. DOI=http://doi.acm.org/10.1145/215585.215977

[4] Björk, S. and Redström, J. Redefining the Focus and Context of Focus+Context Visualizations. PLAY: Applied research on art and technology, The Interactive Institute, Gothenburg, Sweden. http://www.redstrom.se/johan/papers/redefining.pdf

[5] Card, S.K. and Nation, D. 2002. Degree-of-interest trees: a component of an attention-reactive user interface. In *Proceedings of the Working Conference on Advanced Visual Interfaces* (AVI '02), Maria De Marsico, Stefano Levialdi, and Emanuele Panizzi (Eds.). ACM, New York, NY, USA, 231-245. DOI=http://doi.acm.org/10.1145/1556262.1556300

[6] Cockburn, A., Karlson, A. and Bederson, B.B. 2009. A review of overview+detail, zooming, and focus+context interfaces. *ACM Computing Surveys* 41, 1, Article 2 (January 2009), 31 pages. DOI=http://doi.acm.org/10.1145/1456650.1456652

[7] Furnas, G. W. 1986. Generalized fisheye views. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '86), Marilyn Mantei and Peter Orbeton (Eds.). ACM, New York, NY, USA, 16-23. DOI=http://doi.acm.org/10.1145/22627.22342

[8] Holmquist, L.E. 1997. Focus+context visualization with flip zooming and the zoom browser. In *CHI '97 Extended Abstracts on Human Factors in Computing Systems* (CHI EA '97). ACM, New York, NY, USA, 263-264. DOI=http://doi.acm.org/10.1145/1120212.1120383

[9] Huot, S., Lecolinet, E. 2007. Focus+Context Visualization Techniques for Displaying Large Lists with Multiple Points of Interest on Small Tactile Screens. In *Proceedings of the 11th International Conference on Human-Computer Interaction* (INTERACT 2007). Springer-Verlag, Berlin Heidelberg. 219-233. DOI=10.1007/978-3-540-74800-7_18

[10] Hürst, W., Snoek, C.G.M., Spoel, W-J. and Tomin, M. 2011. Size matters! how thumbnail number, size, and motion influence mobile video retrieval. In *Proceedings of the 17th international conference on Advances in multimedia modeling - Volume Part II* (MMM'11), Kuo-Tien Lee, Jun-Wei Hsieh, Wen-Hsiang Tsai, Hong-Yuan Mark Liao, and Tsuhan Chen (Eds.), Vol. Part II. Springer-Verlag, Berlin, Heidelberg, 230-240.

[11] Hürst, W., Snoek, C.G.M., Spoel, W-J. and Tomin, M. 2010. Keep moving!: revisiting thumbnails for mobile video retrieval. In *Proceedings of the international conference on Multimedia* (MM '10). ACM, New York, NY, USA, 963-966. DOI=http://doi.acm.org/10.1145/1873951.1874124

[12] Hürst, W., Darzentas, D. 2012. Quantity versus quality: the role of layout and interaction complexity in thumbnail-based video retrieval interfaces. In *Proceedings of the 2nd ACM International Conference on Multimedia Retrieval* (ICMR '12). ACM, New York, NY, USA, Article 45 . DOI=http://doi.acm.org/10.1145/2324796.2324849

[13] Khella, A. and Bederson, B.B. 2004. Pocket PhotoMesa: a Zoomable image browser for PDAs. In *Proceedings of the 3rd international conference on Mobile and ubiquitous multimedia* (MUM '04). ACM, New York, NY, USA, 19-24. DOI=http://doi.acm.org/10.1145/1052380.1052384

[14] Mackinlay, J.D., Robertson, G.G. and Card, S.K. 1991. The perspective wall: detail and context smoothly integrated. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '91), Scott P. Robertson, Gary M. Olson, and Judith S. Olson (Eds.). ACM, New York, NY, USA, 173-176. DOI=http://doi.acm.org/10.1145/108844.108870

[15] Patel, D., Marsden, G., Jones, M. and Jones, S. 2006. Improving photo searching interfaces for small-screen mobile computers. In *Proceedings of the 8th conference on Human-computer interaction with mobile devices and services* (MobileHCI '06). ACM, New York, NY, USA, 149-156. DOI=http://doi.acm.org/10.1145/1152215.1152247

[16] Robertson, G.G., Card, S.K. and Mackinlay, J.D. 1993. Information visualization using 3D interactive animation. *Communications of the ACM* 36, 4 (April 1993), 57-71. DOI=http://doi.acm.org/10.1145/255950.153577

[17] Sarkar, M., Snibbe, S.S., Tversky, O.J. and Reiss, S.P. 1993. Stretching the rubber sheet: a metaphor for viewing large layouts on small screens. In *Proceedings of the 6th annual ACM symposium on User interface software and technology* (UIST '93). ACM, New York, NY, USA, 81-91. DOI=10.1145/168642.168650 http://doi.acm.org/10.1145/168642.168650

[18] Schaefer, G. 2010. A next generation browsing environment for large image repositories. In *Multimedia Tools and Applications, Volume 47, Issue 1* (March 2010), 105-120 DOI=http://dx.doi.org/10.1007/s11042-009-0409-2

[19] Schoeffmann, K., Ahlstrom, D. and Beecks, B. 2011. 3D Image Browsing on Mobile Devices. In *Proceedings of the 2011 IEEE International Symposium on Multimedia* (ISM '11). IEEE Computer Society, Washington, DC, USA, 335-336. DOI=http://dx.doi.org/10.1109/ISM.2011.60

[20] Schoeffmann, K., Taschwer, M. and Boeszoermenyi, L. 2010. The video explorer: a tool for navigation and searching within a single video based on fast content analysis. In *Proceedings of the first annual ACM SIGMM conference on Multimedia systems* (MMSys '10). ACM, New York, NY, USA, 247-258. DOI=http://doi.acm.org/10.1145/1730836.1730867

[21] Spence, R. and Apperley, M. 1982. Data base navigation: an office enviroment for the professional. In *Behaviour and Information Technology, Volume 1, No.1, 43-54*

[22] Wang Y-S. and Chi, M-T.. 2011. Focus+Context Metro Maps. In *IEEE Transactions on Visual Computer Graphics*. Vol. 17, No. 12, p2528-2535. DOI=http://doi.ieeecomputersociety.org/10.1109/TVCG.2011.205

Muvee: An Alternative Approach to Mobile Video Trimming

Roman Ganhör
Multidisciplinary Design Group
Vienna University of Technology
Vienna, Austria
roman.ganhoer@tuwien.ac.at

***Abstract*— Video content creation on mobile devices has rapidly increased during the last years, whereas the on-site mobile post-production capability has not yet followed this trend. On-the-fly video editing is not a common approach among amateur or professional content producers. This paper presents a proof-of-concept mobile application for video trimming that is practicable and efficient. The implementation is evaluated through a user study involving a task-based exercise, a questionnaire and an open interview. The study indicates that even complex applications can offer positive user experience when the particular design is carefully thought through to overcome the limitations of mobile devices.**

Index Terms: mobile; interface; video; small screen; usability; video production

I. INTRODUCTION

In just a few years smartphones became an integral part of everyday life. Improvements in hardware technology make smartphones a platform for even complex applications. Fully featured desktop applications come into focus for smartphones such as editing photos and audio. Editing video, however, is still a research area where little work has been done so far [1].

The demand for effective and efficient interfaces for mobile video editing rises with the opportunity to record videos on mobile devices [2]. News agencies like CNN and BBC already turn their audience into video based news-generators [3][4]. Newspapers enhance their articles with video content that was recorded and edited by online multimedia journalists transmitting videos directly from the scene [3][5]. Younger users, in contrast, take a more spontaneous approach to mobile video editing as they record video in-situ. Videos are then shared via Bluetooth or websites like YouTube [6][7].

With the introduction of dedicated cameras that are equipped with the Android operating system (Figure 1) the potential users for mobile video editing already stretches from casual users to semi-professionals and professionals.

Despite the wish of various user groups to edit recorded clips in-situ, this is rarely done, mostly because editing implies a different physical and social context [6] or simply because it is too difficult [8][9]. One essential task in video editing is shortening and removing irrelevant material from clips to fulfill filmography-standards or just to confine a clip that is too long [8][10].

Figure 1. Android Photo-/Video-camera with Interchangeable-Lenses

This paper focuses on video trimming as an important part of video editing. To assess the current alternatives for mobile video trimming we evaluated several popular video-editing applications in collaboration with professional video editors in a usability study. The evaluation was set up to find best practices for a joyful and efficient user experience when general video editing tasks are applied. Building upon the findings we implemented a novel interface that, in contrast to most popular mobile video editors, does not mimic desktop-based interfaces. During the design phase we tried to elaborate the advantages of touch-based interfaces and minimize the disadvantage of the limited screen size. To test our design decisions we conducted both a quantitative and a qualitative study.

The contribution of this paper is *Muvee*, an alternative interface approach for mobile video trimming. The interface design is feasible for mobile devices that are based on touch and gesture interaction. Our results show that video trimming on mobile devices can be fluent and practical.

II. BACKGROUND AND RELATED WORK

Searching the literature intensely, we can conclude that little research exists on novel or alternative interfaces for trimming or editing video clips on mobile devices. Most of the research concentrates on the *use* or *context* of existing interfaces [11][12], more efficient algorithms for video applications [13] or software architecture oriented aspects [14]. The few examples of research in this scope primarily focus on video editing on feature phones, e.g. [15].

An essential task in movie making is video trimming. Defining the first frame included in a sequence by marking that frame as the clip's *In point* and defining the last frame

included by marking it as the *Out point* can be referred as trimming [16]. Figure 2 depicts a clip with the length of six frames. The *In point* is set before frame three and the *Out point* is set before frame five. The sequence between *In point* and *Out point* is part of the final movie whereas the rest of the clip is not used. We refer to the sequence finally used as the trimming sequence.

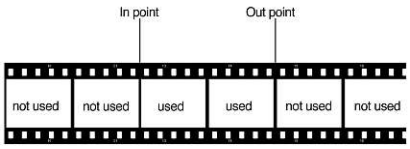


Figure 2. In point and Out point

The task of trimming can be split into two separate subtasks; browsing and marking. Moving with the cursor through the clip and eventually to the *In point* and to the *Out point* is the browsing subtask. Flagging the position as the *In point* respectively as the *Out point* is the marking task. Trimming video clips on desktop computers is supported by specialized software tools like Adobe Premiere or Magix Video. Most of these tools employ a timeline based metaphor for precise video editing. [17].

A. *Mobile Interfaces for Trimming*

Two of the most popular video editors for the now dominating mobile operating systems are Apple *iMovie* and Samsung *Movie Studio*. Apple *iMovie* is a video editor for iOS 4.0 and later. It was in the Apple Appstore Top 50 download charts in September 2013 [18] and is a pre-installed app since iOS 7 [19]. *Movie Studio* by Samsung is a pre-installed video editor for the Samsung Galaxy Nexus S3 smartphone, which was sold over 50 million times by March 2013 [20].

Figure 3. Apple iMovie (left) and Samsung Movie Studio (right)

To the best of the authors' knowledge and according to the download/sales statistics these are the top video editors for their respective platforms. Both apps mimic the time-line based approach (Figure 3) known from desktop interfaces for video editing. In both applications the timeline is placed on the lower part of the screen whereas the preview picture is located in the upper part. To browse and trim a clip, small handles attached to the clip must be targeted and moved to the appropriate frame.

Other mobile video editors break with the familiar timeline metaphor known from desktop applications. *V-Cut Express* does not offer handles to change the clip length directly. Instead the clip is represented as a filmstrip on the lower part of the screen (Figure 4, left). Two scissors indicate the *In point* and the *Out point*. Six buttons on the left allow to position the left scissor to mark the *In point* and six buttons on the right allow to position the Out-point. The step width is stated on the buttons (1 second, 10 seconds or 1 minute). A finer grain can be achieved by means of the play and pause button in the top left corner. *VidTrim - Video Trimmer* makes use of a slider to define the *In point* and *Out point* (Figure 4, right). The left knob on the slider shows the *In point* whereas the right knob shows the *Out point*. Applications like *AndroVid* [21] allow zooming the slider to gain more precise navigation.

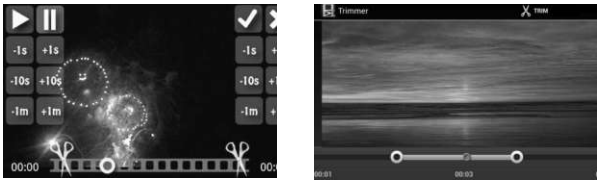


Figure 4. V-Cut Express (left) and VidTrim (right)

B. *Mobile Interfaces for Browsing*

A relatively novel interface for fast and precise browsing on mobile devices is *ProPane* [22]. When taking the ideas of Fitts [23] into consideration, *ProPane* is a suitable example for maximizing the width of the interaction elements and minimizing the distance between these elements. *ProPane* combines frame-by-frame navigation, normal speed browsing and fast browsing in both directions in one, easy to handle interactive interface. The bigger portion of the screen is reserved for the clip whilst the browsing elements are accompanied around the clip. The separation of the navigation elements from the content avoids occluding the clip with the thumb during browsing. However, the main purpose of *ProPane* was browsing videos on mobile devices fast and precise, i.e. trimming is not possible.

C. *Measuring Mobile Usability*

Most operating systems provide their own set of guidelines for designing and implementing user interfaces. These guidelines ensure a consistent user experience. Due to their novelty, new approaches for interface and interaction design cannot always comply with such guidelines. However, one formula that is applicable for almost every screen-based interface is Fitts' law, which states that the time to carry out a task on a pointing device is almost entirely determined by the ratio of target distance and target width. Fitts' law is a robust and quantitative law in human-computer interaction research and design [24][23]. However, the transition of Fitts' Law to the mobile domain is still a topic of interest in the research community [25][26]. These studies indicate that the target width will remain an important factor in the mobile domain. Thus, finding a compromise between maximizing all important interaction

elements and leaving sufficing space for the data will be one of the challenges when implementing complex applications on devices with small screen. Splitting one complex application with a multitude of interaction elements into several smaller sub-applications can turn out favorable when done with adequate care. As long as there is no valid data on balancing between large target width (favors separate screens for different tasks within an application) and low cognitive load for the user (favors one screen for all tasks) the issue will be up to the designers and researchers.

III. IMPLEMENTATION OF MUVEE

The development of *Muvee* was motivated by the need for an interface to trim video clips on mobile devices efficiently and effectively. To gather information about the mobile interfaces presented in Section 2 they were analyzed in collaboration with three professional video editors. The goal was to pinpoint the *pros* and *cons* of the various approaches. Furthermore, the analysis and the discussion was intended to concretize and, if possible, determine the vital functions for a mobile video trimming interface that is both, as precise as a desktop application and feasible to use on a mobile device.

Since all interfaces exhibited in Section 2 allow more interaction than just video trimming, all the tasks that usually occur before and after video trimming were also discussed. Every interface was reviewed in respect to both, the interaction elements needed for video trimming and the limited resources mobile devices offer. In the following we sum up the outcome.

A. Requirements Analysis

The discussants agreed on six distinctive interaction steps for video trimming: (1) c*oarse browsing* to quickly reach any position in the video; (2) *fine browsing* to play forward or backwards at normal speed and slow motion; (3) *marking* the *In point* (4) *marking* the *Out point*; (5) *jumping* to the next clip and (6) *jumping* to the previous clip. These six interactions are repeated in no particular order until all clips are trimmed correctly.

The interfaces of Apple *iMovie* and Samsung *Movie Studio* are easy to understand due to their similarity to existing desktop metaphors such their basic layout and the timeline metaphor. However, the differences between desktop interfaces and mobile interfaces are noticeable and differ not only in detail. So is browsing through a clip on the desktop interface done with the mouse or keyboard shortcuts. In contrast, browsing on the mobile device is done with the tip of the finger on a small screen. Additionally, mobile devices do not offer shortcuts and thus, functions are always accessed through their visible interface element. The more interface elements are visible at any time, the smaller they have to be to fit on the screen. For instance, when changing the length of a clip (*iMovie* and *Movie Studio*) tiny handles must be targeted and moved forward and backward. Despite the tiny handles the finger hides a significant portion of the screen, whereas a mouse cursor on the desktop hides just a small portion of the screen.

V-Cut and *VidTrim* are not complete video editors like the aforementioned and concentrate on video trimming. The interfaces provide fewer options and consequently, their interfaces appear less cluttered. Both interfaces employ a slider at the bottom of the screen representing the length of the clip, whereas the knobs (or scissors) mark the *In point* and *Out point* of the trimming sequence. Since the slider is always visible the user is always aware of the relative position of *In point* and *Out point*. This permanent awareness of clip length and trimming sequence parameter was seen positively by the discussants. A downside of both interfaces was their inability for precise browsing on a frame-by-frame level. The smallest step width of *V-Cut* is one seconds (30 frames) whereas the precision of *VidTrim* depends on the length of clip. Utilizing the *play/pause* buttons for precise browsing was not seen as an adequate workaround.

Summarizing, none of the discussed interfaces allow convenient, fast *and* precise video trimming. Either they concentrate on precision and lack on convenience of interaction (usability) or, they are convenient to use and lack on precision. However, combining the advantages of the different approaches in one interface whilst minimizing or avoiding their downside could eventually lead to a better interface for video trimming. The following list outlines the requirements proposed by the video editors during the discussion:

- large interaction elements as suggested by Fitts' law *(V-Cut)*
- a slider element to jump quickly to any positions within the clip (*V-Cut, VidTrim*)
- constant visible feedback about clip length and the position of *In point* and *Out point* (*V-Cut, VidTrim*)
- two dedicated thumbnails representing the *In point* and *Out point*
- a precise navigation on frame-level (*iMovie*)
- easy access to the next and previous clip (*Movie Studio*)
- unique gestures that cannot be confused with each other and support muscle memory
- minimize clutter on the screen

This set of requirements was the starting point for further research on existing interfaces for video browsing that could serve as a basis for a feasible mobile video trimming interface. The research included scientific papers as well as patents [27][28].

B. Layout

Finally, the basic layout of *Muvee* was derived from *ProPane* [22] due to its capability to browse fast, slow and frame-by-frame with a single set of interaction elements. Thus, two of the six before mentioned interaction steps are fulfilled. Furthermore *ProPane* offers enough free space for additional interface elements that can be utilized for the remaining requirements for video trimming.

Figure 5 depicts the interaction elements of *Muvee* and their usage. Panel D as the main panel shows a preview of the actual frame of the clip. Panel B1 and B2 are used for browsing forward and backward in the video at different

speed levels. These two panel elements are taken over from *ProPane*. *Muvee* introduces the additional panels A, C1 and C2 as well as additional interaction gestures. Panel A is a slider element, C1 is the landing page for setting the *In point* whereas C2 is the landing page for the *Out point*. In order to set a frame as the *In point* a swipe gesture must be made from the clip to the corresponding landing area (D to C1). To set a frame as the *Out point* a swipe gesture must be made from the clip to the corresponding landing area (D to C2). Moving the finger from D to B1 jumps to the next clip. Moving the finger from D to B2 jumps to the previous clip. Moving from D to A or from D to the area between C1 and C2 can be used to switch to other tasks such as importing or organizing clips.

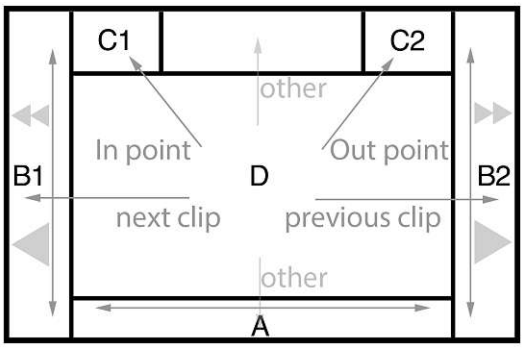


Figure 5. Interface Segmentation (arrows indicate swipes)

Figure 6 shows the implementation of *Muvee* on an Android smartphone. The main panel is reserved for the clip itself (D). Left and right of the clip are the interaction elements for browsing forward and backward (B1 as the red panel, B2 as the green panel). Panel A is segmented into two distinctive areas. In the upper area is a slider where a knob indicates the current position in the clip. Grabbing the knob and moving it to another position on the slider jumps to the corresponding position in the clip. In the lower area a yellow stripe indicates the position of the *In point* and the *Out point* within the clip. The length of a clip is always mapped to the width of panel A. In Figure 6 the marked area in the clip is approximately the middle third of the whole clip. The *In point* and the *Out point* are shown as thumbnails on panel C1 and C2 respectively.

Additionally five time codes are visible: in the lower right corner the total length of the clip; in the lower left corner the current position in the clip; in the upper left corner the time code of the *In point* and in the upper right corner the time code of the Out-point. In the upper center the length of the trimming sequence between *In point* and *Out point* is shown. The time codes were suggested by media artists and can be hidden if wanted.

Figure 6. Muvee Interface Segmentation

IV. EVALUATION

To better understand how *Muvee* performs in the users' hand we conducted a quantitative and a qualitative user study. The overall goal was to understand how well *Muvee* supports video trimming and hence, an important part of the video editing task in general. Furthermore we wanted to collect user feedback and suggestions for further improvements. The user study included 22 participants (17 male, 5 female) from the ages of 15 to 39. All of them possess and use a mobile device on a regular basis. Ten persons are familiar with video editing and edit videos on a regular basis on their desktop computer. Eight participants record videos on their mobile device frequently yet none of the participants edit videos on their mobile device.

A. Setup

During the study the participants evaluate *Muvee* and one other interface that has a conceptually different approach to video trimming. This deliberate distinction exposes the participants to different interaction concepts helping them to better formulate their own thoughts and ideas and thus, gaining valuable qualitative insights. Furthermore, the authors can compare different concepts quantitatively with pre-defined tasks carried out on both interface concepts. The conceptually different alternative to *Muvee* were introduced in Section 2. The authors are aware that some of these alternatives are embedded in bigger applications and offer more functionality than just video trimming (some provide a holistic editing capability). However, for purposes explained above and for discussing the pros and cons of the different *concepts* the authors think that this is a practical and sufficient basis for comparison and reflection.

All alternatives were pre-tested by the authors and ranked by their suitability for the study. *V-Cut* does not allow browsing on a frame-by-frame basis, whereas *Movie Studio* and *VidTrim* were highly unstable during the pre-tests crashing regularly. These made all three alternatives unfeasible candidates for the study. *iMovie*, in contrast, did not crash once during the pre-test, offers a fluent interaction concept on a frame-by-frame basis and has an interaction concept sufficiently different to *Muvee*. This made *iMovie* a suitable candidate for the evaluation task. *iMovie* ran on an fourth generation iPod with a resolution of 960x640 pixel

whereas *Muvee* was implemented on a regular Android smartphone with a screen resolution of 960x540 pixels. Although the hardware were different in their technical specifications (processor, memory, etc.) the devices were sufficiently similar in size and handling.

One aspect of *Muvee* is to gain more insight to the process of mobile video trimming. Therefore eight tasks were prepared that are similar to real world video-trimming tasks. Four clips (clip 1 to 4) have a length of 15 seconds (450 frames) and the four clips (clip 5 to 8) have a length of 1 minute (1800 frames). Every frame of a clip has its frame number visibly written in its center to ease orientation for the participants. Additionally, a trimming list was prepared depicting all clips and their corresponding *In points* and *Out points*. The trimming list denoted *In points* and *Out points* with a bold line and had the exact frame numbers written beside the graphical representation (Figure 7). The trimming sequence of clip number 1, for instance, has its *In point* and Out point at frame number 150 and at frame number 300 respectively. Every trimming sequence has a different length and position within its surrounding clip. The combination of various clip-lengths, trimming sequence lengths and trimming sequence positions should reflect the requirements for real-world video trimming.

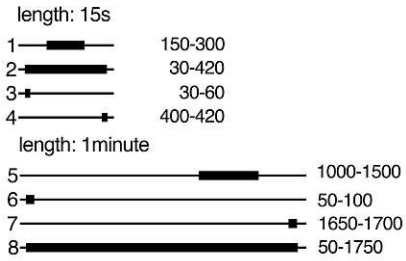


Figure 7. Trimming list depicts areas to be trimmed as bold

One can argue that in a real-world scenario the user is also required to identify the content of a clip in order to determine suitable In and *Out Points*. That is, the authors would like to explain why they think their experimental setup is suitable and sufficient. First, the predetermined In and *Out Points* make the process reproducible and thus, comparable. Second, the applicability of the underlying interface for loose browsing (includes searching for a specific frame within a clip) was outlined in [22] and is not part of this study. This study focuses on the greater task of video trimming and on the comparison of different interface concepts for video trimming. Third, the given targets for In and *Out Points* are precisely defined on a frame-level. Browsing to a predefined frame reflects, to a certain degree, the process of searching in a clip. Fourth, taking the burden from the users to decide (artistically) on the eventual In and *Out Points* helps them to concentrate on the actual task of video trimming.

B. *Execution*

Six participants attend a school for multimedia and were between 18 und 22 years old. Four participants were professional video editors in the age of 27, 34, 35 and 39 years. Twelve participants (15 - 37 years) had no experience in video editing whereof six were students. The study was conducted on various places, chosen by the participants (school, university, working place, coffee house).

The following steps were explained to the participants: how the different interfaces work, how the given tasks can be accomplished and how the evaluation takes place. Every participant carried out four to six tasks from the trimming list. The tasks and which interface was used first (*iMovie* first *Muvee* second or vice versa) were randomly chosen for every participant. Eventually every task was carried out seven times both on *iMovie* and *Muvee*. The participants were filmed during the evaluation to retrieve the various time spans needed to fulfill each task. Subsequently the users were given a questionnaire and an open interview was conducted. The interview included a discussion for improvements on the presented interfaces. Trimming (browsing and marking) clip 1 is referred to as task 1 (t1); trimming clip 2 is referred as task 2 (t2) and so forth.

C. *Results*

Performance was measured across all tasks. Figure 8 depicts the different duration times for each task on the two interfaces in question. The x-axis shows the task-number whereas the y-axis indicates the seconds to fulfill a task. An analysis of variance (ANOVA) between the two tested interfaces, *Muvee* and *iMovie*, showed significant differences ($p < 0{,}001$) for all tasks except task 2 and task 3.

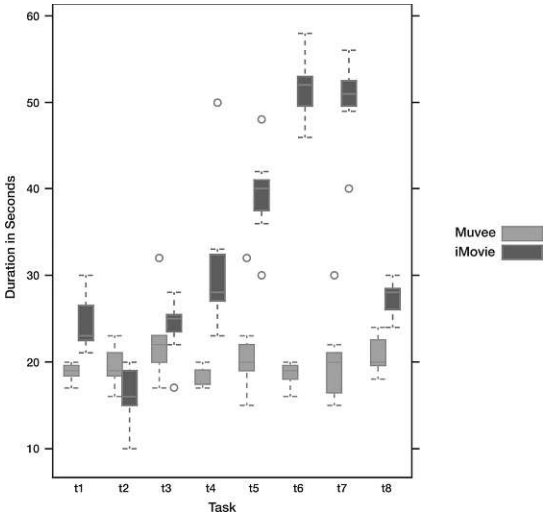


Figure 8. Time per task and interface

Comments made by participants during the evaluation were noted on-site and retrieved from the recordings afterwards. Analyzing the comments we could observe a tendency that participants favored *Muvee* over *iMovie*.

However, the intensity of the comments seemed to correlate to specific task (especially tasks 5, 6, and 7 provoked criticism, especially on behalf of *iMovie*). These comments could be explicable with the completion times on these tasks that were noticeable higher with *iMovie*. Below two comments expressed by the participants while working on the completion of task 6, P1 using iMovie and P2 using Muvee.

P1: *"Your are kidding me? This is the way it has to be done? This takes forever."*

P2: *"It feels like I need always the same time to complete a task no matter what."*

Both comments are backed by the numbers in Figure 8. The irritation of P1 due to the long execution time as well as the assumption of P2 concerning the constant execution time when operating with *Muvee*.

Two other representative statements made by the participants when commenting on *iMovie*.

P3: *"This icon is far too small. You only hit them by chance."*

P4: *"The structure of the interface is clear. But it does not support fast editing."*

In contrast, the participants commented on *Muvee* quiet favorable.

P5: *"This interface feels much more efficient."*

P6: *"I don't know what it is, but this [interface] seems more practical."*

The general tendency of the statements was that *Muvee* is more supportive and efficient when trimming different video clips. However, it was also mentioned several times by the participants that *iMovie* offers more functions needed for video editing.

After the tasks were carried out a small questionnaire was made to gather structured information about the perceived user experience.

TABLE I. PARTICIPANTS INTEREST IN VIDEO EDITING

Question	*yes*	*no*
1. Do you edit videos on desktop computers?	12	10
2. Did you ever edit videos on a mobile device?	7	15
3. After this evaluation, do you could imagine to edit videos on mobile devices?	15	7

TABLE II. DIRECT COMPARISION OF EVALUATED INTERFACES

Question	*iMovie*	*Muvee*
1. What interface do you think is faster for general purpose video editing?	4	18
2. What interface is more fun to use?	2	20

The questionnaire hinted that the demand for video editing on mobile devices is still low in general even some people have genuine interest in it. The second finding of the questionnaire was the rather unambiguous favoring for one of the two interfaces in question. These were the two main issues of an eventual open discussion that focused on video editing in general and video editing on mobile devices in particular. The interviews revealed that photo and video creation on mobiles devices is done on a regular basis. While editing or modifying photos is done frequently most of the participants do not edit videos because of a perceived usability complexity. Especially users with little or no experience in video editing are easily overwhelmed by the complexity of such programs. On the other hand experienced video editors are accustomed to the editing software they know from desktop computers. However, during the discussion the participants agreed on the idea that video editing on mobile phones is different to editing on desktop computers. The interviews brought up that mobile video editing can be seen as a more spontaneous act and thus, does not necessarily require all the features expected from desktop video editing. As a field of application the participants mentioned a simple trimmer or loop generator for the video platforms like Vine. While the younger or inexperienced participants tended to propose more light-weight or funny video applications such as extracting a frame from a video clip, the older or more experienced participants discussed the option for a downgraded but functional mobile video editor.

D. Discussion

At this point it has to be made clear again that *iMovie* is a full video editing application whereas *Muvee* is an interface for browsing and trimming clips. However, *Muvee* is not seen as competing to *iMovie*, rather an alternative concept worth being tested against a well-known and established design. Furthermore, the tasks were designed with these differences in mind to avoid bias for one interface/concept.

While the conducted questionnaire and the interview with the participants revealed that video content creation on mobile devices has increased and is an issue for users, applications have not matched this trend for carrying out video post-production in-situ. Spontaneous video editing is not commonplace although users of mobile devices would like to do so.

During the quantitative evaluation *iMovie's* task execution times strongly depended on clip length, trimming sequence length and position of the trimming sequence (trimming parameter) whereas the execution times on *Muvee* were almost constant over all tasks (trimming parameters). Even though *iMovie* was faster in task 2 and comparably fast in task 3 the comments of the participants do not reflect that fact. In contrast, the participants preferred the interaction technique of *Muvee* during all tasks suggesting that slightly slower interaction is not considered a big loss when the overall interaction is provides a better user experience. Due to the limited screen space participants noted that they would prefer multiple but specialized screens for video editing instead of one like in *iMovie*. One screen per task leaves more space for bigger interaction elements, which complies with Fitts' law. On the other hand, *Muvee* tends to use an entire screen for gestures and thus gesture span over comparable distances, which is in disfavor to the user, according to Fitt's law. Nevertheless, the distance between interaction elements does not seem to have great influence on efficiency on small screens. In contrast, it seems to

minimize user errors and thus, leading to increased user satisfaction.

Splitting one complex task into several simple tasks and allocating an entire screen for each simple task appears a legitimate approach when implementing complex tasks for mobile devices. Nonetheless, convenient and practical gestures must be found to switch between screens within a single application. The evaluation suggests that interaction models for desktop usage do not necessarily work equally well on mobile devices with small screen factors. Furthermore, it seems that experienced users (in our case, media designers) do not necessarily expect mobile applications being similar to their desktop equivalent in terms of interface and interaction design. And, on the other hand, unexperienced users do not have the knowledge about existing interfaces they can refer to.

However, the participants liked *Muvee* as an alternative approach to mobile video trimming. They especially mentioned the clear separation of the subtasks like browsing and marking. Stating from the questionnaire more users would trim their videos more often or start doing it, if interfaces are available that are easy to use. Professional users mentioned that they understood the idea of deconstructing the timeline metaphor for mobile video editing. The study indicates that even complex applications can offer efficient user interaction on mobile devices when the particular design is carefully thought of. The main challenges will be to overcome the limitations of small screens and to fully exploit the possibilities of mobile devices.

V. CONCLUSION AND FURTHER WORK

This paper presents and evaluates *Muvee*, an alternative approach for video trimming on mobile devices with small screens. The design goals are *efficiency* and *ease of use*. To fulfill the goals relevant prior works and concepts are introduced and, in addition, similarities, differences and novelties within the different concepts are discussed in depth. Based on the comparison and discussion a novel interface for video trimming (*Muvee*) is proposed and implemented. Some of the design decisions for *Muvee* are in accordance and some are in contrast to established interface guidelines like Fitt's law.

To gather data about the viability of the proposed interface a set of tasks was designed and participants were timed when carrying out the tasks. Furthermore, the participants were filmed during the tasks and the comments made were transcribed. After the tasks were accomplished a questionnaire was given and an open discussion was conducted. Despite the fact that *Muvee* was not faster on all tasks (in comparison to existing interfaces) all users felt more comfortable with the interface *Muvee* offered. Due to the complexity of the task itself the comments brought up that all interfaces needed explanation in advance. However, *Muvee* demonstrated that even complex interactions can be implemented for small touch screen devices and still offering a positive user sensation. The results show that existing concepts for mobile video trimming can be improved by incorporating contemporary research.

More and more complex multimedia applications are transferred to mobile devices with small screens and limited interaction possibilities. Finding viable concepts and interfaces for such mobile applications will be *one* challenge for future research. Encouraged by the survey we plan to explore further developments to expand the limited capabilities of *Muvee*. Moreover, research should study the limitations of desktop concepts and metaphores for mobile devices like Fitts' law and filmstrips.

VI. ACKNOWLEDGMENTS

We thank all the volunteers who wrote and provided helpful comments on previous versions of this document. Our special thanks to Susanne Stigberg for her help with the statistics portion of this document.

REFERENCES

[1] A. Puikkonen, J. Häkkilä, R. Ballagas and J. Mäntyjärvi. 2009. Practices in creating videos with mobile phones. In *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services* (MobileHCI '09). ACM, New York, NY, USA, , Article 3 , 10 pages.

[2] D. Kirk, A. Sellen, H. Harper and K. Wood. 2007. Understanding videowork. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '07). ACM, New York, NY, USA, 61-70.

[3] H. Väätäjä. 2010. User experience evaluation criteria for mobile news making technology: findings from a case study. In *Proceedings of the 22nd Conference of the Computer-Human Interaction Special Interest Group of Australia on Computer-Human Interaction* OZCHI 2010. ACM, New York, NY, USA, 152-159.

[4] A. Lehmuskallio and R. Sarvas. 2008. Snapshot video: everyday photographers taking short video-clips. In *Proceedings of the 5th Nordic conference on Human-computer interaction: building bridges* (NordiCHI '08). ACM, New York, NY, USA, 257-265.

[5] S. Vihavainen, S. Mate, L. Seppälä, F. Cricri and I. Curcio. 2011. We want more: human-computer collaboration in mobile social video remixing of music concerts. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Vancouver, Canada, May 07–12, 2011). CHI 2011. ACM, New York, NY, USA, 287-296.

[6] L. Terrenghi, T. Fritsche and A. Butz. 2008. Designing Enviroments for Collaborative Video Editing. International *Conference on Intelligent Environments* (Seattle, USA, July 21 - 22, 2008). IET'08.

[7] N. Bornoe and L. Barkhuus. 2010. Video microblogging: your 12 seconds of fame. In *CHI '10 Extended Abstracts on Human Factors in Computing Systems* (CHI EA '10). ACM, New York, NY, USA, 3325-3330.

[8] H. Jokela, H. Karukka and K. Mäkelä. 2007. Empirical observations on video editing in the mobile context. In *Proceedings of the 4th international conference on mobile technology, applications,* MC'07. ACM, New York, NY, USA, 482-489

[9] V. Zsombori, M. Frantzis, R. L. Guimaraes, M.F. Ursu, C. Cesar, I. Kegel, R. Craigie and D.C.A. Bulterman. 2011. Automatic generation of video narratives from shared UGC. In *Proceedings of the 22nd ACM conference on Hypertext and hypermedia* (Eindhoven, The Netherlands, June 06–09, 2011). HT'11. ACM, New York, NY, USA, 325-334.

[10] E. Laurier, I. Strebel and B. Brown. 2008. Video Analysis: Lessons from Professional Video Editing Practice . In *Forum Qualitative Sozialforschung / Forum: Qualitative Social Research*, 9(3), Art. 37

[11] A. Puikkonen, L. Ventä, J. Häkkilä and J. Beekhuyze. 2008. Playing, performing, reporting: a case study of mobile minimovies composed by teenage girls. In *Proceedings of the 20th Australasian Conference*

on Computer-Human Interaction: Designing for Habitus and Habitat (OZCHI '08). ACM, New York, NY, USA, 140-147.

[12] E.d.C. Valderrama-Bahamondez, J. Kauko, J. Häkkilä, and A. Schmidt. 2011. In class adoption of multimedia mobile phones by gender - results from a field study. In *Proceedings of the 13th IFIP TC 13 international conference on Human-computer interaction*. Lisbon. Portugal

[13] A. Islam, F. Chebil, A. Hourunranta. 2006. Efficient Algorithms for Editing H.263 and MPEG-4 Videos on Mobile Terminals. In Proceedings of IEEE International Conference on Image Processing, p. 3181-3184.

[14] A. Hourunranta, A. Islam, F. Chebil. 2006. Video and Audio Editing for Mobile Applications. In Proceeding on IEEE International Conference on Multimedia and Expo. p. 1305-1308.

[15] T. Jokela, M. Karukka. and K. Mäkelä. 2007. Mobile Video Editor: Design and Evaluation. In *Proceedings of the 12th International Conference on HCI*. Beijing, China, Part II. 2007, pp 344-353

[16] Adobe, Inc. 2013. Adobe Premiere Pro Help / Trimming clips (CS5 and CS5.5), http://helpx.adobe.com/premiere-pro/using/trimming-clips.html

[17] G. Rebholz. Three Approaches to Basic Editing In Vegas Pro. http://www.sonycreativesoftware.com/basic_editing_in_vegas_pro

[18] Apple, Inc. 2013. Apple Appstore - iTunes Charts for September 2013, https://www.apple.com/itunes/charts/paid-apps/

[19] N. McAllister. Apple seeds final iOS 7 code to devs, announces September 18 ship date. Not good enough? How about some free apps, as well? 11th September 2013. The Register. http://www.theregister.co.uk/2013/09/11/apple_ios_7_ship_date/

[20] The Wall Street Journal Online. Q&A With Samsung's Mobile Chief - Korean Giant Unveils Galaxy Phone; Mobile Chief Isn't Satisfied With U.S. Share. 15 March 2013. http://online.wsj.com/article/SB10001424127887324077704578358283252725470.html

[21] zeoxy Software. AndroVid. 2013. https://play.google.com/store/apps/details?id=com.androvid

[22] R. Ganhör. 2012. ProPane: Fast and Precise Video Browsing on Mobile Phones. In *Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia* (MUM '12). ACM, New York, USA.

[23] P.M. Fitts. (1954). The information capacity of the human motor system in controlling the amplitude of movement. Journal of Experimental Psychology, 47,381-391

[24] J. Accot and S. Zhai. 1997. Beyond Fitts' law: models for trajectory-based HCI tasks. In Proceedings of the ACM SIGCHI Conference on Human factors in computing systems (CHI '97). ACM, New York, NY, USA, 295-302

[25] P. Holleis, F. Otto, H. Hussmann, and A. Schmidt. 2007. Keystroke-level model for advanced mobile phone interaction. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '07). ACM, New York, NY, USA, 1505-1514

[26] T. Schulz. 2008. Using the Keystroke-Level Model to Evaluate Mobile Phones. Trolltech ASA. University of Oslo. http://hdl.handle.net/10852/9883

[27] W. Hürst, K. Meier and G. Götz. 2008. Timeline-based video browsing on handheld devices. In Proceedings of the 16th ACM International Conference on Multimedia (MM '08). ACM, New York993-994.

[28] H. Cho, H. Choi, W. Jun, C. Kim, H. Kwon, S. Yeom. 2012. Mobile terminal and method for controlling playback speed thereof. European Patent Agency. EP 2434490 A2. http://worldwide.espacenet.com/publicationDetails/biblio?CC=EP&NR=2434490A2&KC=A2&FT=D

INSERT: Efficient Sorting of Images on Mobile Devices

Roman Ganhör
Multidisciplinary Design Group
TU Wien, Austria
roman.ganhoer@tuwien.ac.at

Florian Güldenpfennig
Human Computer Interaction Group
TU Wien, Austria
florian.gueldenpfennig@tuwien.ac.at

ABSTRACT
We amass increasing amounts of photos on our mobile devices, primarily captured by built-in cameras. These cameras provide precious opportunities to preserve memories or serve for creative engagement. However, creating order over these vast photo collections gets more difficult as we create more and more photos and this puts these valuable resources at risk. People fail to sort their photo collections manually and automated algorithms are not yet able to identify and group images based on the features that are most relevant to the human beholder. For these reasons we present INSERT, a novel mobile phone application for supporting manual sorting of photo collections in an efficient fashion. A user study featuring 21 participants showed that the proposed interaction mechanisms were well perceived and that there is yet much research to be conducted aiming at the management of image collections on mobile device, in particular with small screens.

Author Keywords
Mobile phones; photography; image management; file management; Design, Experimentations, Human Factors.

ACM Classification Keywords
H.5.2 [**Information Interfaces and Presentation (e.g. HCI)**]: User Interfaces – *Graphical user interfaces (GUI), input devices and strategies, interaction styles, screen design*

INTRODUCTION
While the numbers of photos captured with mobile devices grows rapidly, people usually do not sort or manage their assets (Whittaker et al., 2010). However, operating systems like Android or iOS offer automatized grouping based on location or time (Elliott, 2014; Ybanez, 2014). In addition, there is intensive research to advance sophisticated algorithms in pattern recognition to pool images, e.g., photos depicting the same persons (Darwaish et al., 2014). Even though automatized grouping can be a powerful and convenient feature, it often does not satisfy the users' intentions or needs. All too often the discrepancy between user expectation and automated results lead to frustrating user experiences. Thus, it comes as little surprise that there are also less 'mechanical' and much more playful approaches to exploring photo collections, e.g., as presented by Ott et al., 2012.

OzCHI '15 , December 07 - 10 2015, Melbourne, VIC, Australia
Copyright © 2015 ACM 978-1-4503-3673-4/15/12... $15.00
http://dx.doi.org/xx.xxxx/xxxxxxx.xxxxxxx.

As an example for a task that is hard to automate, consider a user compiling a photo book as a keepsake. The most *precious* photos for such a personal compilation cannot easily be determined by a computer algorithm as this involves questions of personal preference, relevance, and the wealth of memories. Algorithms may satisfyingly recognize photographical standards like exposure time or even classify different objects and persons, but can these features make a photo book, which tells the story that we want to be told?

Another relevant example would be an architect, who wants to group all photos made during the process of planning and building a house. These photos will probably depict diverse content, such as sketches made during discussions, small-scale models of the house, various states of the construction progress and finally the completed house. However, the photos just mentioned do not share common attributes such as date, place or content. Thus, automated algorithms will probably fail when grouping the *right* photos for a specific architectural project.

Switching from still photography to video editing, storytelling also fulfills an important and integral part in the searching and sorting process. In video editing it is not only video grouping that matters, it is also the order of the footage that is important for narrating a story (Barrett, 2008). Again, algorithms exist in products (Muvee, 2015) and are presented in academic research (Zsombori et al, 2011) that are mimicking the human ability of storytelling. However, these algorithms will hardly be able to obtain the desired artistic goal as expected from films or videos and thus, automatized video editing is mainly employed by 'leisure artists' to avoid the burden of video editing.

Research focusing on manual and automated photo sorting and grouping, as well as work on managing videos on desktop computers, laptop computers or even table computers is intensively discussed in academia (Hilliges et al., 2007), and a plethora of differing applications exist (Laurie, 2014). Nevertheless, research and applications for touch based handheld devices is still rare in this area, dealing mostly with novel alternatives to browsing large multimedia collections (Dragicevic et al., 2008; Sun et al., 2008) or particularly dealing with the affordances of small screens (Patel et al., 2004). While

browsing is an important task, its outcome is very volatile without the possibility to store the browsing results. Storing browsing results can depict useful information as it groups digital assets according to an overall user objective.

