

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

VIDEO-MANUÁL JAKO APLIKACE NA MOBILNÍM TELEFONU S ROZPOZNÁNÍM OBJEKTŮ

MOBILE VIDEO MANUAL WITH AUTOMATIC OBJECT RECOGNITION

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

STANISLAV NEDBÁLEK

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. MICHAL ŠPANĚL, Ph.D.

BRNO 2013

Abstrakt

Tato bakalářská práce se zabývá možnostmi uplatnění počítačového vidění a rozšířené reality na mobilní platformě – operačním systému Android. