

Barrierefreiheit auf der Android-Plattform mit Schwerpunkt auf Untertitel bei der Wiedergabe von Medieninhalten

Universität Hamburg
Fachbereich Informatik
Technische Aspekte multimodaler Systeme

InfB-BA/SSE: Abschlussmodul B.Sc. Software-System-Entwicklung

Vorgelegt von:
Gökhan Kalender
Matr.-Nr: 6375522
Studiengang: SSE

Vorgelegt am:
20.08.2015

Erstgutachter:
Dr. Norman Hendrich
MIN-Fakultät
Fachbereich Informatik
Arbeitsgruppe: TAMS

Zweitgutachter:
Prof. Dr. Christopher Habel
MIN-Fakultät
Fachbereich Informatik
Arbeitsgruppe: WSV

Abbildungsverzeichnis

1	Entwicklung der Schwerhörigkeit in Deutschland von 2005 bis 2011, Quelle [1]	1
2	Marktanteile der Betriebssysteme an der Smartphone-Nutzung in Deutschland von Dezember 2011 bis Februar 2015, Quelle[2]	1
3	Ein Beispiel für Untertitel bei Bildern, Quelle[4]	4
4	Ausnahmeposition der Untertitel bei wichtigen Bildinformationen am Beispiel von "King of Queens", Quelle[22]	6
5	Aufsteigende inhaltliche Untertitel am Beispiel von "Star Wars". Quelle[23]	8
6	Statistik: TV-Sendungen in Deutschland mit Untertitel (Januar 2012)Quelle [9]	10
7	Statistik: TV-Sendungen in Deutschland mit Untertitel (Juni 2015), Quelle [10]	10
8	Tabellarische Darstellung der 'Control-Codes'	12
9	Effektive Größe eines Smartphones im Vergleich zum TV-Gerät	18
10	Übersicht des DemoPlayers	22

Inhaltsverzeichnis

1	Einleitung	1
2	Zielsetzung	2
3	Barrierefreiheit	3
4	Untertitel	4
4.1	Was sind Untertitel?	4
4.2	Beschränkungen	5
4.3	Konventionen	5
4.4	Arten	6
4.5	Kategorien	6
4.5.1	Erzählende Untertitel	7
4.5.2	Untertitel für Hörgeschädigte	7
4.5.3	Erzwungene Untertitel	7
4.5.4	Inhaltliche Untertitel	7
4.5.5	Lokalisierte Untertitel	8
4.5.6	3D-Untertitel	8
4.5.7	Untertitel als besonderes Element im Film	8
4.6	Untertitel im deutschen Fernsehen	9
4.7	Automatische Generierung	11
4.8	Formate	12
4.8.1	DVD-Video	12
4.8.2	MicroDVD	12
4.8.3	SubRip	13
4.8.4	WebVTT (Web Video Text Tracks)	14
5	Die Lesbarkeit von Untertiteln	16
5.1	Typografie	16
5.2	Schriftart und -größe	16
5.3	Einblendedauer	16
5.4	Kontraste	17
6	Umsetzung auf Android	18
6.1	Vergleich zum TV-Gerät	18
6.2	Bekannte Möglichkeiten	18
6.2.1	Der native MediaPlayer	18
6.2.2	ExoPlayer	20
6.3	SubtitleController	20
6.3.1	Funktionsweise	21
6.4	Demo	22

1 Einleitung

Im Jahr 2011 gab es laut Bundesamt der Schwerhörigen und Ertaubten ca. 14.9 Millionen Schwerhörige in Deutschland. Das hat zu dem Zeitpunkt ca. 20.9% der deutschen Gesamtbevölkerung über 14 Jahren ausgemacht. Die Zahlen der vorherigen Jahre zeigen, dass Anzahl der Schwerhörigen immer mehr zunimmt.

Anteil an der Gesamtbevölkerung über 14 Jahren

2005	2006	2007	2008	2009	2010	2011
19.7%	19.5%	19.7%	19.9%	20.2%	20.5%	20.9%
(14.0 Mio.)	(14.0 Mio.)	(14.1 Mio.)	(14.3 Mio.)	(14.5 Mio.)	(14.7 Mio.)	(14.9 Mio.)

Abbildung 1: Entwicklung der Schwerhörigkeit in Deutschland von 2005 bis 2011, Quelle [1]

Die Internet-Marktforschungsfirma Comscore hat eine Statistik veröffentlicht, die die Marktanteile der verschiedenen Betriebssysteme an der Smartphone-Nutzung in Deutschland von Dezember 2011 bis Februar 2015 darstellt. Aus dieser geht deutlich hervor, dass der Marktanteil des Android-Betriebssystems in diesem Zeitraum von deutlich unter 40% bis auf ca. 70% gestiegen ist. Den nächst kleineren Marktanteil hält iOS mit nur etwa 20%.

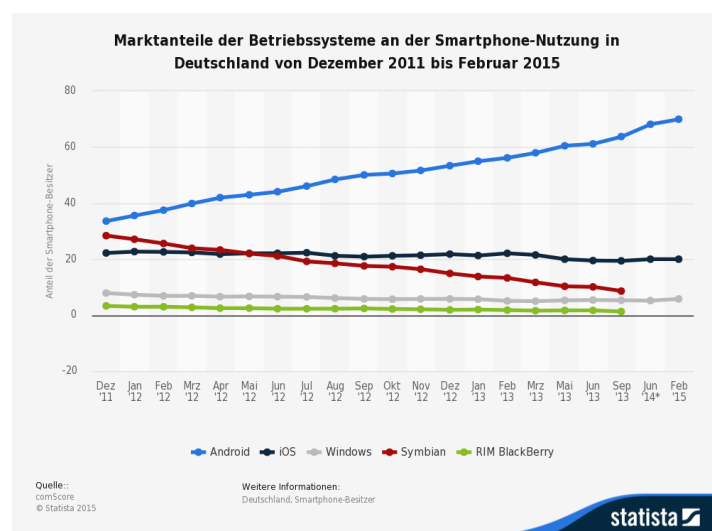


Abbildung 2: Marktanteile der Betriebssysteme an der Smartphone-Nutzung in Deutschland von Dezember 2011 bis Februar 2015, Quelle[2]

Aus diesen beiden Tatsachen ergibt sich eine hohe Wahrscheinlichkeit, dass einige dieser Schwerhörigen auch Android-Nutzer sind. Daraus ergibt sich die Relevanz diese Seminararbeit diesem Thema zu widmen.

2 Zielsetzung

Zum einen möchte ich mit dieser Ausarbeitung einen Überblick über Untertitel an sich geben, was es für verschiedene Arten und Kategorien gibt und mit welchen Formaten sie ausgespielt werden. Zum anderen möchte ich zeigen, dass es durchaus möglich ist auf Smartphones Untertitel so anzuzeigen, dass sie ihren Zweck erfüllen. Außerdem soll aus dieser Arbeit eine Komponente hervorgehen, die in der Lage ist Untertitel wiederzugeben, ohne dabei an einen bestimmten Player gebunden zu sein.

3 Barrierefreiheit

Zur Definition folgt ein Auszug aus dem “Gesetz zur Gleichstellung behinderter Menschen”:

§ 4 Barrierefreiheit

“Barrierefrei sind bauliche und sonstige Anlagen, Verkehrsmittel, technische Gebrauchsgegenstände, Systeme der Informationsverarbeitung, akustische und visuelle Informationsquellen und Kommunikationseinrichtungen sowie andere gestaltete Lebensbereiche, wenn sie für behinderte Menschen in der allgemein üblichen Weise, ohne besondere Erschwernis und grundsätzlich ohne fremde Hilfe zugänglich und nutzbar sind.” [3]

Im Kontext dieser Seminararbeit soll diese Definition vereinfacht aber erweitert werden. Die Barrierefreiheit soll die Zugänglichkeit für alle Personengruppen garantieren. Dabei geht es zum Beispiel um die Zugänglichkeit bei folgenden Aspekten:

Architektur

Rollstuhlfahrern ist es kaum möglich über Treppen in ein Gebäude zu gelangen. Rollstuhlrampen sorgen hier für die Barrierefreiheit. Älteren Menschen könnte das Treppensteigen schwerfallen. Hierbei helfen Aufzüge.

Kommunikation und Information

Die Gebärdensprache hilft Taubstummen bei der Kommunikation und Analphabeten können den Inhalt oder die Informationen eines Buches mittels Hörbücher aufnehmen.

Computer und Internet

Am Computer können hohe Kontraste die Lesbarkeit erhöhen. Anweisungen über die Benutzung einer Software können barrierefrei formuliert sein. Zum Beispiel sollten farbgebundene Anweisungen vermieden werden: “Drücken Sie den grünen Knopf zum Speichern, den roten zum Abbrechen.”

4 Untertitel

4.1 Was sind Untertitel?

Untertitel im Allgemeinen sind Textzeilen, die schriftliche Informationen zu einem Medieninhalt liefern. Diese Medieninhalte sind häufig Bilder oder Videos. Untertitel bei Bildern beschreiben meist das Abgebildete selbst, können aber auch Aufschluss über den Kontext eines Bildes geben. Bei Videos werden Untertitel meist begleitend zu gesprochener Sprache angezeigt. Während Untertitel bei Bildern meist außerhalb/unterhalb des Mediums dargestellt werden (vgl. Abb. 3), ist es bei Videos üblich, die Untertitel am unteren Bildrand zu platzieren. Bei Untertiteln zu gesprochener Sprache kann es eine Begleitung in der gleichen Sprache oder eine Übersetzung einer Fremdsprache sein. Für Hörgeschädigte gibt es spezielle Untertitel, die auch Aufschluss über wichtige akustische Ereignisse geben. So können Untertitel die Spannung und Dramatik eines Filmes erhalten, die sonst nur über die Akustik vermittelt werden. Das könnten zum Beispiel Explosionen, plötzliche Schreie vom Dachboden, plötzliche Stille oder andere Umgebungsgeräusche sein. Dabei soll der Leser allerdings nie mehr und nichts früher erfahren als der Hörer. Dazu ist es wichtig, dass Audio und Text synchron laufen.



Abbildung 3: Ein Beispiel für Untertitel bei Bildern, Quelle[4]

4.2 Beschränkungen

In Filmen ist es oft der Fall, dass die Untertitel nicht wortwörtlich das Gesprochene wiedergeben. Das liegt daran, dass bei der Untertitelung von Videos nur begrenzten Platz und begrenzte Zeit zur Verfügung steht und viele Sätze vereinfacht werden können. Der Nutzer muss genug Zeit haben, den Text zu lesen aber gleichzeitig auch einen Eindruck vom Bildmaterial zu bekommen. Es folgt ein Beispiel aus dem Film “The Matrix”, von 1999.

Gesprochen/Untertitel:

“Dies ist ein Sparring-Programm. Ähnlich wie die programmierte Realität der Matrix.”

Gesprochen:

“Es folgt den gleichen Gesetzen, denen der Schwerkraft zum Beispiel. Im Grunde unterscheiden sich diese Gesetze nicht von denen eines Computersystems.”

- 151 Zeichen. -

Untertitel: “Es hat dieselben Regeln. Wie Schwerkraft. Diese Regeln sind genau wie die eines Computers.”

- 91 Zeichen -

Das Gesprochene umfasst im Beispiel 151 Zeichen, die Leerzeichen sind dabei nicht mitgezählt. Der Untertitel hat nur 91 Zeichen und ist somit um ca. 40 % kürzer als das Gesprochene.

4.3 Konventionen

In den meisten Fällen werden Untertitel, aufgrund der guten Lesbarkeit, als weißer Text auf dunklem Hintergrund oder mit einer dunklen Umrandung dargestellt. Bei Videos befindet er sich meist zentriert am unteren Bildrand. Das ist allerdings nicht immer möglich. Es gibt Situationen, in denen im Videomaterial wichtige Informationen am unteren Bildrand vorhanden sind. In diesen Fällen wird der Untertitel ersatzweise am oberen Bildschirmrand oder zumindest erhöht angezeigt. Dies ist häufig der Fall, wenn zu Anfang eines Videos noch die Darsteller und Produzenten namentlich erwähnt werden aber schon ein Gespräch im Video stattfindet. Bei Serien passiert dies in der Regel häufiger als bei Filmen, da viele Serien mit einem Aufhänger in die Episode starten. Es folgt ein Beispiel aus der US-amerikanischen Sitcom “King of Queens”.



Abbildung 4: Ausnahmeposition der Untertitel bei wichtigen Bildinformationen am Beispiel von "King of Queens", Quelle[22]

4.4 Arten

Es gibt verschiedene Arten von Untertiteln, die zu verschiedenen Zwecken eingesetzt werden. Zum einen unterscheidet man zwischen geschlossenen (engl. closed subtitles/closed captions) und offenen (engl. open subtitles) Untertiteln. Bei den geschlossenen Untertiteln handelt es sich um optional zuschaltbare, nicht im Bildmaterial integrierte Untertitel, die aus Textinformationen und zugehörige Zeitmarken bestehen. Offene Untertitel dagegen sind im Bildmaterial enthalten und somit unausweichlich vorhanden. Dann gibt es noch eine Mischungvariante, die vorgefertigten (engl. prerendered) Untertitel. Diese sind im Gegensatz zu den geschlossenen Untertiteln nicht als Text- sondern als Bildinformationen vorhanden, die dann, wenn benötigt, einfach über dem originalen Video eingeblendet werden. Also werden hier zwei Videos übereinander gelegt, wohingegen bei den geschlossenen Untertiteln die Textinformationen mit den Zeitmarken erst interpretiert werden müssen, bevor sie angezeigt werden können.

4.5 Kategorien

Untertitel können, selbst wenn sie die gleiche Sprache wiedergeben, die im Video gesprochen wird, verschiedenen Kategorien angehören. Im Folgenden werden einige Kategorien vorgestellt. [18]

4.5.1 Erzählende Untertitel

Das ist die am häufigsten benutzte Form von Untertiteln, bei der das Gesprochene inhaltlich wiedergegeben wird. Dies kann begleitend zu der Sprache oder die Übersetzung einer Fremdsprache sein

4.5.2 Untertitel für Hörgeschädigte

Diese Untertitel erläutern neben den Dialogen auch akustische Ereignisse, die für das Video in jeglicher Form wichtig sind.

4.5.3 Erzwungene Untertitel

Diese kommen zum Einsatz, wenn im Film etwas in einer Fremdsprache oder sogar für den Film ausgedachten Sprache vorkommt und nicht beim Synchronisieren oder woanders übersetzt werden. Dies kann bei verschiedenen Science-Fiction Filmen beobachtet werden, wenn dort Außerirdische in ihrer eigenen Sprache sprechen, aber der Zuschauer verstehen soll worum es geht. Allerdings gibt es auch Situationen, wo der Zuschauer eben diese Unterhaltung nicht verstehen soll, dort gibt es dann keine Untertitel. Die erzwungenen Untertitel werden nicht in das Bildmaterial integriert, sondern als eigene Datei bereitgestellt. Herkömmliche Blu-ray- und DVD-Player spielen die erzwungene Untertitel-Spur automatisch ab, am Computer hat man mit den entsprechenden Programmen die Möglichkeit, diese nicht mit abzuspielen.

4.5.4 Inhaltliche Untertitel

Inhaltliche Untertitel in Filmen ersetzen Szenen, in denen die Handlung mit Dialogen zwischen Darstellern vermittelt wird. Wenn sich während der Filmproduktion zum Beispiel rausstellt, dass das geplante Budget nicht mehr ausreicht, werden nur noch die nötigsten Szenen gedreht und die restlichen inhaltlichen Zusammenhänge durch Einblenden der entsprechenden Untertitel geknüpft. Dies kann auch mit voller Absicht gemacht werden, wenn es zum Beispiel eine kleine Vorgeschichte zum Film gibt, die nicht mit verfilmt wurde oder wenn ein Film an einen anderen inhaltlich anknüpft und im Vorfeld sichergestellt werden soll, dass jeder Zuschauer das nötige Vorwissen hat, um den Film zu verstehen. Ein Beispiel dafür ist die Filmreihe "Star Wars". Am Anfang eines jeden Films dieser Reihe wird die Vorgeschichte als aufsteigender Text erzählt (siehe Abb. 5).



Abbildung 5: Aufsteigende inhaltliche Untertitel am Beispiel von “Star Wars”. Quelle[23]

4.5.5 Lokalisierte Untertitel

Wenn Referenzen, Gleichnisse oder auch Sprichwörter nach der Übersetzung nicht mehr gut funktionieren oder in der Region nicht geläufig sind und somit nicht den Effekt haben, wie sie ihn unübersetzt auf die Landsleute der Originalsprache haben, dann werden sie lokalisiert, also an die jeweilige Region angepasst. Das können zum Beispiel landesweit berühmte Personen und Marken oder regionale Spezialitäten betreffen. So wird beispielsweise “Dieser Sake ist wirklich gut.” zu “Dieser Wein ist wirklich gut.”.

4.5.6 3D-Untertitel

Die bisher genannten Untertitel sind alle auf den zweidimensionalen Raum, also auf die Positionierung entlang der X- und Y-Achse, beschränkt. Bei den 3D-Untertiteln kommt noch die Z-Achse hinzu, was den Untertiteln erlaubt, vor/über dem Bild zu “schweben” und in verschiedenen Tiefen zu erscheinen. Dies kommt im 3D-Kino und auf 3D-TV-Geräten mit 3D-BluRays zum Einsatz.

4.5.7 Untertitel als besonderes Element im Film

Untertitel können bei der Filmproduktion bewusst als besonderes Mittel eingesetzt werden. Dabei kann der Untertitel zum Beispiel humoristisch eingesetzt werden, in dem ein Protagonist den eingeblendeten Untertitel bemerkt und mit ihm interagiert oder sogar die vierte Wand durchbricht und den Zuschauer auf den Untertitel anspricht.

4.6 Untertitel im deutschen Fernsehen

Momentan werden im deutschen Fernsehen noch vergleichsweise wenige Sendungen mit Untertiteln angeboten. In den USA ist die Untertitelungspflicht gesetzlich festgelegt. Dabei gibt es allerdings einige Ausnahmen und Zusätze, wodurch es auch dort keine hundertprozentige Untertitelung gibt. Dennoch wird die USA oft bei den Themen Untertitel und Barrierefreiheit als Vorbild genannt. Als Paradebeispiel wird zumeist auf die Amtseinführung von Präsident Obama am 21. Januar 2009 verwiesen. [19] Dort wurde die gesamte Veranstaltung vor Ort auf Leinwänden und im Fernsehen mit Live-Untertitelung und Sprachdolmetschern begleitet.

Aus diesem Vorbild einer wenigstens theoretischen hundertprozentigen Untertitelung hat sich in Deutschland die Bewegung "100 % Untertitel-Pflicht für alle" gebildet. Diese veröffentlicht seit 2012 monatliche Statistiken über die Untertitelungsquoten diverser deutscher Sender. Dort wird zwischen drei verschiedenen Programmtypen unterschieden: Das Vollprogramm, das Spartenprogramm und das Privatprogramm. Im Januar 2012 hat das Vollprogramm im Vergleich zu den anderen die höchste Quote mit einer Untertitelung von 31,50 %. Drei Jahre später, im Juni 2015, waren es 56,86 %. Man erkennt einen deutlichen Anstieg im Vollprogramm. Im Vergleich dazu wurde die Quote bei Privatprogrammen in der gleichen Zeit von 2,11 % auf 4,03 % angehoben. Dadurch ergibt sich für die Gesamtquote aller aufgezählten Sender von 2012 bis 2015 ein geringer Anstieg von 15,64 % auf 25,66 %.

Als Grund für dieses geringe Angebot wird oft der enorme Aufwand genannt, der die Untertitelung mit sich bringt. Laut dem Bayerischen Rundfunk (kurz: BR) ergibt sich eine besondere Herausforderung bei der Untertitelung von Live-Sendungen. Auf seiner Webseite [16] erklärt der BR, dass meistens mehrere Mitarbeiter parallel an einer Live-Sendung arbeiten und noch während der Sendung Beiträge fertiggestellt werden.

Statistik: TV-Sendungen in Deutschland mit Untertitel (Januar 2012)

Kanäle	im Zeitraum 1-31.01.2012 ermittelte Untertitel in					Typ	Gesamtquote	Gesamtquote	Gesamtquote
	Minute	Anzahl	Quote	Stunden	Tage				
WDR	23.548	630	52,75%	24	31	Vollprogramm	31,50%	20,94%	15,64%
ARD	21.153	415	47,39%	24	31	Vollprogramm			
ZDF	19.794	428	44,34%	24	31	Vollprogramm			
NDR	18.430	467	41,29%	24	31	Vollprogramm			
Radio Bremen TV	17.080	437	38,26%	24	31	Vollprogramm			
BR	16.990	380	38,06%	24	31	Vollprogramm			
MDR	11.872	268	26,59%	24	31	Vollprogramm			
RBB	10.297	217	23,07%	24	31	Vollprogramm			
SWR BW	9.735	242	21,81%	24	31	Vollprogramm			
SWR RP	9.430	205	21,12%	24	31	Vollprogramm			
SR	8.825	171	19,77%	24	31	Vollprogramm			
HR	7.895	178	17,69%	24	31	Vollprogramm			
3sat	7.755	183	17,37%	24	31	Vollprogramm			
EinsFestival	16.068	240	35,99%	24	31	Spartenprogramm	7,20%		
EinsPlus	6.780	206	15,19%	24	31	Spartenprogramm			
BR-alpha	4.455	178	9,98%	24	31	Spartenprogramm			
ZDFinfo	1.342	33	3,01%	24	31	Spartenprogramm			
arte	975	33	2,18%	24	31	Spartenprogramm			
ZDFkultur	765	19	1,71%	24	31	Spartenprogramm			
EinsExtra	525	16	1,18%	24	31	Spartenprogramm			
KiKa	490	19	1,76%	15	31	Spartenprogramm			
Phoenix	450	11	1,01%	24	31	Spartenprogramm			
ZDFneo	0	0	0,00%	24	31	Spartenprogramm			
ProSieben	5.840	150	13,08%	24	31	Privatprogramm	2,11%		
RTL	1.280	10	2,87%	24	31	Privatprogramm			
VOX	850	6	1,90%	24	31	Privatprogramm			
Kabel1	489	4	1,10%	24	31	Privatprogramm			
Sat1	0	0	0,00%	24	31	Privatprogramm			
RTL II	0	0	0,00%	24	31	Privatprogramm			
SuperRTL	0	0	0,00%	24	31	Privatprogramm			
Tele 5	0	0	0,00%	24	31	Privatprogramm			
N24	0	0	0,00%	24	31	Privatprogramm			

Quelle: 100% Untertitel-Pflicht für alle - <http://untertitel.superdeaf.de> [Facebook-Fansseite: <http://fb.com/UntertitelPflicht>]

Abbildung 6: Statistik: TV-Sendungen in Deutschland mit Untertitel (Januar 2012)Quelle [9]

Statistik: TV-Sendungen in Deutschland, Österreich und Schweiz mit Untertitel (Juni 2015)

Kanäle	im Zeitraum 1-30.06.2015 ermittelte Untertitel in					Gesamt	Typ	Gesamtquote	Gesamtquote	Gesamtquote
	Minute	Anzahl	Quote	Stunden	Tage					
ARD	40.936	963	94,76%	24	30	43200	Vollprogramm	56,86%	40,42%	26,30%
ZDF	34.047	730	78,81%	24	30	43200	Vollprogramm			
MDR	33.941	801	78,57%	24	30	43200	Vollprogramm			
NDR	32.890	871	76,13%	24	30	43200	Vollprogramm			
Radio Bremen TV	31.518	888	72,96%	24	30	43200	Vollprogramm			
WDR	29.245	732	67,70%	24	30	43200	Vollprogramm			
BR	25.105	614	58,11%	24	30	43200	Vollprogramm			
HR	22.694	478	52,53%	24	30	43200	Vollprogramm			
RBB	19.546	492	45,25%	24	30	43200	Vollprogramm			
SWR	17.585	547	40,71%	24	30	43200	Vollprogramm			
SR	16.505	456	38,21%	24	30	43200	Vollprogramm			
3sat	13.810	328	31,97%	24	30	43200	Vollprogramm			
arte	1.481	33	3,43%	24	30	43200	Vollprogramm			
EinsFestival	26.025	459	60,24%	24	30	43200	Spartenprogramm	16,69%		
ZDFneo	16.165	302	37,42%	24	30	43200	Spartenprogramm			
KiKa	6.777	250	25,10%	15	30	27000	Spartenprogramm			
EinsPlus	7.345	170	17,00%	24	30	43200	Spartenprogramm			
ARD-alpha	4.420	178	10,23%	24	30	43200	Spartenprogramm			
ZDFinfo	70	2	0,16%	24	30	43200	Spartenprogramm			
tagesschau24	15	1	0,03%	24	30	43200	Spartenprogramm			
Phoenix	0	0	0,00%	24	30	43200	Spartenprogramm			
ZDFkultur	0	0	0,00%	24	30	43200	Spartenprogramm			
ProSieben	7.510	138	17,38%	24	30	43200	Privatprogramm 1	5,37%	4,10%	
Kabel1	3.700	30	8,56%	24	30	43200	Privatprogramm 1			
Sat1	3.665	31	8,48%	24	30	43200	Privatprogramm 1			
ProSiebenMAXX	1.375	11	3,18%	24	30	43200	Privatprogramm 1			
Kabel 1 Classic	0	0	0,00%	24	30	43200	Privatprogramm 1			
ProSiebenFun	0	0	0,00%	24	30	43200	Privatprogramm 1			
Sat.1 Gold	0	0	0,00%	24	30	43200	Privatprogramm 1			
VOX	3.580	44	8,29%	24	30	43200	Privatprogramm 2	3,29%		
RTL	3.365	40	7,79%	24	30	43200	Privatprogramm 2			
SuperRTL	1.075	10	2,49%	24	30	43200	Privatprogramm 2			
RTL Nitro	505	10	1,17%	24	30	43200	Privatprogramm 2			
N-TV	0	0	0,00%	24	30	43200	Privatprogramm 2			
RTL II	0	0	0,00%	24	30	43200	Privatprogramm 2			
N24	0	0	0,00%	24	30	43200	Privatprogramm 3			
ORF2	31.379	835	72,64%	24	30	43200	Österreich			
ORF eins	30.800	855	71,30%	24	30	43200	Österreich			
SF1	21.600	615	50,00%	24	30	43200	Schweiz			
SF zwei	16.935	246	39,20%	24	30	43200	Schweiz			

Quelle: 100% Untertitel-Pflicht für alle - <http://untertitel.superdeaf.de> [Facebook-Fansseite: <http://fb.com/UntertitelPflicht>]

Abbildung 7: Statistik: TV-Sendungen in Deutschland mit Untertitel (Juni 2015), Quelle [10]

4.7 Automatische Generierung

Die automatische Untertitelgenerierung ist sowohl für aufgezeichnete als auch für Live-Sendungen interessant. Im Grunde geht es um Spracherkennung beziehungsweise um eine Text-to-speech-Funktion. Auf dem Smartphone funktioniert das mit "Siri" von Apple oder der Spracherkennung auf Android schon recht gut. Allerdings kann man die Erkennung dort trainieren und auf den Benutzer anpassen. Da es allerdings in Filmen viele verschiedene Darsteller gibt, ist es schwierig ein Profil anzulegen, das bei der Spracherkennung helfen könnte. Dazu kommt, dass im Film nicht immer ruhig und deutlich gesprochen wird. Oft gibt es Hintergrundmusik und -geräusche, die die Sprache übertönen und eine automatische Erkennung erschweren.

YouTube bietet Nutzern, die ein Video hochladen, die Möglichkeit, Untertitel automatisch generieren zu lassen. Es folgt ein Beispiel, bei dem ein Ausschnitt aus dem Film "Burn after Reading" mithilfe von YouTube untertitelt wurde.

Erwartet:

"Ist das Ziegenkäse? - Chèvre, ja. Das ist ein Ziegenkäse."

Erzeugt:

"ist das so sie genieße"

"Schäfer ja das ist einzig die Katz"

Man erkennt, dass das Ergebnis allenfalls humoristischen Charakter hat, für die automatische Untertitelung von ganzen Filmen aber ungeeignet ist. Da die Spracherkennung auf dem Smartphone auf Englisch besser funktioniert als auf Deutsch, folgt ein weiteres Beispiel auf Englisch. Es geht um einen Ausschnitt aus einer Dokumentation über eine Gebirgslandschaft.

Erwartet:

"It's spring and the peaks are covered in clouds and snow."

Erzeugt:

"its brain and the pizza covered in clouds in snow"

Das Ergebnis ist ähnlich zu bewerten wie das für das deutsche Beispiel.

4.8 Formate

4.8.1 DVD-Video

Bei diesem Format handelt es sich um das am meisten verbreitete Format für DVDs in Asien, Australien, Europa und Nord-Amerika. Dieses Format kann bis zu 32 verschiedene Untertitel-Spuren enthalten. Die Untertitel sind dabei jeweils als Bitmaps mit maximal 4 Farben enthalten. Dadurch können diese Untertitel weniger fein und gleichmäßig aussehen, als im Videomaterial enthaltene Untertitel. Die Untertitel sind zusammen mit den Video- und Audioinformationen gebündelt in einer ".vob"-Datei.

4.8.2 MicroDVD

Im Gegensatz zu DVD-Video, sind hier die Untertitel in einer eigenen Datei gelagert. Seinen Namen hat dieses Format vom MicroDVD-Player, einem Software-Player, der für die Wiedergabe von Videos mit Untertiteln programmiert wurde. Benutzt wird die Dateiformatendung ".sub".

Die Syntax für den einfachsten Fall lautet:

```
{start-frame}{stop-frame}Text
```

Eine Zeitmarke wird mit Start- und End-Bildern festgelegt. Daran erkennt man, dass die Präzision dieses Formats abhängig von der Anzahl der Bilder ist, die pro Sekunde dargestellt werden. Nach der Zeitmarke wird einfach der gewünschte Text angegeben. Dabei gibt es noch einige "control codes", die es erlauben, die Schriftart, -farbe und -größe zu manipulieren. Diese können zwischen Text und Zeitmarken in beliebiger Reihenfolge angehängt werden. Im Folgenden werden kurz die einzelnen "control codes" vorgestellt und jeweils an einem Beispiel demonstriert.

<u>Control code</u>	<u>Parameter</u>	<u>Beispiel</u>	<u>Ergebnis</u>
<u>{y:fontstyle}</u>	[i, b, u, s]	{0}{30}{y:b} Test	Test
<u>{f:fontname}</u>	[Arial, Times, ...]	{0}{30}{f:Georgia} Test	Test
<u>{s:size}</u>	[... 8, 9, 10, ...]	{0}{30}{s:8} Test	Test
<u>{c:\$BBGGRR}</u>	[0000FF, ...]	{0}{30}{c:\$FF0000} Test	Test
<u>{P:X,Y}</u>	[0, ...], [0, ...]	{0}{30}{P:10,0} Test	Test
<u>{H:character-set}</u>	[Greek, Chinese, ...]	{0}{30}{H:Chinese} Test	范甘

Abbildung 8: Tabellarische Darstellung der 'Control-Codes'

4.8.3 SubRip

SubRip ist ein Programm für Windows, das Untertitel aus Videos extrahieren kann. Gleichzeitig steht SubRip für die Datei, die dabei entsteht. Sie hat die Dateierdung “.srt”. Sie enthält neben den Texten auch die zugehörigen Zeitmarken und kann außerdem Informationen über die Position und die zu verwendende Schriftart und -farbe haben.

Ein Beispiel für den einfachsten Fall lautet:

```
1
00:00:12,500 --> 00:00:15,500
Sample Text.
```

Dieses Beispiel zeigt, dass die Einträge hier aus mehreren Zeilen bestehen. In der ersten Zeile steht ein numerischer Index über den jeder einzelne Eintrag identifiziert werden kann. Die zweite Zeile enthält die Textmarke in folgendem Format:

```
HH:MM:SS,mmm --> HH:MM:SS,mmm
```

Dabei stehen H, M, S und m in der Reihenfolge für Stunden, Minuten, Sekunden und Millisekunden.

Zu beachten ist, dass das ”->” in der Mitte genauso zur Syntax gehört wie das Zeitformat. In der nächsten Zeile folgt dann der Text. Dieser kann sich auf beliebig viele Zeilen danach erstrecken, allerdings ist wichtig, dass nach dem Text und vor dem nächsten Eintrag eine Leerzeile kommt.

Optional können auch Koordinaten für ein Rechteck angegeben werden, in dem der Text angezeigt werden soll. Dies wird in der gleichen Zeile angegeben, wie die Zeitmarke:

```
00:00:12,500 --> 00:00:15,500 X1:63 X2:223 Y1:43 Y2:58
```

Ähnlich wie beim MicroDVD-Format gibt es hier die Möglichkeit den Text zu gestalten. Dabei wird der zu gestaltende Teil des Textes mit einem “tag” umschlossen, ähnlich wie in HTML. Es folgen Beispiele für jeweils einen Text in ‘bold’, ‘italic’ und ‘underlined’:

```
<b>{This text will be displayed bold}</b>
<i>{This text will be displayed italic}</i>
<u>{This text will be displayed underlined}</u>
```

4.8.4 WebVTT (Web Video Text Tracks)

WebVTT ist ein W3C Standard für die Darstellung von zeitabhängigen Texten in HTML5. Aus einer Diskussion darüber, welche Untertitel-Formate HTML5 unterstützen sollte, entstand ein neues, teilweise auf dem SRT-Format aufbauende Format, das WebSRT(Web Subtitle Resource Tracks) genannt wurde. Auch wenn es nicht komplett kompatibel zum SubRip-Format war hat es die gleiche Dateiendung benutzt. Als dann in HTML5 der “<track>”-tag eingeführt wurde, wurde das Format angepasst und in WebVTT umbenannt. Die Dateiformatendung “.vtt” wurde eingeführt. Auch wenn die Spezifikationen für WebVTT noch nicht komplett ausgearbeitet sind, unterstützen die gängigsten Browser schon dieses Format. [6]

Ein Beispiel für den einfachsten Fall lautet:

```
WEBVTT
00:01.000 --> 00:04.000
This is a simple sample.
```

Eine .vtt-Datei fängt immer mit “WEBVTT” in der ersten Zeile. Man kann sehen, dass der Index, den es bei den SRT-Dateien geben muss, hier optional ist. Dieser muss hier auch nicht numerischer Natur sein. Die Zeitmarke sieht noch ähnlich aus wie die, aus dem SRT-Format, allerdings sind die Stunden jetzt auch optional und anstelle des Kommas zum Trennen der Millisekunden wird hier ein Punkt verwendet. In der darauf folgenden Zeile kommt wie gewohnt der Text.

Erstmalig sind hier auch Anmerkungen oder Kommentare in der Datei erlaubt. Diese werden mit dem Schlüsselwort “NOTE” eingeleitet und können wie folgt aussehen:

```
WEBVTT
00:04.000 --> 00:04.100
This is some text.
```

```
NOTE Should the text above really only show for 100 ms?
```

```
00:05.000 --> 00:09.000
This is another text.
```

Neben den gestalterischen Möglichkeiten, die im SRT-Format schon möglich waren, kommen hier noch diverse Möglichkeiten hinzu. Es kann nicht nur die Textbox positioniert werden, sondern auch der Text selbst innerhalb der Box. Man kann also eine Zentrierung oder Links-/Rechtsbündigkeit des Textes festlegen. Außerdem können bestimmte Regionen für verschiedene

Sprecher deklariert werden. So kann eine Art "style" deklariert werden, der dann immer wieder benutzt werden kann.

Dieses Format bietet noch viele weitere Vorteile und Möglichkeiten im Zusammenhang mit HTML5. Da für diese Seminararbeit das SRT-Format von Bedeutung ist, werde ich die Aufzählung für das WebVTT-Format an dieser Stelle beenden.

5 Die Lesbarkeit von Untertiteln

Die allgemeine Lesbarkeit von Texten ist abhängig von der Zeit, die benötigt wird, um einen Text zu lesen in Verbindung mit der Mühe die dabei aufgewandt wird. Dabei ist zwischen langen und kurzen Texten zu unterscheiden. Bei längeren Texten findet der Leser einen bestimmten Leserrhythmus in dem er angenehm lesen kann. Dort kann eine Serifenschrift das Lesen erleichtern, da die Serifen dem Leser eine optische Führung entlang der Schriftlinie bieten. Serifen garantieren allerdings nicht immer eine gute Lesbarkeit. Es gibt sowohl gut als auch schlecht lesbare Serifenschriften. Die geläufigste Serifenschrift ist "Times New Roman" und die geläufigsten serifenlosen Schriften sind "Arial" und "Helvetica". [15]

Bei sehr kurzen Texten findet der Leser keinen Rhythmus, er "scannt". Bei Untertiteln ergibt sich eine besondere Herausforderung durch das abwechselnde "Scannen" des Textes und der schnellen Bilderfassung.

5.1 Typografie

Im Allgemeinen bezeichnet die Typografie (griech.: "typos" (Muster, Gestalt), "graphein" (schreiben)) die Kunst des Druckens, bei dem die Texte aus beweglichen Lettern zusammengesetzt werden. Im Kontext dieser Arbeit beschreibt Typografie den Gestaltungsprozess von Texten, ohne die Verwendung von beweglichen Lettern. Dafür geht es um Schrifttypen und -größen, Farben für Vorder- und Hintergründe und die Darstellung und Anordnung auf dem Bildschirm.

5.2 Schriftart und -größe

Für Untertitel sind serifenlose Schriften am besten geeignet [15]. Sie sollten dabei für kleine Schriftgrößen optimiert sein, wie z.B. "Verdana", "SegoeUI" oder "Myriad Web".

5.3 Einblendedauer

Die durchschnittliche Lesegeschwindigkeit liegt bei 150 bis 180 Wörtern pro Minute. Die Einblendedauer sollte sich daran orientieren. Optimal ist eine Einblendedauer von 4 Sekunden pro Untertitel, dabei sollte es aber nicht länger als 6 bis 8 Sekunden eingeblendet bleiben, da der Nutzer sonst zum mehrfachen Lesen neigt. [20]

5.4 Kontraste

Mit am stärksten ausschlaggebend für eine gute Lesbarkeit sind hohe Kontraste. Durch sie kann das Auge die Schrift besser vom Hintergrund abgrenzen kann.

6 Umsetzung auf Android

6.1 Vergleich zum TV-Gerät

Für die Darstellung von Untertiteln auf einem Smartphone ergibt sich im Vergleich zur Darstellung auf einem TV-Gerät als erstes ein klarer Unterschied: Die Displaygröße. TV-Geräte sind in der Regel deutlich größer als Smartphones - auch wenn der Trend zu immer größeren Smartphones geht. Aus einem kleineren Display folgt weniger Platz für die Darstellung der Untertitel. Dort kommt die Frage auf, ob es überhaupt möglich ist, Untertitel auf einem kleinen Display so darzustellen, dass sie immer noch lesbar sind und dabei nicht das ganze Video verdecken. Heutzutage ist Full HD mit einer Auflösung von 1920 x 1080 Pixeln ein Standard für sowohl TV-Geräte als auch Smartphones. Das bedeutet, dass das TV-Gerät, das etwa 2 bis 3 Meter vom Zuschauer entfernt ist, die gleiche Auflösung bietet, wie das Smartphone, das nur wenige Zentimeter vor dem Gesicht des Nutzers gehalten wird. Durch diesen Distanzunterschied wird das vermeintliche Größenproblem wieder aufgehoben (siehe Abb. 9). Wenn man eine eventuelle Weit-/Kurzsichtigkeit oder andere nutzerbedingte Einschränkungen außer Acht lässt, dann ist ein solches Smartphone kein schlechteres Gerät zur Wiedergabe von Videos mit Untertiteln als ein TV-Gerät.

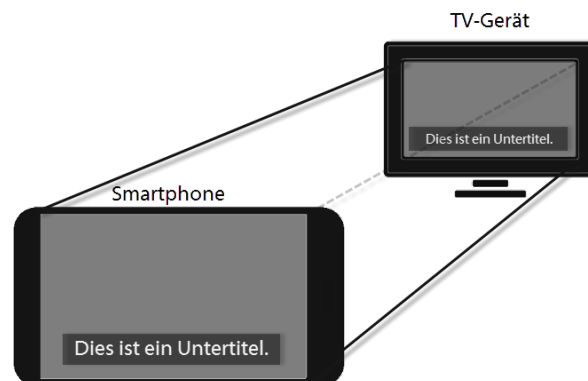


Abbildung 9: Effektive Größe eines Smartphones im Vergleich zum TV-Gerät

6.2 Bekannte Möglichkeiten

6.2.1 Der native MediaPlayer

Der MediaPlayer oder die MediaPlayer-Klasse ist eine leichtgewichtige Komponente zur Medienwiedergabe aus dem Android-Framework. Der Media-

Player bietet eine Möglichkeit komfortabel und schnell Untertitel mitabzuspielen. Dabei werden momentan nur .SRT-Dateien unterstützt [11]. Zuerst wird eine Instanz der Klasse initialisiert. Anschließend kann man mit der Methode "addTimedTextSource(String location)" eine Online- oder Offline-Ressource angeben. Um über das Ereignis eines Untertitels informiert zu werden, muss man nur noch mit Hilfe der Methode "setOnTimedTextListener(MediaPlayer.OnTimedTextListener listener)" einen Listener angeben.

Beispiel:

```
MediaPlayer p = MediaPlayer.create(context, vidLocation);  
p.addTimedTextSource(mTestUrl);  
p.setOnTimedTextListener(mListener);
```

Der MediaPlayer.OnTimedTextListener muss nur eine Methode überschreiben und kann wie folgt aussehen:

```
mListener = new MediaPlayer.OnTimedTextListener() {  
    @Override  
    public void onTimedText(MediaPlayer mp, TimedText tt) {  
        // Do Something with the TimedText.  
    }  
});
```

In dieser Methode muss jetzt dafür gesorgt werden, dass der Text, den das TimedText-Objekt hält, angezeigt wird. Dies kann, wie in dem Beispiel gezeigt einfach durch das Zuweisen des Textes an eine TextView geschehen. So weit ist der Aufwand gering und der Umgang mit dem MediaPlayer komfortabel. Allerdings gibt es ein Problem bei dieser Lösung. Es gibt keine Methode, die aufgerufen wird, wenn der Text wieder ausgeblendet werden soll. Und das TimedText-Objekt bietet nur die folgenden öffentlichen Methoden an:

```
timedText.getText(), returns String  
timedText.getBounds(), returns Rect
```

Somit kann man nur den Text und das Rechteck, in dem dieser angezeigt werden soll, abfragen. Das bedeutet, wenn nun eine Passage im Untertitel kommt, wo lange Zeit kein neuer Text anzuzeigen ist, dann wird weiterhin der alte Text angezeigt. Das verhindert eine korrekte Darstellung der Untertitel und sorgt im schlimmsten Fall dafür, dass der Zuschauer entnervt die Wiedergabe stoppt.

6.2.2 ExoPlayer

Neben dem einfachen und schnell zu verwendenden MediaPlayer bietet das Android-Framework auch noch die Klassen MediaCodec und MediaExtractor [12] um eigene Player zu bauen. Mit der Open-Source ExoPlayer-Komponente stellt Google einen Kompromiss aus der schnellen Einsetzbarkeit und einem erweiterbaren Player zur Verfügung. Der ExoPlayer ist, wenn man die reine Medienwiedergabe nutzen möchte, ähnlich einfach zu benutzen wie der MediaPlayer. Sobald es allerdings darüber hinausgeht gibt es ca. 230 Klassen [13], die einem dabei als Auswahl aus einem Baukasten dienen. Als Vorteil gegenüber dem einfachen MediaPlayer unterstützt der ExoPlayer unter anderem das sogenannte "Dynamic adaptive streaming over HTTP"(kurz:"DASH") [12], welches hochqualitatives Streamen über HTTP ermöglicht.

Da es bisher kaum Code-Beispiele für die jeweiligen Möglichkeiten gibt, ist ein intensives Einarbeiten in viele Klassen nötig, bevor man sich aus diesem Baukasten bedienen kann. Wenn man den ExoPlayer nicht bereits in seiner App verwendet, ist es fraglich, ob eine Umstellung auf den ExoPlayer den Aufwand wert ist. Wenn die betreffende App allerdings noch in der Planung ist oder noch keine Player-Komponente verwendet wird, bietet der ExoPlayer mit den erwähnten Klassen eine Vielzahl von Vorteilen und Möglichkeiten. Aber auch hier ist die Möglichkeit der Untertitelwiedergabe an den Player gebunden

6.3 SubtitleController

Bisher gab es nur die Möglichkeit, Untertitel mit Hilfe des jeweiligen Players anzuzeigen. Im Folgenden wird der "SubtitleController" vorgestellt, eine Komponente, die Untertitel ohne eine Bindung zu einem Player anzeigen kann.

Der SubtitleController besteht im Grunde aus vier Klassen:

- SubtitleController
- SubtitleParser
- SubTextElement
- SubTextElementList

6.3.1 Funktionsweise

Der SubtitleController stellt die Hauptklasse dieser Komponente dar, die einzige Klasse, an der zur Darstellung von Untertiteln Methoden aufgerufen werden müssen. Der einfachste Konstruktor bekommt einen Context, eine Liste mit Parsern, die er akzeptieren soll und dem Pfad zu der Untertiteldatei, die angezeigt werden soll. Es ist dabei egal, ob es sich um eine online oder offline Ressource und ob es sich um den Pfad zu einer bestimmten Datei oder zu einem Verzeichnis handelt. Der übergebene Pfad wird kontrolliert und sobald Dateien gefunden wurden, wird in der Liste der zu akzeptierenden Parsern geguckt, ob einer davon die Datei parsen kann. Dazu gibt es die Methode "canParse(String location)", die von allen Parsern überschrieben werden muss. Standardmäßig gibt es einen Parser für SRT-Dateien. Dabei handelt es sich um eine minimal optimierte Version vom SRT-Format. Die Möglichkeit in der Datei mittels "<color>"-tags verschiedene Farben auf verschiedene Sätze anzuwenden habe ich umgebaut in die Möglichkeit verschiedene Sprecher von 1 bis 6 mit den tags "<1>" bis "<6>" zu deklarieren. So kann während der Wiedergabe die Farbwahl dynamisch für einzelne Sprecher angepasst und das Benutzererlebnis verbessert werden. Um weitere Parser zu implementieren muss man die Klasse 'SubtitleParser' erweitern und dort die folgenden Methoden überschreiben: beginlstlisting

- protected abstract String getSupportedFileExtension()
- protected abstract boolean canParse(String fileName)
- protected abstract void parseLine(String line) endlstlisting

Dabei gibt "getSupportedFileExtension()" einfach einen String mit der Dateierweiterung zurück, die unterstützt werden soll. Die Methode "canParse(...)" prüft im Standardfall, ob die Datei die unterstützte Dateierweiterung hat, kann aber nach Belieben implementiert werden.

Zuletzt noch die eigentliche Methode, die den Parser ausmacht, die Methode 'parseLine(...)'. Diese wird so oft aufgerufen, wie die zu parsende Datei Zeilen hat. Dort muss dann entschieden werden, wie jede einzelne Zeile zu behandeln ist.

Wenn das Parsen erfolgreich war, hat der SubtitleController eine Map, die die Namen der Dateien auf eine entsprechende SubTextElementList abbildet. Jede dieser Listen hält SubTextElements und repräsentiert eine Untertitel-Datei. Am SubtitleController kann nun abgefragt werden, welche SubTextElementLists es gibt. Dies kann genutzt werden um dem Benutzer

eine Auswahl der verschiedenen Untertitel anzubieten. Entscheidet sich der Benutzer nun für einen Untertitel, wird die Entsprechende SubTextElementList im SubtitleController als aktuelle Liste gesetzt. An dieser Stelle kann ein beliebiger Player zur Wiedergabe eines Videos genutzt werden. Dafür hat der SubtitleController die Methode "onUpdate(Textview view, int millis)". Sie verlangt also das Element, auf dem der Text dargestellt werden soll und die Position in Millisekunden, die aus der Untertitel-Datei genommen werden soll. Im Standardfall werden Untertitel auf der Android-Plattform mittels einer TextView dargestellt.

Man ist als Programmierer also mit der SubtitleController-Komponente völlig unabhängig vom verwendeten Player und kann diesen mühelos austauschen. Mühelos bedeutet in dem Fall, dass die Untertiteldarstellung nicht angepasst werden muss. Ein weiterer Vorteil ist, dass man die Kontrolle darüber hat in welchen Abständen der Untertitel aktualisiert werden soll. Und die entscheidende Verbesserung gegenüber der Untertiteldarstellung mit dem MediaPlayer ist die Tatsache, dass der SubtitleController bei jedem Aktualisierungsvorgang über die Methode "onUpdate(...)" den anzuzeigen Text leert, wenn kein Untertitel für die angegebene Zeit gefunden wurde.

6.4 Demo

Der SubtitleController an sich bietet keinerlei GUI-Elemente, zum Testen entstand aber im Laufe dieser Arbeit ein kleiner DemoPlayer auf Basis des MediaPlayer, der zur Untertiteldarstellung den SubtitleController benutzt.



Abbildung 10: Übersicht des DemoPlayers

Auf Abb. 10 kann man alle wichtigen Elemente des DemoPlayers sehen. Unten ist, wie üblich bei Playern, die sogenannte ControlBar oder BottomBar. Über sie wird die Wiedergabe gestartet/gestoppt. Außerdem kann man mit einer Suchleiste im Video vor und zurück springen. Rechts daneben befindet sich die Anzeige mit der aktuellen Position im Video und zuletzt gibt es außen rechts einen Knopf um in den Vollbild-Modus zu wechseln. Oben sieht man die TopBar bzw. HeaderBar. Sie hat links einen Knopf, der eine Liste mit den verschiedenen Untertiteln anzeigt. Im SubtitleController sind diese als SubTextElementLists implementiert. Mittig befindet sich der Name der derzeit abgesehenen Datei und rechts daneben befinden sich jeweils ein Knopf zum Ändern der Schriftart und der vorher erwähnten sprecherspezifischen Textfarben.

In der Mitte befindet sich die "Leinwand" und auf ihr befindet sich die TextView für die Untertiteldarstellung. Die Untertitel können mit verschiedenen Gesten manipuliert werden. Zum einen kann man sie mit zwei Finger und sogenannten "pinch" und "zoom" Gesten verkleinern und vergrößern, zum anderen kann man die vertikale Position des Textes durch einfaches Wischen mit einem Finger manipulieren.

Der DemoPlayer kann nur Videos und Untertitel abspielen, die sich auf der SD-Karte des Gerätes befinden und dort in einem Order mit dem Namen "BA_SampleSources" liegen, der selbst im Hauptverzeichnis der SD-Karte liegt. Dort muss jedes Video mit den entsprechenden Untertiteln in einem eigenen Unterordner liegen. Auch wenn der SubtitleController es anders kann, macht der DemoPlayer momentan davon nicht Gebrauch.

7 Résumé

Das Thema Untertitel ist ein Thema, das immer relevanter wird. Nicht nur Hörgeschädigte profitieren davon. Kleine Kinder können auf diese Weise zum Beispiel leichter das Lesen erlernen. Die Nachfrage ist groß und tatsächlich hat sich einiges in den vergangenen Jahren getan. Ob es allerdings jemals eine hundertprozentige Untertitelung gibt, wage ich noch zu bezweifeln. Auch wenn die Spracherkennung ziemlich fortgeschritten ist, bleibt es doch noch eine schwierige Aufgabe für die Zukunft. Die Möglichkeiten, die beim DemoPlayer angesprochen wurden, sind nicht zu unterschätzen. Die Player, die bisher angeboten werden, die tatsächlich auch eine Untertitelunterstützung bieten, bieten gleichzeitig noch zu wenig Freiheit im Umgang mit diesen. Es ist schwierig eine optimale Untertitelgröße für alle zu finden,

besser ist eine dynamisch anpassbare Größe. Und die Positionierung der Untertitel kann derjenige, der die Untertitel schreibt noch so gewissenhaft festlegen. Es ist einfach nicht möglich jedermanns Interesse dabei zu vertreten, weil es für jeden unterschiedliche Ausschnitte auf dem Bildschirm gibt, die er in dem Moment am interessantesten findet.

Mit dem SubtitleController biete ich eine Grundlage für eine unabhängige Untertitel-Komponente und der DemoPlayer kann sicherlich für etwas Anregung zu neuen Funktionen bei bestehenden Playern sorgen.

Literatur

- [1] Karsten Ständer,
"Statistische Angaben zur Hörschädigung in Deutschland",
<http://www.schwerhoerigen-netz.de/main/statistik.asp?inhalt=statistik03l>,
17:02 Uhr, 01.07.2015
- [2] comScore via Statista.com,
"Marktanteile der Betriebssysteme an der Smartphone-Nutzung in Deutschland von Dezember 2011 bis Februar 2015",
<http://de.statista.com/graphic/1/170408/marktanteile-der-betriebssysteme-fuer-smartphones-in-deutschland.jpg>,
11:02 Uhr, 07.07.2015
- [3] Bundesministerium der Justiz und für Verbraucherschutz,
"Gesetz zur Gleichstellung behinderter Menschen" (§4),
http://www.gesetze-im-internet.de/bgg/___4.html,
14:36 Uhr, 02.07.2015
- [4] Sergey Polezhaka für Bild.de,
"Bild Reporter Ronzheimer vor einem zerstörten Haus in Mironovskaya", <http://www.bild.de/sport/fussball/bayern-muenchen/bild-reporter-im-krisengebiet-39798578.bild.html>,
17:45 Uhr, 02.07.2015
- [5] Flossmanuals.net,
"Subtitling - File formats", http://en.flossmanuals.net/video-subtitling/ch005_file-formats/,
08:32 Uhr, 03.07.2015
- [6] Tom Leadbetter,
"Video Subtitling and WebVTT",
<http://html5doctor.com/video-subtitling-and-webvtt/>,
10:26 Uhr, 07.07.2015
- [7] Multimedia.cx,
"VOBsub", <http://wiki.multimedia.cx/index.php?title=VOBsub>,
08:14 Uhr, 03.07.2015
- [8] Dipl. Des. Manuela Rohde,
"Lesbarkeit am Bildschirm .Digitale Typografie",
http://www.gestaltung.hs-magdeburg.de/fileadmin/user_upload/seiten/lehrgebiete/interaction_design

*/01_ba_industrial_design/04_visual_communication/WS_11_12/
3_Lesbarkeit_am_Bildschirm_Digitale_Typografie.pdf,
13:42 Uhr, 09.07.2015*

- [9] *untertitel.superdeaf.de,
"Statistik: TV-Sendungen in Deutschland mit Untertitel - Jahr
2012",
[http://untertitel.superdeaf.de/download/Statistik_Untertitel-
DeutschlandTV_2012.pdf](http://untertitel.superdeaf.de/download/Statistik_Untertitel-
DeutschlandTV_2012.pdf),
17:23 Uhr, 09.07.2015*
- [10] *untertitel.superdeaf.de,
"Statistik: TV-Sendungen in Deutschland, Österreich und
Schweiz mit Untertitel - Jahr 2015",
[http://untertitel.superdeaf.de/download/Statistik_Untertitel-
DeutschlandTV_2015.pdf](http://untertitel.superdeaf.de/download/Statistik_Untertitel-
DeutschlandTV_2015.pdf),
17:30 Uhr, 09.07.2015*
- [11] *developer.android.com,
"MediaPlayer",
<http://developer.android.com/reference/android/media/MediaPlayer.html>,
04:12 Uhr, 12.07.2015*
- [12] *developer.android.com,
"ExoPlayer", [http://](http://developer.android.com/guide/topics/media/exoplayer.html)
developer.android.com/guide/topics/media/exoplayer.html,
04:12 Uhr, 12.07.2015*
- [13] *github.com/google,
"ExoPlayer", [http://](http://github.com/google/ExoPlayer/tree/master/library/src/main)
github.com/google/ExoPlayer/tree/master/library/src/main,
05:51 Uhr, 12.07.2015*
- [14] *Boris Guenebaut,
"Master Thesis: Automatic Subtitle Generation for Sound in
Videos", [http://hv.diva-
portal.org/smash/get/diva2:241802/FULLTEXT01.pdf](http://hv.diva-portal.org/smash/get/diva2:241802/FULLTEXT01.pdf),
07:16 Uhr, 14.07.2015*
- [15] *Martina Ljubenova,
"Auswirkung von
Schriftart und Satz auf die Lesbarkeit und Lesegeschwindigkeit",
http://www.medien.ifl.lmu.de/fileadmin/mimuc/mmi_ws0304/exercise*

/aufsaetze/Martina_Ljubenova.html,
08:03 Uhr, 14.07.2015

- [16] br.de,
"TV für Hörgeschädigte - Untertitelung",
<http://www.br.de/service/programm/untertitel/tv-fuer-hoergeschaedigte100.html>,
17:46 Uhr, 16.07.2015
- [17] daserste.de,
"Barrierefreie Angebote im Ersten",
<http://www.daserste.de/specials/service/barrierefreie-angebote-ard100.html>,
18:03 Uhr, 16.07.2015
- [18] wikipedia.org,
"Subtitles (captioning)",
[https://en.wikipedia.org/wiki/Subtitle_\(captioning\)](https://en.wikipedia.org/wiki/Subtitle_(captioning)),
18:32 Uhr, 16.07.2015
- [19] Oliver Markwirth,
"100-prozentige Untertitelung in den USA - Wahrlich?",
http://sign-dialog.de/files/Dokumenten/UT_in_Amerika01.pdf,
18:32 Uhr, 16.07.2015
- [20] FH Potsdam,
"Zur Qualität von Untertiteln", *<http://forge.fh-potsdam.de/emw/filmlanguage/deutsch/untertitel/04qualitaet.htm>,*
18:32 Uhr, 16.07.2015
- [21] DVD "King of Queens", 2008, Staffel 8, Folge 3, 01:10:00
- [22] DVD "Star Wars Episode II: Angriff der Klonkrieger", 2002,
01:00:00

Ich versichere, dass ich die Bachelorarbeit im Studiengang Software-System- Entwicklung selbstständig verfasst und keine anderen als die angegebenen Hilfsmittel – insbesondere keine im Quellenverzeichnis nicht benannten Internet- Quellen – benutzt habe. Alle Stellen, die wörtlich oder sinngemäß aus Veröffentlichungen entnommen wurden, sind als solche kenntlich gemacht. Ich versichere weiterhin, dass ich die Arbeit vorher nicht in einem anderen Prüfungsverfahren eingereicht habe und die eingereichte schriftliche Fassung der auf dem elektronischen Speichermedium entspricht.

Ort, Datum

Unterschrift