As the quality of mobile devices in terms of cameras and displays is ever increasing and as technology like cloud computing allows users ubiquitous access to their media assets, we face new challenges. Mobile devices become a common and preferred means to present, receive and discuss multimedia data. Thus, nowadays users carry out tasks in the mobile context they were not able to do before, such as manually sorting and grouping digital assets.

In this paper, we evaluate existing research in the field of mobile multimedia and combine several aspects for sorting and grouping multimedia assets on mobile handheld devices in our proposed interface. The final interface, INSERT, is designed for *fluent* and *efficient* use and was qualitatively evaluated with 21 participants depicting its potential strengths and weaknesses. In more detail, we state the research problem of this paper as follows.

Research Statement

Considering the ever-increasing amount of media assets made with and stored on mobile devices, the importance of these assets' order for the purpose of storytelling and the lack of academic research were the main motives for a closer examination of this research area. Hereby a main challenge for feasible and effective interfaces on mobile devices is leveraging their possibilities while evading their constraints (Xiao et al., 2010), e.g., balancing the image (*thumbnail*) size in dependency to the screen size. Here, the opposing goals are having big thumbnails for good content perception versus as many thumbnails as possible on the screen at once for a good overview.

As indicated by the set of different application examples from above (photo book, documentation of architecture, editing and searching in video), we don't target a specific application area. That is, in this paper, we are not so much interested in, say, supporting specifically creating photo books or photo documenting. Rather, we seek to operate on a more *abstract* level to investigate and design for the elementary task of the searching for and sorting of images on mobile devices (be it personal photos, video frames, etc.). Thus, the objective of this paper is to evaluate the basic interaction mechanisms as proposed by INSERT, independent from too specific application domains. This qualitative feedback again we intend to use for further design iterations of this software. We go on to motivate INSERT drawing on related background literature.

BACKGROUND

Smartphone interfaces vary in many qualities from desktop interfaces due to various evident differences such screen size or input modalities. However, smartphones are increasingly becoming more powerful and users are starting to carry out even complex tasks on their mobile devices. Research in the area of multimedia applications stretches out in various directions: examining the interaction possibilities and affordances of touch-based devices or investigating and extending the boundaries of the limiting factors.

Affordances of Touch Based Devices

In contrast to desktop computers mobile devices are not stationary and hence, can be brought to different contexts. Modern sensor technology enriches mobile applications and enables novel aspects in interaction design. Mobile devices can potentially be used anytime and anywhere to fulfill various kinds of tasks. Users can employ their fingers for 'direct manipulation', i.e., there is no mapping necessary between hand and cursor. The user's finger marks the spot where the input and reaction take place, i.e., the most direct feedback is delivered to the user.

Limiting Factors

Despite the unquestioned possibilities smartphones offer, they suffer from some inherent limitations. One of the most apparent advantages turns out being a major disadvantage with respect to interface and interaction design - *the size*. Both the human eye and screen display have a limited resolution and interface elements must feature a specific size to be (easily) perceivable. Since the size of the smartphone itself is restricted compared to, e.g., a LCD display, the number of elements per screen is limited as well. Due to this constraint even simple interactions often have to be distributed across several screens. Hence, maximizing screen elements, avoiding clutter and providing a smooth user experience is an area of conflict for designers and researchers (Gong et al., 2004).

Thumbnails generally represent video clips and pictures, depicting a small copy of the original content. Studies about the size of thumbnails show that users can identify the content on even comparatively small thumbnails (Hürst et al., 2011). However, while thumbnails are useful for identifying the general content of an asset, they are not appropriate for identifying detailed differences between assets. On one hand, the smaller the thumbnails are the more content can be displayed at once. On the other hand, the bigger the thumbnails are the better the details of one thumbnail can be identified.

Keyboard and mouse are among the most common input devices for desktop computers. The strength of the mouse is its resolution and accuracy, besides, the mouse pointer does not occlude elements on the screen. Experienced users often use keyboard shortcuts as a very efficient input method when interacting with a computer. Touch based devices in contrast have to deal with more inaccurate input methods (Forlines et al., 2007), as the thumb cannot reliably be mapped to a pixel or even a small pixel area. The limiting factor is the size of a fingertip. Additionally, it is difficult to implement shortcuts for touch-based devices without restricting other interaction metaphors.

In contrast to most stationary computer systems, mobile devices allow handling in two orientations, horizontally and vertically. However, the 'natural' orientation for smartphones is vertically as the device can be grabbed

firmly and effortlessly with one hand. Depending on the size of the smartphone it can even be operated with one hand. Albeit the 'natural' orientation mobile applications can sometimes be more effective when used horizontally. Besides rather trivial and obvious observations of existing application exist little guidelines that can be generalized for novel interaction mechanisms. Therefore it is up the designer to argue for a novel application's preferred orientation.

Sorting and Rearranging Images on Mobile Devices

Even though grouping thumbnails on mobile devices became a feature widely adopted exploiting meta-data, manually sorting, rearranging, and refining is still not supported by the standards apps of the major mobile platforms. Third party applications have to be purchased or free software has to be obtained to allow the user to rearrange their media items according to their personal preferences. *F-Stop Media Gallery* for Android, for example, allows the user to pick up a media item with a long press gesture, move it to the favored position and drop it wherever wanted.

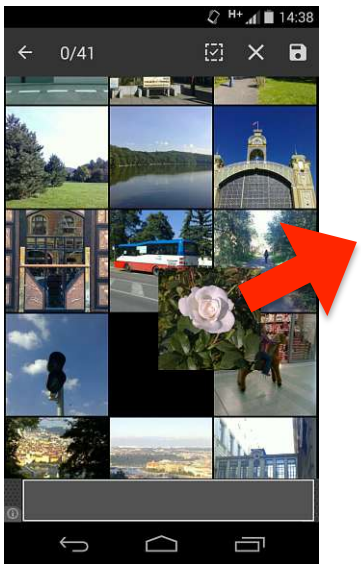


Figure 1: Rearranging photos with a standard grid layout. The rose-picture is dragged into the direction of the arrow.

This process is illustrated by Figure 1, which depicts the interface while inserting a media item (featuring a the picture of a rose) between two other media items. While this approach is "straight forward" and complies with the guidelines for touched based devices it still has a few drawbacks.

First, moving to the start or to the end in a long list of items can be tedious. A series of quick swipes can overcome this, however, this gesture cannot be done when moving a media item at the same time (i.e., fast browsing and drag&drop "don't mix"). The item would drop right after the first swipe when the finger leaves the surface preparing for the second swipe. Second, the interface does not provide an overview over the total collection of media items. This is especially true since the latest updates in the style guidelines of mobile interface manufacturers (e.g., for Android) omit the permanent presence of a scrollbar[1]. Third, the interface does not provide the process of creating new collections from compilations of media items sharing some aspects important for the user. In the common image viewer applications, dragging & dropping an image will result in moving this particular image to another location, not creating a copy. That is, certain pictures can only exist in one collection/folder at a time, and an additional file manager application is needed for instantiating a duplicate. In reality, however, it is a different picture. A photo of an ancient house taken during holidays, for example, can potentially be an interesting member to two different collections: *Holiday in France* and *Houses*. To overcome the first two limitations discussed above various approaches were proposed, some well known are described in the following subsection.

Adapting for Mobile Devices

Focus+Context is an established desktop computer approach for combining the inspection of details (large thumbnails) with the investigation of broader contextual information (small thumbnails) (Furnas, 1986; Rao et al., 1994) in one (distorted) photo mosaic. This approach is also named as *Fisheye View* as it reminds of a *fisheye lens* where just the center of an image is displayed proportionally correct while the outer bounds of the image are visually distorted. *Focus+Context* was also implemented for mobile devices, dating back to 1997 where some interaction features where built on PDA's (Holmquist, 1997). More recently, an application was created for the iPad, which projects thumbnails onto spherical objects (Ahlström et al., 2012) in a *Focus+Context* fashion (see Figure 2).

Overview+Detail is another approach dealing with the challenge of representing data on different levels of detail. The software divides the screen into two distinct areas. One area shows an overview over all or most part of the available information, whereas the second area displays details of a specific subset of information. Thus, an *Overview+Detail* interface design is characterized by the simultaneous display of both an overview and detailed view of an information space. Even though its interface is separated into two distinctive views, user interaction within one view is reflected in the other immediately (e.g., manipulations on the detailed level will be visible on the overview level).

The *Dominant Color Diagram* (Schöffmann et al, 2010) can be described as a combination of *Focus+Context* and *Overview+Detail*. It takes the concepts of *context* and *overview* and extend them to an 'extreme' level whenever necessary. The authors of the algorithm implemented an example where a television show was rendered by means of the *Dominant Color Diagram*, and each frame of the television show is represented by a one-pixel bar in the diagram. While the one pixel diagram does not convey

[1] http://developer.android.com/design/index.html

much information, it provides a valuable overview of the complete data set.

Figure 2: Spherical Projection - mobile example of *Focus+Context* (Ahlström et al., 2012).

Many variations and combinations of the aforementioned approaches exist for both, desktop and mobile computers. Tailoring the most appropriate interactions and combinations to a specific application depicts an important challenge for designers and engineers.

IMPLEMENTATION OF INSERT

The design and implementation of INSERT was motivated by overcoming the limitations of current implementations as described in the section above. This comprises in particular the tedious task of scrolling through a vast amount of media assets and the absence of contextual information. The interface of INSERT and its interaction design particularly considers the strengths and weaknesses of mobile devices and also builds on successful design elements as outlined in the previous section. Novel features of INSERT comprise the implicit definition of distinctive collections when ordering media items and the possibility to add one media asset to several collections in a (and this is one of our hypotheses) efficient and appropriate fashion.

In more detail, INSERT is a user interface for sorting and grouping media assets on touch-based smartphones. The application focuses on three tasks for sorting and grouping media assets that we identified as important in the literature and in our own research: *browsing*, *selecting*, and *filing*.

Browsing through photo collections with a detailed view while at the same time providing contextual information leverages the idea of *Focus+Context*. Thus, the interface provides a visualization of all photos whereas it also conveys detailed information about at least one specific photo at the same time. The interaction concept also supports fast browsing (context) when skimming through the complete collection and precise browsing (focus) when identifying a specific asset (see Figure 3 first and second row).

Selecting a single photo for sorting or grouping is the next step in the interaction cascade of INSERT. We carefully designed the interaction of selecting a media item in an unambiguous/precise fashion to deliver an enjoyable and productive user experience.

Filing photos completes the application as proposed by INSERT and refers to sorting and grouping. This interaction consists of two steps, dragging a photo from the original collection and dropping it to a *selection* (see Figure 3 for an illustration of a *selection*). Thereby, dropping a photo to a *selection* defines its position in this *selection*. In contrast to the drag and drop interaction metaphor, INSERT treats the original collection as 'read only'. This is motivated by two considerations. First, users generally have a good knowledge of their original collections, knowing their structure and thus, removing photos from the original collection could potentially lead to irritation. Second, leaving an item in its original place allows the user to add an item to more than one collection.

Additionally, three non-interactive design features were considered to optimize the user experience. First, most of the screen estate was used by the program leaving little space unused. Second, the interface was kept plain and simple without the need for additional menus or similar complex application structures. Third, sorting photos and compiling groups of photos were considered equally important tasks.

Interface

INSERT was implemented for Android mobile phones. The development device featured operating system version 4.4 and a screen resolution of 1280x720 pixels. The application is optimized for landscape mode for practical reasons. This orientation allowed us to make best use of the screen estate (e.g., in adjusting the overview bar) and, moreover, we are also interested in using INSERT for video assets where the default orientation is widescreen. It distinguishes two main areas (see Figure 3). One area allows users to browse their original collection (browsing) and a second area allows organizing (i.e., filing) the media items in individual *selections* (selecting).

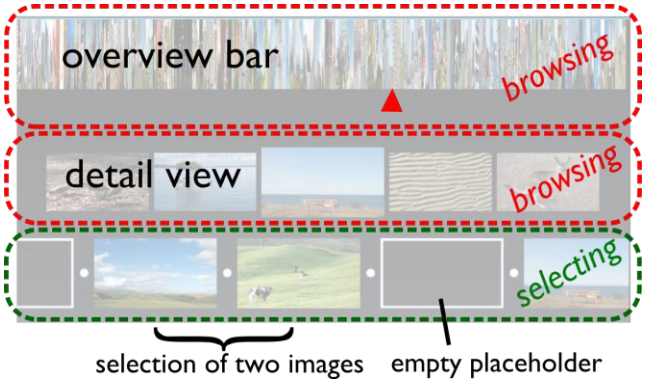


Figure 3: Different areas of the INSERT interface.

The browsing area is divided into two parts, one depicting an overview and the other exhibiting details. In the

overview all photos of the original collection are 'squeezed' into the width of the device. This feature was inspired, among others, by work from information visualization (Tang et al., 2009; Viegas et al., 2004). While this operation distorts the photos, it provides the user with rich contextual information. Underneath the overview section an arrow indicates the current position within the data collection, defining the focus or the area of interest of the user. The detail area shows five thumbnails as extracted from the photo subset marked by the arrow.

Figure 4: Screenshot of the implementation of INSERT.

Figure 4 shows a screenshot of the actual implementation of INSERT depicting the threefold division as described above. The upper third is filled with small bars, each bar representing a compressed picture. The second third contains five thumbnails big enough to perceive the content of the associating photo. The thumbnail in the center is slightly bigger then the thumbnails to the left and right and is associated with the position of the arrow. The last third comprises a slider holding the photos ordered and grouped in separate collections (*selections*). While the upper two thirds always depict the complete original collection, the lower third (slider) only presents a small fraction of the individual *selections*. The slider can be moved to the left and to the right. In Figure 4 the slider depicts a small *selection* holding two photos, one with a landscape and fluffy clouds on it and one showing green hills with cows. An empty slot defines the end of a *selection* respectively the beginning of the next *selection*.

Interaction

The interaction mechanisms consist of swipes for browsing, drag and drop for selecting and filing. Figure 5 illustrates the swipe and Figure 6 the drag and drop interaction.

Browsing: the task has two levels, coarse and fine. Coarse browsing denotes jumping to an arbitrary point within the thumbnail collection quickly (Figure 5, a), whereas fine browsing is to find a particular thumbnail within a narrow area (Figure 5, b). Browsing the *selections* is executed by swiping over the *selection* area (Figure 5, c).

For selecting and filing the thumbnails into individual *selections* the user drags the corresponding thumbnail from the detail area and drops it onto the *selection* area (Figure 6, d and e). A long click on any of the thumbnails marks a thumbnail for dragging and lifting the finger drops the thumbnail. If a thumbnail is dropped on a regular placeholder (Figure 6, d) that placeholder is filled with the thumbnail. If the placeholder already holds a thumbnail the existing thumbnail will be overwritten with the newer thumbnail. Thumbnails can also be dropped between placeholders (Figure 6, e), the area marked with a bullet. Whenever a thumbnail is dropped on a bullet, the bullet turns into a regular placeholder populated with the thumbnail and all thumbnails right of the new placeholder move one position to the right. This is described in Figure 6 when the selection order of w,x,y,z turns into w,x,y,m,z. To remove a thumbnail from a placeholder the thumbnail gets dragged and dropped as depicted in Figure 6f.

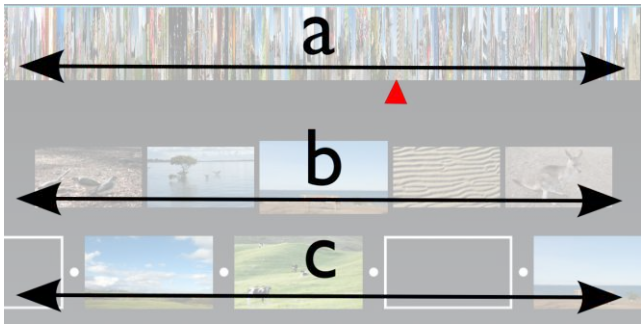


Figure 5: Interactions areas (a,b,c) for swipes gestures.

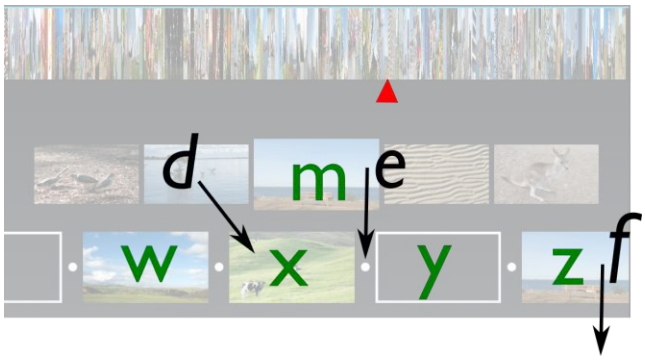


Figure 6: Interactions areas (d,e,f) for drag&drop gestures.

We suggest that the strength of this application is given by the simplicity of the interaction mechanisms while at the same time providing a powerful tool allowing the users to conveniently arrange and re-arrange all photos stored on their smartphones. Since only thumbnails in the *selections* can be deleted, the original set of thumbnails is not altered. INSERT aids re-arranging the order of thumbnails in a media collection and the pooling of thumbnails in *selections*. Thereby, two empty placeholders, one leading and one trailing a set of thumbnails automatically define a selection.

EVALUATION

The design of the interface and the interaction mechanisms were built on prior work as presented above and we followed common design guidelines, and best practices in building the system. Nevertheless, INSERT depicts a novel design concept and consequently we were keen on gathering user feedback on the interface and the interaction. We decided for a qualitative evaluation

approach to gain *holistic* insights about this new mechanism for sorting images on smartphones and recruited 21 participants in the age from 16 to 45 through our extended social network. Evaluations sessions were video taped and the participants' comments were transcribed for later analysis.

Study Setup / Procedure

Each evaluation session was initialized with a short questionnaire about the creation and consumption of media data on mobile devices, especially photos. Furthermore, we surveyed their habits regarding mobile phones and how they make use of the capabilities of their devices. Then the participants were explained a use-case tailored to their personal habits, interests and capabilities that involved the re-organization of their media assets. The use-cases were formed on the fly and asked as a question or presented as an argument, like: *Wouldn't it be nice to have a best-of collection of your kid seeing it growing?* or *With the right app you easily could make various collections of different styles of graffiti*. The use-cases were meant to create interest in an application that can solve their brand-new demands. Subsequently the participants received instructions for INSERT and its user interface and were invited to 'play with' 150 prepared pictures. While doing so they were encouraged repeatedly to speak-out loud about their spontaneous thoughts. The whole process, including questionnaire and familiarizing, lasted six to thirteen minutes. As the interface of INSERT waives rich affordances (it is designed for users with some experience and to function as an *efficient* sorting tool) the participants needed an explanation of its interaction mechanics. The length of this training period depended on the participants' familiarity with such tools or mobile phone apps in general. However, after the initial training all participants' were able to utilize the interface sufficiently.

After the participants felt comfortable with the interface an elaborated exercise was given. The purpose of this exercise was to provide a set of real life picture-sorting tasks involving all features offered by INSERT. As such, the participants were asked to identify all pictures containing a red car, and insert them into a new photo *selection*. Subsequently, they had to move particular images (photos that also contained human beings) of the *selection* to yet another *selection*, and eventually delete some of these pictures (containing cars by a specific manufacturer). Again, the participants were encouraged to speak out loud their thoughts, positive or negative. The comments of the participants were recorded and used for later analysis (see next section). The exercises lasted additional 10-15 minutes and subsequently the participants were asked about their concluding opinion about INSERT and for suggestions for further improvements.

Analysis

For analysis, we used an adapted approach to thematic analysis as described by Braun and Clarke (2006). That is, we transcribed the recorded user tests and iteratively coded this data to let prominent themes emerge. Hence, this was an inductive process of coding. Still, it also involved deduction as we were interested in a specific set of a priori categories (and we also framed our questions to the participants accordingly), for example, the *mobile creation and consumption behavior of the participants* or the *overall user experience* (see next section). As stated above, other themes or patterns emerged inductively; in particular, in the discussion section we draw on such exemplary salient user comments that we identified as *important* during analysis. *Importance* or *salience* is defined by the interpretation and judgment of the authors, i.e. comments or patterns don't have to occur often to be relevant.

Study Results

Mobile Creation and Consumption Behavior of the Participants

All participants owned a smartphone with photo capability and used their device on a regular basis for general purposes. Nine participants captured at least five photos per week (*power users*). Seven participants utilized the camera of their smartphone at least once a week (*casual users*) and five participants used their smartphone camera less than once a week (*irregular users*). The average age of the three user groups is depicted in Table 1.

	Mean	Mean Dev.
power-user	28,44	5,72
casual user	29,28	8,04
irregular user	34,2	8,64

Table 1: Mean age and mean deviation for user groups

Sorting and Grouping Habits of the Participants

When asked for their photo managing efforts only 5 of the 9 power users (5/21 total) stated that they did some kind of sorting, all other participants denied grouping images etc. However, after the evaluation session each participant was asked for their interest in using software like INSERT for photo management. 15 out of 21 now indicated that they wanted to receive and use a private copy of the software (see Table 2).

	yes	no
power-user	8	1
casual user	4	3
irregular user	3	2

Table 2: Number of participants interested in an app like INSERT for grouping and sorting (after the evaluation session)

Immutable Original Collections

Based on the presumption that users know the approximate structure of their photo collection, the authors decided to leave the original collection immutable, i.e., not removing pictures or changing their order. Even in the relatively short familiarization period (for the 150 images) of our study it was evident that the users quickly gained a 'feeling' for the photo structure of the unaltered photo collection. When asked about their judgment on this design decision, 10 out of 21 participants thought it was very useful (see Table 3). Despites this rather mixed feedback (at least on the first glance), *immutable collections* probed valuable feedback and was heavily commented. This will be discussed in the next section.

	yes	no
power-user	5	4
casual user	3	4
irregular user	2	3

Table 3: Number of persons preferring for immutable original collections.

Overall User Experience

After the 'hands-on' evaluation the participants were asked about the overall user experience they had. This question also probed a lot of comments and resulted in rather polarizing opinions (*rather good* versus *rather bad*) as summed up in Table 4. We go on to discuss this finding in the following section.

	rather good	*rather bad*
power-user	7	2
casual user	4	3
irregular user	3	2

Table 4: Final judgment of INSERT's user experience

DISCUSSION

In this section the authors discuss their design decisions against the outcome of the evaluation. For the qualitative evaluation small exercises were designed and the participants were encouraged to think-out aloud (see above). Consecutively, we asked for ideas, changes or improvements regarding the interface and interaction design concept. These comments of the participants were pooled and assigned to a design decision. By doing this, the authors collected statements favoring or refusing single design decisions.

The initial idea for INSERT was to provide a useful and easy to use tool for arranging photos on mobile devices. Providing overview while supporting precise interaction mechanism was another important concept. Allowing the sorting and grouping of *all* photos on one *single* screen (i.e., not spanning several screens on the mobile phone) was maybe the single most important and controversially discussed design choice. In contrast, allowing the user to put a media item in more than one collection was a rather unanimous decision. In the following paragraphs we discuss comments, notes and supplementary ideas we found relevant. They are represented by salient comments we identified during analysis.

"The app should mark the items [in the original collection] that are already in one of the selections." (P3) Most participants, even those, who did not prefer immutable original collections, proposed to *mark items* in the original collection. Proposals ranged from using a colored frame around an item and highlighting its usage in a *selection* to placing small numbers in a corner of the item indicating the number of the associated *selection*. This improvement could aid in identifying specific photos in larger *selections*.

Design decisions were always made in favor for an effective and fluent interaction. Even obvious drawbacks of a decision were taken into account to push efficiency. One such decision was allowing thumbnails in the selections to get overridden by another thumbnails without any request. The underlying motivation of this was that users could easily get annoyed when rearranging an existing selection. Not surprisingly, several participants made comments like *"Oh, that [thumbnail in the selection] gets overridden very easily. Maybe this should be confirmed?" (P7)* However, when the authors discussed this issue with the participants many of them agreed that recurring confirmations would be annoying. A reasonable compromise could be an *undo function*.

The *selections* as displayed in the last row in the interface are separated by one empty placeholder (Figure 6, marked as 'y'). When scrolling through *selections* the termination placeholder can easily be missed and the user (unknowingly) navigates to the next *selection*. P19 as well as several other participants noted that *"... this stripe on the bottom holding the selection [...] should stop automatically whenever a new selection starts."* (P19) The drawback of such an automatic stop after each *selection* could result in tedious re-swiping to go from one *selection* to the next. To overcome this several approaches were discussed. A promising idea proposed by one participant was a combination of simple gestures.

"Always a long click ... why do drag and drop always need a long click?" (P4) Filing a thumbnail from the original collection to the *selections* was completed by a drag and drop gesture. We initially decided to stick to the original Android metaphor for dragging a screen item (long click on the screen item). As it turned out the participants accustomed to the workflow very quickly and felt hampered by the comparable time consuming and non-productive long click. A few participants (P1, P4, P13) highlighted that the swipe interaction for fine browsing is horizontal, whereas adding a thumbnail to a *selection* requires a vertical gesture. Thus, these gestures will hardly interfere mutually. However, an unfamiliar

implementation of a familiar gesture could potentially lead to confusion on the users' side.

As users interacted with INSERT they quickly got used to its capabilities. Six users (five of them power-users) tried to probe its limits and created dozens of *selections*, each containing dozens of thumbnails. When sliding left and right in the *selection* area users can easily get confused about their overall position and their position in a *selection*. To overcome this drawback various proposals were made. One proposal recommended a pair of sliders indicating the overall positions and the position in the *selection*. Another proposal was put into the words: *"[m]inimize the selection and represent it with just one [thumbnail]" (P11)*. The metaphor for minimizing could be implemented by one double click. Whenever a double click is carried out on a thumbnail in a *selection* the *selection* collapses into one thumbnail. This one thumbnail is marked to identify it as a representative for a whole *selection* (e.g., by a red border frame). A double click on such a representative thumbnail again would expand a collapsed *selection*. However, the double click gesture was also proposed for maximizing a thumbnail to full screen mode or for naming a *selection*.

"This [...] would come in handy for video editing" (P7) These considerations regarding the handling of *selections* (see previous paragraph) led to some interesting reflections around possible application domains that go beyond simple image sorting. As one of the participants, a professional video editor, mentioned, he was regretting the lack of feasible video editing software for mobile devices. In his opinion, and after some further investigations into this matter we do agree, that the easy way to duplicate and insert photos of INSERT, might also be valuable for mobile video editing apps. In this domain, it is often required to create multiple copies of short video clips or clip fragments to be further processed and *inserted* into the target movie. Thus, for future work, it may well be worth the time investigating whether mechanisms of INSERT can be used to create video editors for the mobile devices.

"But what will happen if the overview bar is too small for my photo collection?" (P6) P6 realized an important limitation of the current implementation of INSERT when she pointed out that the overview bar wouldn't be able to contain an infinite number of images. While this fact is true, this limitation can be avoided by implementing established UI ideas and concepts. An adapted version of *Focus+Context* can increase the amount of photos displayed, i.e. instead of *every* photo every second (third, fourth, etc.) photo is represented with one bar. To indicate these omissions between images, the corresponding bar is indicated by a reduced height (i.e., bars representing more images are shorter). Since the user can still browse through the photos one by one this is no considerable drawback for the usability, and the interface still provides a contextual overview.

Broader Reflections

The partly passionate discussion with the participants during our evaluation showed their strong interest and the potential for advanced multimedia tasks on smartphones. As we kept the initial design simple and concentrated on the basic tasks for sorting and grouping several questions arose about future extensions. Participants also asked for the integration of INSERT into other mobile application, expanding their functionality. This however touches upon one of the essential questions in mobile application design: should we concentrate on one single task like sorting/grouping or should we provide additional features, as well? The study results from this paper indicate that for the purpose of image sorting the first option should be favored, even though this was not always verbalized by the participants. Future work is needed for answering this question of complexity more reliably.

FUTURE WORK

Choosing a qualitative interview and user narrative approach when investigating INSERT allowed us to understand the way participants experienced the application. As discussed before, compressing all available images into one overview bar (see Figure 3) turned out to be an effective and appealing feature, if not the most interesting feature according to the participants. For this reason, we plan to further investigate this interaction mechanism employing *quantitative* methods to complement our qualitative data. Thus, we are currently implementing an alternative application identical to INSERT, named *InsertGrid*, however, with a conventional horizontal scrollable grid of photos instead of the overview bar. Hence, *InsertGrid* only shows a small subset of all available photos simultaneously. We will compare INSERT against *InsertGrid* and evaluate a set of benchmarks such as time to complete and total errors, allowing us to describe the efficiency of the overview bar statistically while keeping all other variables constant. We favor evaluating isolated UI elements of INSERT systematically (e.g., the overview bar; the single interaction steps) over comparing INSERT with an existing image sorting applications (“gold standard”) as it yields more insights about the various elements of INSERT. Additionally, investigating various variables at the same time (i.e., comparing INSERT with a gold standard) exacerbates identifying and explaining causalities. In parallel, we intend to advance our application with respect to two concrete steps.

First, we started implementing some of the participants’ comments as reported above into a new version of the application. Second, in this iterated version we also integrated online logging capabilities, i.e., user interactions are sent to our interaction log server. We plan to deploy INSERT to Google’s app store to gather a larger data set of participants who will use the proposed interface on their own devices. This additional data will complement our evaluation with quantitative observations and external validity. A final issue will be the question whether or not the concept of INSERT is able to scale to very large collections of photos and what adjustments to the interface and interaction design have to be made to support the sorting of many images.

REFERENCES

Ahlström, D., Hudelist, M. A., Schoeffmann, K., & Schaefer, G. 2012. A user study on image browsing on touchscreens. In Proceedings of the 20th ACM international conference on Multimedia (pp. 925-928). ACM.

Barrett, H. 2008. Digital storytelling. University of Alaska Anchorage. Retrieved February 1, 2015.

Braun, V ., & Clarke, V . 2006. Using Thematic Analysis in Psychology. *Qualitative Research in Psychology* 3, 2, 77-101.

Darwaish, S.F., Moradian, E., Rahmani, T., Knauer, M. Biometric Identification on Android Smartphones, Procedia Computer Science, Volume 35, 2014, Pages 832-841

Dragicevic, P., Ramos, G., Bibliowitcz, J., Nowrouzezahrai, D., Balakrishnan, R., Singh, K. 2008. Video browsing by direct manipulation. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (pp. 237-246). ACM.

Elliott, M. 2014. How and where to find your photos in iOS 8. cnet.com. http://www.cnet.com/how-to/how-and-where-to-find-your-photos-in-ios-8/

Forlines, C., Wigdor, D., Shen, C., Balakrishnan, R. 2007. Direct-touch vs. mouse input for tabletop displays. In Proceedings of the SIGCHI conference on Human factors in computing systems (pp. 647-656). ACM.

Furnas, G. W. 1986. Generalized fisheye views. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '86), Marilyn Mantei and Peter Orbeton (Eds.). ACM

Gong, J., Tarasewich, P. 2004. Guidelines for handheld mobile device interface design. In Proceedings of DSI 2004 Annual Meeting (pp. 3751-3756).

Hilliges, O., Baur, D., and Butz, A. 2007. Photohelix: Browsing, sorting and sharing digital photo collections. In Horizontal Interactive Human-Computer Systems. TABLETOP'07. Second Annual IEEE International Workshop on (pp. 87-94).

Holmquist, L. E. (1997, March). Focus+ context visualization with flip zooming and the zoom browser. In CHI'97 Extended Abstracts on Human Factors in Computing Systems (pp. 263-264). ACM.

Hürst, W., Snoek, C.G.M., Spoel, W-J. and Tomin, M. 2011. Size matters! how thumbnail number, size, and motion influence mobile video retrieval. In Proceedings of the 17th international conference on Advances in multimedia modeling - Volume Part II (MMM'11)

Laurie, V. 2014. Best Free Digital Photo Organizer. Gizmo's Freeware. http://www.techsupportalert.com/best-free-digital-photo-organizer.htm

Muvee Technologies Pte. Ltd. 2015. Muvee. http://www.muvee.com/products/muvee-reveal-11

Ott, C., Hebecker, R., & Wakes, S. 2012, Picture the space: three concepts for management and presentation of personal digital photographs. In *Proceedings of the 13th International Conference of the NZ Chapter of the ACM's Special Interest Group on Human-Computer Interaction* (pp. 1-8). ACM.

Patel, D., Marsden, G., Jones, S., & Jones, M. 2004. An evaluation of techniques for browsing photograph collections on small displays. In *Mobile Human-Computer Interaction-MobileHCI 2004* (pp. 132-143). Springer Berlin Heidelberg.

Rao, R., Card, S. K. 1994. The table lens: merging graphical and symbolic representations in an interactive focus+ context visualization for tabular information. In Proceedings of the SIGCHI conference on Human factors in computing systems (pp. 318-322). ACM.

Schoeffmann, K., Taschwer, M. and Boeszoermenyi, L. 2010. The video explorer: a tool for navigation and searching within a single video based on fast content analysis. In Proceedings of the first annual ACM SIGMM conference on Multimedia systems (MMSys '10). ACM, p247-258.

Sun, Q., Hurst, W. (2008). Video Browsing on Handheld Devices - Interface Designs for the Next Generation of Mobile Video Players. MultiMedia, IEEE, 15(3), 76-83.

Tang, Anthony, Saul Greenberg, and Sidney Fels. "Exploring video streams using slit-tear visualizations." CHI'09 Extended Abstracts on Human Factors in Computing Systems. ACM, 2009.

Viégas, Fernanda B., Martin Wattenberg, and Kushal Dave. "Studying cooperation and conflict between authors with history flow visualizations." Proceedings of the SIGCHI conference on Human factors in computing systems. ACM, 2004.

Whittaker, S., Bergman, O., & Clough, P. (2010). Easy on that trigger dad: a study of long term family photo retrieval. Personal and Ubiquitous Computing, 14(1), 31-43.

Xiao, J., Lyons, N., Atkins, C. B., Gao, Y., Chao, H., and Zhang, X. 2010. iPhotobook: creating photo books on mobile devices. In Proceedings of the international conference on Multimedia (pp. 1551-1554). ACM.

Ybanez, A. 2014. Impala for Android recognizes your pics and automatically sorts them into categories. Android Authority. http://www.androidauthority.com/impala-android-366554/

Zsombori, V., Frantzis, M., Guimaraes, R. L., Ursu, M.F., Cesar, C., Kegel, I., Craigie R. and Bulterman, D.C.A. 2011. Automatic generation of video narratives from shared UGC. In Proceedings of the 22nd ACM conference on Hypertext and hypermedia. ACM.

Monox: Extensible Gesture Notation for Mobile Devices

Roman Ganhör
Vienna University of Technology
Favoritenstrasse 9-11/187
1040 Vienna, Austria
roman.ganhoer@tuwien.ac.at

Wolfgang Spreicer
Vienna University of Technology
Argentinierstrasse 8/187
1040 Vienna, Austria
wolfgang.spreicer@tuwien.ac.at

ABSTRACT
The rise of modern smartphones brought gesture-based interaction to our daily lives. As the number of different operating systems and graphical user interfaces increases, designers and researchers can benefit from a common notation for mobile interaction design. In this paper, we present a concept of an extensible sketching notation for mobile gestures. The proposed notation, *Monox*, provides a common basis for collaborative design and analysis of mobile interactions. *Monox* is platform independent and enables general discussions and negotiations on topics of mobile gestures. An extensive evaluation showed the practicability and ability of *Monox* to serve as a common denominator for discussion and communication within interdisciplinary groups of researchers, designers and developers.

Author Keywords
Design; Interaction Design; Interface Design; Mobile Interaction; Gesture; Expression; Interaction Pattern; Fitts' Law; GOMS;Software Engineering; KLM.

ACM Classification Keywords
H.5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

General Terms
Human Factors; Design; Measurement.

INTRODUCTION
Sketching is a useful tool when discussing novel interaction mechanisms and application designs [2]. Regardless of whether researchers scrutinizing new approaches for mobile interactions or a team of developers exploring different variations of an interface design, they all utilize sketches to communicate their ideas. However, since there is no commonly accepted general notation for mobile gestures researchers and designers use their very own notation or use one of the notations provided by one of various mobile operating systems.

MobileHCI '14, September 23 - 26 2014, Toronto, ON, Canada

ACM 978-1-4503-3004-6/14/09…$15.00.
http://dx.doi.org/10.1145/2628363.2628394

The mobile operating systems with the biggest market shares are Google Android, Apple iOS and Microsoft Windows Phone [11]. In addition to these widely known operating systems, there exist some less popular ones such as Asha and MeeGo (Nokia), Blackberry or Bada (Samsung). According to recent announcements new mobile operating systems such as Ubuntu Phone (Canonical), Firefox OS (Mozilla) and Tizen (Samsung, Intel) have already entered the market or will enter the market soon [5].

Despite the variations in the underlying platforms and the targeted markets, all of these operating systems utilize touch-based gestures as their main interaction concept. Basic interaction gestures such as *tap* or *swipe* are part of all mobile operating systems. However, there are differences and thus, every implementation is unique in its own sense. For example, the number of fixed hardware and software buttons ranges from zero (Meego on Nokia N9) to four (Android on HTC Sensation).

The number of hardware buttons is a basic design decision that influences the overall interaction concept. Devices with dedicated buttons can rely on them and thus, can provide a consistent user experience for every application (home or search). On the other hand, devices with no buttons have to provide different strategies to compensate for the lack of dedicated buttons. As a result, every platform promotes its own style guide to ensure a consistent user experience for all their applications. Consequently, there exist various notations and expressions for the same gesture. This is not only potentially confusing when discussing sketches, it also complicates the comparability and interchangeability of sketches and design ideas. Moreover, evaluating user experience is unnecessarily cumbersome due to the lack of a standardized set of gesture expressions.

In this paper, we propose an extensible notation for gestures (*Monox*, MObile NOtation - eXtensible) on touch-based devices. *Monox* can support comparability over different platforms and provides extensibility for the pre-defined basic set of gesture expressions. *Monox* comprises of a superset of gestures for mobile devices and serves as a common ground for researchers, designers, programmers and executives. Through its handwriting-like notation it is suitable for collaborative sketching in early phases of design sessions. Furthermore, due to its platform-

independency, *Monox* can be a first step in building a cross-platform analysis tool. We envision a standardized set of gesture expressions that supports discussions in interdisciplinary groups consisting of individuals with various professional backgrounds.

BACKGROUND & RELATED WORK

Every company or organization on the market distributing mobile operating systems provides guidelines on how applications should be written and how specific interactions should be referred to [1,3,7]. Besides, various independent designers made exhaustive compilations of existing gestures [17,24]. However, simpler and more dynamic collaborative evaluation tools could assist in making decisions and in finding a common ground for discussions. Existing tools focus on software supporting design [12] or distributed collaboration tools [16]. In the following, we provide a short overview of existing approaches for describing and evaluating touch based mobile interaction.

Mobile Extensions to GOMS/KLM

Various methods for measuring and analyzing usability and design issues on desktop interfaces have been introduced, such as Fitts' Law, GOMS and KLM [4]. However, exploring novel methods for quantifying user experience is still an active research area [14], especially for mobile interfaces. Using regular expressions is such an approach to describe user interaction mechanisms [12]. Other approaches are KLM-qt [22] or the framework suggested by Heo et al [8].

Holleis et.al. [9] added "gestures" to the standard KLM making the method more applicable for modern mobile devices. Moreover, the authors specified existing parameters in more detail, e.g. splitting "attention shift" in two categories: a minor attention shift happens within the mobile device, whereas a major attention shift happens between the mobile device and the "real" world. The evaluation of the proposed mobile extension brought up promising results. However, the extensions in their publication focus on feature phones and not on smartphones. Thus, all possible gestures for touch interaction are subsumed into a single category "gesture", which is not appropriate for modern smartphones.

Jung et.al. [10] extended KLM for smartphone interaction by introducing more detailed gesture elements. For their experiment, the authors compared a novel *flick and press* gesture that outperformed the standard gestures for selecting a web-link on mobile devices. To underline the superiority of their design the authors used an adapted GOMS model for mobile phones.

To better apply GOMS for the mobile domain an extension was presented by Lee and Paik [15]. The authors introduced the term of *mental preparation time*, which depends on the experience of the user and the difficulty of the task. For example, it is much more challenging for an inexperienced user to deal with applications that regularly swap the whole content of the screen than it is for an experienced user who is familiar with the workflow of a given application. The experience of users influence their mental load and thus, *mental preparations time* can vary significantly among users. However, complex interactions are hardly covered by the proposed extensions.

Gesture Notations

While GOMS and KLM are useful for measuring interactions these methods are of limited use during the design phase where basic ideas are discussed. Sketches are widely used in early stages of design [2]. Thus, designing and discussing applications for touch-based devices involves both, sketching the elements of the interface and sketching the user interaction (gestures).

Notations for gestures are provided by most mobile platforms (e.g. Android, Ubuntu). At least written guidelines are supplied (e.g. Apple). Figure 1 depicts the notation styles of Windows Phone and Ubuntu Phone.

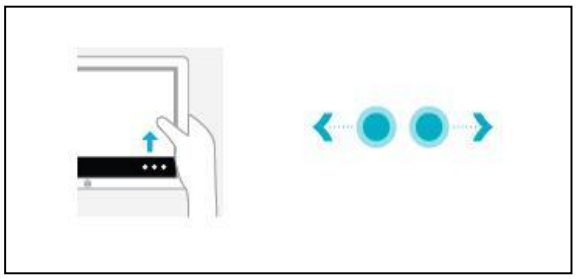


Figure 1: Gesture notation of Windows Phone (left) and Ubuntu Phone (right)

Comparing these guidelines and notation styles reveals various issues on interoperability, accuracy and extensibility.

Interoperability: since the guidelines differ in look and appearance designers have to agree on one notation and its implicit meaning. This implicates a limited interoperability between groups due to limited definitions of existing notations.

Accuracy: existing guidelines illustrate just the gesture and leave out additional information. For example, in certain scenarios it can be important to define the start area and the stop area of a *"swipe"* or *"fling"*. Communicating this extra information with a gesture expression can be crucial for the understanding of its usefulness.

Extensibility: none of the guidelines offer explicit rules to extend the given notation. Especially when exploring novel gestures it can be tedious to discuss the meaning and boundaries of newly generated expressions.

To overcome the limitation of interoperability designers already published vendor-independent gesture collections

[24,6]. However, the artwork of these collections is thoroughly advanced and not primarily designed for quick sketches. Furthermore, these compilations do not provide particular accuracy (exact start of and end of a gesture) nor do they provide extensibility.

Models and Diagrams

In the domain of software engineering applications are often described with diagrams such as UML. Diagrams provide graphical representation of tasks and subtasks and express their relationship amongst each other [18]. Thus, diagrams can help to set a common ground for communicating and discussing ideas. Ambiguity and other problematic cases can be identified in early process stages.

Programming software for gesture recognition raises the need for appropriate gesture sketches. During the design phase sufficient precise sketches help to identify and pinpoint the gestures and afterwards the same sketches can be used in a manual to describe the operating principles of the software [13,19,21,23]. Figure 2 depicts various styles used to sketch and notate gestures. The drawings range from formal (left) to descriptive (center) and artistic (right).

Figure 2: Various notations styles for gestures

A useful gesture notation would both serve as a base for measuring user experience (GOMS, KLM and other) and provide a set of extensible gesture expressions for designers (sketches for discussion). In the next section we propose *Monox*, out approach to such a gesture notation.

MONOX - GESTURE GRAPHICS

Monox addresses the need for distinct and extensible expressions for mobile gestures. One aim is to provide graphical representations for the gestures offered by various mobile platforms. In the following, the expressions are described in detail. Since the notation should be easy to draw by hand - especially by non-artists - all sketches throughout this paper are kept simple in their appearance.

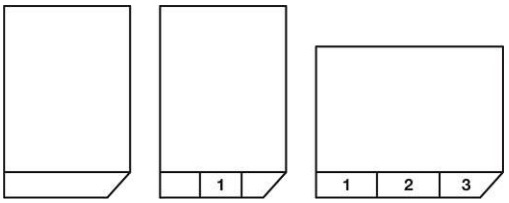


Figure 3: Phone outline with platform-dependent buttons

The primary layout of a *Monox* sketch is a simple rectangle representing the front side of the mobile device. The skewed corner indicates the lower right corner of the device when holding it in its intended position, portrait for mobile phones, landscape for tablets. Whenever the device rotates the skewed corner rotates in the same direction. The lower part of the phone is separated from the screen and contains the platform-dependent buttons (software buttons or hardware buttons). The buttons are numbered from left to right in an increasing order. If a phone does not provide any buttons, this area stays empty. Figure 3 depicts three devices: a smartphone with no buttons (left), a smartphone with one button (middle), and a tablet with three buttons (right). The skewed corner indicates the device's orientation.

Tapping Gesture

A common gesture on touch-based devices is pointing and clicking a specific icon (tapping). This gesture is covered by most guidelines and gesture collections. However, optional definition of a launch area is missing. The launch area defines the boundaries within the tap must be performed. The bigger the launch area is, the easier it is to hit it (Fitts ´Law [4]). In *Monox*, the launch area is defined by a rectangle around the pointing icon. If a user misses the launch area, the system can ignore the pointing gesture or accept it as a (unintentional) pointing to a different item. The earlier is without consequences for the user the latter leads to an unintended input.

Figure 4: Pointing gesture with launching area

Figure 4 depicts: (a) a normal tap; (b) a double tap; (c) a long tap. Combining a short and a long tap is shown in (d) and (e): (d) starts with a short tap and ends with a long tap; (e) starts with a long tap and ends with a short tap. (f) and (g) add additional information to the basic tap gesture. A single outline (f) indicates the area wherein the tap can be carried out. This single outline can be narrow like in (f) or as big as the whole screen. For the sake of clarity every tap should be encircled. A single outline as shown in (f)

indicates, that a tap outside the outline has no further consequences. In contrast, the double outline of (g) indicates that a tap outside the outline has consequences like closing the application or entering unintended input. (h) hints to consequences left and right of the outline and no consequences above and below the outline.

Extending the presented basic pointing gestures can be achieved by using the basic rules presented above. A thin round outline represents a short tap, a thick round outline represents a long tap. Sketching short and long taps around each other combines them to tap sequences (like (b), (d) or (e)). To perform a tap sequence the user starts with the outermost circle and end with the innermost circle. If an interaction is intended for two fingers two tap sketches are drawn next to each other. The same applies for three or four fingers.

Moving Gesture

While tapping gestures have just one coordinate, *swipes* and *moves* have two: the start coordinate and the end coordinate of the gesture. The time needed to carry out a gesture is another important variable. While a *swipe* is a more swinging movement of a person's hand (little time is needed) a *move* can be slower and thus, more time consuming. However, both gestures start within a starting area and end within a landing area. Another similarity is the presence of a directional movement.

In *Monox,* a (slow) moving gesture is indicated with an arrowhead (Figure 5a), whereas a (fast) swiping gesture is drafted with a double arrowhead (Figure 5b). Moving or swiping in either way is depicted with arrowheads on both ends (Figure 5c). Whenever a *move* or *swipe* should carried out with more than one finger, the number of fingers is hinted by the number of dashes crossing the arrow (Figure 5d).

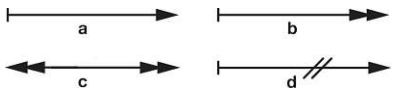


Figure 5: Move, swipe, drag and drop

The meaning of *move* and *swipe* gestures can depend on their starting and landing areas. These areas are notated as follows: a starting area is drawn as a rectangle with sharp corners, whereas the landing area has rounded corners (Figure 6a). If both areas overlap, just the sharp corners are drawn (Figure 6b). An arrow starts and ends within the boundaries of a starting or landing area, usually touching the boarder.

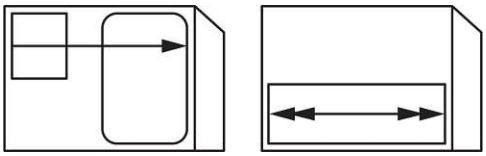


Figure 6: Move, swipe

The *move* and *swipe* gesture can be extended with the tapping gesture to create known or new interaction mechanisms. Figure 7 shows such a combination to form a drag and drop gesture.

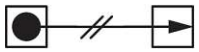


Figure 7: Drag and drop, move with 2 fingers

Multi-Touch Gestures

Multi-touch gestures are carried out with two or more fingers and include gesture like spreading (Figure 8a), pinching (Figure 8b) and rotating (Figure 8c and 8d).

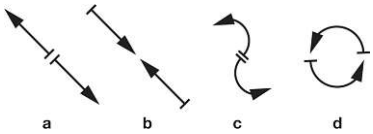


Figure 8: Multi touch gestures

Again, multi-touch gestures can be combined with existing gestures such as the tapping or starting and landing areas.

Screen Change

Due to the small screen sizes of mobile devices the entire screen is usually utilized by a single application. Typically, every interaction leads to a new screen and thus, to a (minimal) cognitive load for the user who has to remember the content of the previous screen. This cognitive load depends on the familiarity with the application and on the importance of the (now not visible) information. However, depending on the application and its implementation not every interaction changes the whole content of a screen. Often only a partial area is changed. Since the way the screen is updated can influence the 'flow' of the interaction, it seems practical to distinguish between *full* and *partial* screen update.

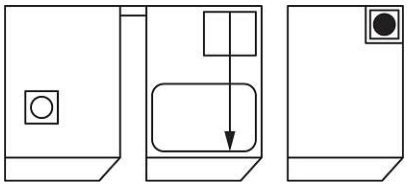


Figure 9: New screen and updated screen

The connection on the upper edge between Figure 9a and Figure 9b indicates a partial screen update when carrying out a short tap (Figure 9a). After the *move* in Figure 9b the screen performs a complete screen update since there is no connection between (b) and (c) on the upper edge between Figure 9b and Figure 9c.

Rare Gestures

Rare gestures are uncommon or novel gestures. Such gestures can be illustrated by combining the appropriate expressions presented above. An example for an uncommon gesture is a Bezel Swipe [20] or Edge Swipe used by MeeGo and Ubuntu Phone (Figure 10).

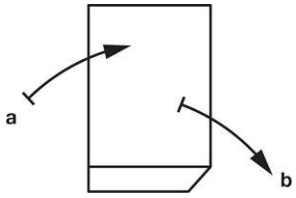


Figure 10: Bezel Swipe as an uncommon or novel gesture

Especially the field of mobile gaming creates novel an unique gestures. In *Monox*, new gestures can be introduced by combining the given set of expressions.

EVALUATION

We conducted an iterative evaluation process to evaluate our approach and identify missing expressions. In the first iteration (initial study) we discussed *Monox* with HCI experts to set the general direction for the notation. In the second iteration (usability study) we conducted an evaluation with a large number of students to proof the basic concept of *Monox*. The findings of these two iterations completed the set of gesture expressions of *Monox*. This extended notation formed the basis for the third iteration (field study), which evaluated *Monox* in real world settings.

First Iteration (Initial Study)

A first iteration was carried out with four HCI experts in a workshop setting to test the basic idea of the proposed gesture notation. The participants were asked to carry out four different tasks. Each participant used a different platform on a daily basis: Apple iOS, Google Android, Microsoft Windows Phone and Nokia MeeGo. The tasks involved two low-level tasks (turning on wireless network and switching between two applications) and two high-level tasks (writing an e-mail and bookmarking a website). In conducting the workshop we were able to do both evaluate the applicability of *Monox* and compare existing interaction designs over various platforms.

The gestures made to fulfill the given tasks were observed by the authors and notated using *Monox* expressions. Afterwards, the tasks were discussed with the participants. During the discussion, we got a detailed picture of the different approaches made by each smartphone manufacturer. Some participants were able to derive basic guidelines for an operating system (unknown to them) just by observing the collected gesture expressions. Furthermore, potentially weak or error-prone interaction techniques were spotted during discussing the collected gesture expressions.

However, this initial evaluation showed the possible potential of the proposed notation. Through *Monox*, HCI experts were able to compare different interactions on various applications spanning over different platforms. Furthermore, the participants were able to discuss improvements for platforms they were not aware of through a unified gesture set.

Second Iteration (Usability Study)

Although the insights of the workshop with HCI experts helped to resolve minor issues with the expressions, we wanted to evaluate our approach with a large user base. Furthermore, we intended to find missing expressions and issues with existing expressions. Thus, we instructed 120 bachelor students in a HCI course (M=23,7yr, SD=3,5) to sketch one of the following tasks: 1) take a photo and change its brightness 2) plan a route to a given destination 3) follow a given user on Twitter or 4) describe a mobile game you recently played. The prerequisites were carrying out the task on a mobile touch-based device and using *Monox* notation. The students were asked to give hints about missing expressions and their overall experience when using *Monox*.

Categorization

The feedback from the students was analyzed and grouped in categories. Based on these categories we identified missing expressions and extended the existing expressions. In the following we present the most often mentioned categories: text input, sensor input, hardware buttons and gesture macros.

Text Input: this category addresses the possibility to type alphanumeric characters by means of a software-emulated keyboard. Participants criticized the exaggerated effort to illustrate arbitrary textual input in *Monox*. An often-proposed approach by the students was a single expression that covers general alphanumeric input, such as a stylized keyboard.

Sensor Input: modern mobile devices are equipped with numerous sensors (accelerator, gyroscope, GPS, NFC, etc.). This category summarizes the possibilities to express sensor input using *Monox*. Participants suggested expressions for shaking, rotating, contactless cross-device communications and audio-based input.

Hardware Buttons: besides touch-based input, mobile devices possess a number of hardware-based buttons to interact with the user, such as volume and power buttons. As these buttons can be an integral part of an interaction sequence visualizing user interaction with hardware buttons should be considered.

Gesture Macros: various interaction sequences contain recurring steps. Often mentioned repetitious tasks were power on and unlock the phone as well as repeating touch interactions in complex applications such as mobile games. To condense such recurring interaction steps in a single *Monox* expression we gathered suggestions for gesture macros made by the participants.

These four categories form the basis for a revision of *Monox*. In the next section we present and discuss the additions we made to complete the proposed notation.

Results

The overall purpose of *Monox* is to foster a systematic approach towards a useable illustration of mobile user interaction. Therefore a basic set of expressions was reduced to a necessary minimum in order to support learnability as well as usability. The idea and notation were well received by the participants, however, the findings of the iterative evaluation with HCI experts and 120 students showed some missing basic, and hence important expressions. These missing expressions belong to four categories we defined above: text input, sensor input, hardware buttons, gesture macros.

Text Input: To achieve a simple and useful illustration for text input the context of the user interaction should be considered. Text input can be letters only, numbers only, mixed, lower case, upper case or special characters. Therefore, we propose following solutions for *Monox* (Figure 11).

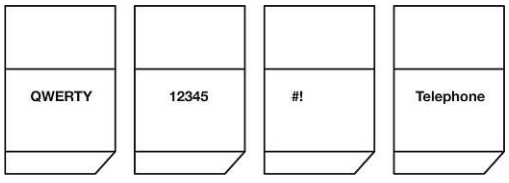


Figure 11: Text input

A software-emulated keyboard usually allocates approximately the lower half of a mobile screen. *Monox* reflects this with a simple schematic depiction of such a keyboard. Additionally, the preset of the keyboard layout is written in the stylized keyboard: "QWERTY" for alphanumerical input, "12345" for numerical input, "#!" for special characters and "Telephone" for a phone like keyboard. Considering the keyboard layout during the sketching phase in a design process can support the awareness of the proper keyboard layout and thus, can lead to more consistent interfaces.

Sensor Input: Sensor data is part of a growing number of mobile applications and should be reflected in *Monox*. Sensor input can be generated actively by moving or shaking the device (accelerometer, gyroscope, NFC) or can be collected passively without direct user involvement (GPS, iBeacon). Figure 12 depicts the notation in *Monox*.

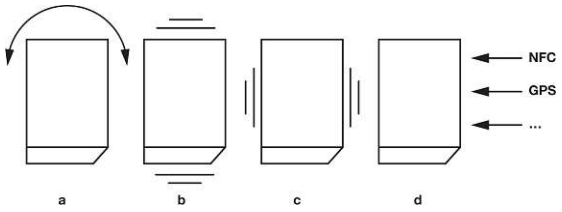


Figure 12: Sensor input

Hardware Button: In addition to input via touch screen mobile devices often use hardware buttons to perform certain functions, such as setting volume or adjusting the camera zoom. Such interaction is notated using the tap expressions (long tap, short tap, etc.) and located outside the device (Figure 13). The expression of hardware buttons in *Monox* follows the principle of simplicity and the idea of re-using given expressions. In this way, the hardware buttons follow the interaction logic already described and do not introduce an additional type of expression.

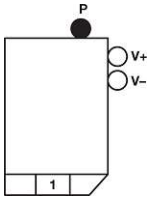


Figure 13: Hardware buttons; long tap on power (P) and short taps on volume up/down (V+/V-)

Gesture Macros: Macros represent a sequence of interactions as a single expression, making the notation task less tedious and error-prone. Using gesture macros follows a two-step process. First, defining a macro by putting the desired interaction sequence in brackets and labeling the sequence with a unique name. Second, using the macro by

adding an empty device outline with the desired label on top of it (Figure 14).

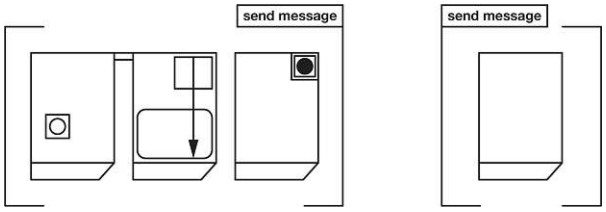


Figure 14: Gesture macros

Together with our initial suggestions for a mobile gesture notation, these additional expressions form an extended version of *Monox*. An evaluation of this new version was realized by means of a field study in real world setting.

Third Iteration (Field Study)

For the field study we visited three different project teams. Every team was in a different stage of the development cycle (setting up the project, defining the requirements, redesigning an existing implementation). To evaluate *Monox* in real world settings, we asked each team to use *Monox* in one of their design sessions.

mApp - discussing ideas for a mobile application

mApp is a small company that focuses on mobile applications. We conducted a design session and an interview with one employee who acts as a technical project manager. The design session and the interview were set up in the office of the company. In the design session the participant (employee) was asked to draw a few sketches (using his own expressions) and explain the idea of the upcoming project. This was done to reveal any similarities in his expressions with *Monox*. It turned out that the company followed a completely different strategy to communicate and discuss interaction design ideas.

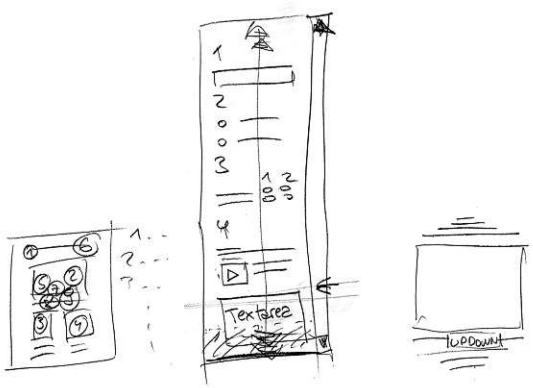


Figure 15: Sketches made by the project manager

Figure 15 on the left depicts a typical sketch used in this company. The sketch is numbered all the way through and on an additional sheet of paper the meaning of the numbers is explained in detail with additional sketches. The participant agreed on the argument of a messy sketch and the potential usefulness of a standardized notation. *"However, I doubt this (standardized expressions) will help us much. It is not about the notation itself. It is just the people are ignorant about anything new"*, he explained. Furthermore, the participant revealed an use-case which is not covered by *Monox* now (Figure 15, middle). *"Mobile apps are more and more going to be a single page. You can jump here and there within this page. I guess, in a few years time page flipping will be dead anyhow"*.

After the design session an interview about *Monox* was conducted. Asking the participant where he thinks *Monox* could fit best in their existing workflow he answered "... *wireframe tools like balsamiq.com or similar*". He drew the shaking-sketch from *Monox*, which he found particularly useful (depicted in Figure 15, right), and explained his thought: "... *every tool comes up with its own notation. This is weird and makes group work impossible. If you add a small hint to the sketches this would help a lot. In a context like wireframing tools this make complete sense for me*".

Prool - designing a mobile prototyping tool

Prool is intended to be a mobile application to ease process how mobile prototypes are built and evaluated. In a first design session the process of initializing the application was found too complicated. The goal of this design session was to simplify the initialization process. Two participants with interface- and interaction design background took part in the design session. During the previous development of Prool both participants used a self-made notation that was continuously adapted to the current situations and problems. Thus, depending on situation and problem an expression could have diverse meanings. The participants did not considered this as a major problem although it sometimes led to lengthy and unnecessary discussions.

However, the participants agreed on the presumption that their approach could be problematic on larger projects with more people involved. They further agreed on the idea that a standardized notation could be useful in general. To gather information about the strengths and weaknesses of *Monox* in the described setting we chose the following setup: one participant (Person M=*Monox*) was introduced to the notation of *Monox* and explained its expected advantages like unambiguousness and reproducibility. Person M was also encouraged to use *Monox* throughout and explain the expressions while drawing them. The second participant (Person Z) was just informed about *Monox*, but not instructed. The idea was to see how these two notations (*Monox* vs. non-standardized ad-hoc) "compete" against each other and what kind of discussion starts between the participants about the new notation. Therefore both participants were asked to think out loud.

The design session was set up at the participants' workplace, where the sketches were drawn on whiteboards. Person M (aware of the *Monox* notation) used *Monox* consequently from the beginning whereas Person Z used the informal notation he was used to. Both participants explained the meaning of their sketches during drawing. After a short period of time Person Z began to adopt touch-expressions from *Monox*. Person Z commented this with: "*... these touch expressions are somehow well defined and easy to remember. So I don´t have to explain every small change in the interaction design*". In contrast, Person M fell back to the old behavior every now and then and had to redraw sketches. This could be observed especially with small/little interactions. In this case Person M wanted to avoid drawing a complete new sketch and added the interaction to the previous sketch (Figure 16 right).

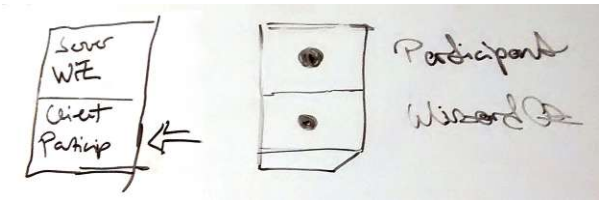


Figure 16: Participants's informal notation (left), Monox (right)

Figure 16 depicts a sketch in both notations used during this session. On the left hand side the informal notation, whereas the right hand side shows the same interaction with *Monox* (with all interaction possibilities on the screen at one time). This contrasting juxtaposition shows the different approaches of these two notations quite well. The informal notation concentrates on the content on the screen and not the interaction. The interaction is explained verbally while sketching and is marked with arbitrary markers (in this case an arrow). *Monox* in contrast prioritizes the interaction. This difference was made the subject of a discussion during the session and the following statements are pointing in this direction:

"*I agree on the idea of Monox, but wouldn't it be nice to have both world (interface design and interaction design) in one sketch?* " (Person Z)

"*Sometimes it is hard to draw a full sketch from scratch just to add a minor interaction. It sometimes feels more natural to add a second interaction to an existing sketch*" (Person M).

After the design session both participants were familiar with the idea of *Monox* and a short interview was conducted. Hereafter a few statements made during the interview.

"*I see the advantage when working over distance. Monox is much clearer and above all, specified. This leaves out a lot of confusion and misunderstandings. Definitely can make things easier.* " (Person Z)

"*You have to learn it, but I guess, when you know it, you just use it like you use hand writing*" (Person M)

"*Does extensible mean, everybody can extend it their way they want?* " (Person Z)

The interview revealed the need for standardization and the participants back ambitions for standardization. At the same time the participants emphasized the (assumed) advantages of case-dependent notations. However, extensible notations like Monox could be both, standardized and open for case-dependent ad-hoc notations as well.

Private Banking - redesigning a Tablet Application

We were able to take part in a usability workshop for a tablet application in the private banking sector. The application in question runs on Windows 8 and is implemented with C# .NET. The workshop participants consisted of a software engineer, who was part of the development team, a graphic designer, an executive of the IT company running this project and one of the authors for moderation and assisting the participants in the usage of *Monox*. The workshop was organized to evaluate the user experience of the application. The participants agreed on redesigning "*investment consultancy*" as the primary use case for this session.

After an introduction to *Monox* by the moderator, the workshop participants started a heuristic evaluation of the consultancy workflow. Whenever a flaw was detected the team utilized *Monox* to discus the problem and potential solutions. In the following we provide an example of an interaction design issue that was discussed by the workshop participants.

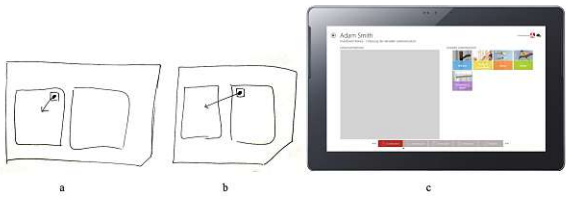


Figure 17: Drag and drop interface elements (courtesy of CPB Software Austria GmbH)

The example shows a problem with an inconsistency following a drag and drop interaction. The interaction is meant to add interface elements to the grey area on the left side of the screen (Figure 17c, right). Therefore the user has to choose one of the interface elements provided on the right side of the screen and move it to the grey area via drag and drop. The *Monox* sketch for this interaction is shown in Figure 17b (center). After this interaction a pop-up window appears providing additional parameters for the element.

Figure 17a (left) shows the interaction design problem found by the programmer: whenever the user applies a drag and drop option to one of the elements already added to the grey area on the left the application also instantiates a new element and calls the corresponding pop-up, which should not be the case. The programmer made a sketch of the problem using *Monox*. He said in the direction of the moderator: "*Because of you, I have to renounce my own notation*". Although it would have been easier for the programmer to use his own notation, the graphic designer and the executive were not aware of this notation. This shows, that a common notation like *Monox* supports cooperative discussions about interaction design, especially when working in interdisciplinary teams. By sketching the problem on paper the programmer could do both, explaining the problem to the other participants and taking the paper sketch with him as a reminder and guide how to fix the problem.

DISCUSSION AND CONCLUSION

In this paper we presented a flexible, combinable and extensible notation (*Monox* - MObile NOtations eXtensible) for touch-based gestures on mobile devices. The proposed expressions are intended to be simple to sketch and easy to extend. *Monox* allows the illustration of interaction tasks and the comparison of interaction patterns regardless of the platform. Thus, *Monox* can help to discuss and pinpoint strenghts and weaknesses in an interaction design for mobile applications. Our findings show that the proposed notation can provide a common ground for discussing and exchanging ideas during all phases in a project (design, implementation, evaluation).

Monox was developed in an iterative process and reflects multiple aspects of touch-based interaction on modern mobile devices. The first two iterations included an initial workshop with HCI-experts and a usability evaluation with 120 participants. Based on the results of the evaluation we designed a revised version of *Monox* with additional expressions. The third iteration was a field study utilizing *Monox* in three different real world settings. The field study included various stakeholders like researcher, project manager, programmer and designer.

During the first field study *Monox* was proposed as a tool for detached groups supporting collaborative working. The participant explained the different settings the groups had to work in. These groups include persons who are not willing to learn something new, like *Monox*. However, the participant also suggested that a unified and well-defined notation could form the basis for XML- or JSON-based gesture descriptions. And these technical descriptions could be used as an interchange format for wireframing tools.

The second field study was conducted with two participants who were designing a mobile application for rapid prototyping. During the observed design session various interaction sequences were discussed by the participants utilizing both, *Monox* and their own non-standardized expression set. Depending on the particular interaction sequence they decided in-situ which notation would fit best. Both participants agreed on the idea of a unified and expandable notation for mobile applications. The participants saw the main advantage of *Monox* in its unambiguity that eases discussions about complex interaction sequences. However, especially one participant urged for joining the content and interaction expression in one sketch.

In the third field study we applied *Monox* in a redesign workshop for a tablet application in the private banking sector. The results showed, that the usage of *Monox* provides a common ground for discussing usability issues in an interdisciplinary team. This allowed the team members to focus on redesigning user interaction rather than struggling with different self-defined notations. However, we noted that the participants of the workshop focused on a basic set of expressions ignoring the more advanced expressions. The participants mentioned that they would enjoy a tool for tablets or mobile phones, which provides the possibility to add *Monox* sketches as an overlay to the current screen.

In each iteration of our design process we gathered useful feedback from the participants. Thus, we propose future work should further explore how to merge content with expressions, avoiding ambiguity of screen elements and interaction expressions at the same time. Furthermore, the integration of *Monox* into existing tools and applications could beneficial for rapid prototyping. Both could increase the usefulness and acceptance of the notation.

Extensive work has been done to qualify and quantify touch-based gestures, mostly utilizing proprietary vendor-specific notations.

An open alternative could help to compare designs across platforms and to gain more valuable insights. Therefore we see *Monox* as a first step in establishing a common, unified notation easing the burden of finding best practices in interaction design across various platforms and applications. Serving as a tool for researchers, designers and programmers, *Monox* can also lower the gap between distinct professions whenever discussing or evaluating novel ideas and designs. We hope, that the ideas presented in this paper will motivate others to contribute to the process of establishing a comprehensive notation for touch-based gestures.

ACKNOWLEDGMENTS

We thank all the participants involved in the studies and everyone who helped to shape this paper.

REFERENCES

1. Apple Inc. 2013. iOS Human Interface Guidelines. http://developer.apple.com/library/ios/#documentation/userexperience/conceptual/mobilehig/Characteristics/Characteristi cs.html

2. Buxton, B. 2010. Sketching User Experiences: Getting the Design Right and the Right Design: Getting the Design Right and the Right Design. Morgan Kaufmann, 2010.
3. Canonical Ltd. 2013. Ubuntu Gestures http://design.ubuntu.com/apps/get-started/gestures
4. Fitts, P.M. 1954. The information capacity of the human motor system in controlling the amplitude of movement. Journal of Experimental Psychology. 47(6): p. 381-391.
5. Gandhe, S. 2014. Press render of Samsung's Tizen-based ZEQ 9000 leaks online. Neowin LLC. http://www.neowin.net/news/press-render-of-samsungs-tizen-based-zeq-9000-leaks-online
6. Gesture Works. 2013. http://gestureworks.com/pages/core-features-gestures
7. Google, Inc. 2013. Android Patterns and Gestures http://developer.android.com/design/patterns/gestures.html
8. Heo, J., Ham, D-H., Park, S., Song, C. and Yoon. W.C. 2009. A framework for evaluating the usability of mobile phones based on multi-level, hierarchical model of usability factors. Interaction Computing 21, 4 (August 2009), 263-275.
9. Holleis, P., Otto, F., Hussmann, H. and Schmidt, A. 2007. Keystroke-level model for advanced mobile phone interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '07). ACM, New York, NY, USA, 1505-1514.
10. Jung, K. and Jang, J. 2013. A Two-Step Click Interaction for Mobile Internet on Smartphone. In *Communications in Computer and Information Science - Posters' Extended Abstracts*. Springer Berlin Heidelberg 2013.
11. Kantar Worldpanel Comtech Report 2013 http://www.kantarworldpanel.com/dwl.php?sn=news_downl oads&id=151 03.08.2013
12. Kin, K., Hartmann, B., DeRose, T. and Agrawala, M. 2012. Proton: multitouch gestures as regular expressions. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems* (CHI '12). ACM, New York, NY, USA, 2885-2894.
13. Kin, K., Hartmann, B., DeRose, T. and Agrawala, M. 2012. Proton++: a customizable declarative multitouch framework. In *Proceedings of the 25th annual ACM symposium on User interface software and technology* (UIST '12). ACM, New York, NY, USA, 477-486. DOI=10.1145/2380116.2380176 http://doi.acm.org/10.1145/2380116.2380176
14. Law, E.L-C. 2011. The measurability and predictability of user experience. In *Proceedings of the 3rd ACM SIGCHI symposium on Engineering interactive computing systems* (EICS '11). ACM, New York, NY, USA, 1-10.
15. Lee, J. and Paik, D. 2007. Human-Computer Interaction. Interaction Platforms and Techniques. In *Human-Computer Interaction, Part II, HCII 2007*, Springer-Verlag Berling Heidelberg. LNCS 4551, p. 401-407.
16. Li, G., Cao, X., Paolantonio, S. and Tian, F. 2012. SketchComm: a tool to support rich and flexible asynchronous communication of early design ideas. In *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work* (CSCW '12). ACM, New York, NY , USA, 359-368.
17. McMahon, K. 2013. App Dev Wiki. http://appdevwiki.com/wiki/show/HomePage
18. Paternò, F., Mancini, C. and Meniconi, S. 1997. ConcurTaskTrees: A Diagrammatic Notation for Specifying Task Models. In *Proceedings of the IFIP TC13 International Conference on Human-Computer Interaction* (INTERACT '97), Steve Howard, Judy Hammond, and Gitte Lindgaard (Eds.). Chapman & Hall, Ltd., London, UK, UK, 362-369.
19. Ramanahally, P., Gilbert, S., Niedzielski, T., Velázquez, D. and Anagnost C. 2009. Sparsh UI: A Multi-Touch Framework for Collaboration and Modular Gesture Recognition. In *Proceedings of World Conference on Innovative Virtual Reality*. ASME-AFM.
20. Roth. V. and Turner, T. 2009. Bezel swipe: conflict-free scrolling and multiple selection on mobile touch screen devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '09). ACM, New York, NY , USA, 1523-1526.
21. Schmidt, M. and Weber, G. 2013. Template based classification of multi-touch gestures. Journal of Pattern Recognition. Elsevier. DOI=http://dx.doi.org/10.1016/j.patcog.2013.02.001
22. Schulz, T. 2008. Using the Keystroke-Level Model to Evaluate Mobile Phones. Trolltech ASA. University of Oslo. http://hdl.handle.net/10852/9883
23. Wobbrock, J.O., Ringel Morris, M. and Wilson, A.D. 2009. User-defined gestures for surface computing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '09). ACM, New York, NY, USA, 1083-1092. DOI=10.1145/1518701.1518866 http://doi.acm.org/10.1145/1518701.1518866
24. Wroblewski, L. 2013. Touch Gesture Reference Guide http://www.lukew.com/ff/entry.asp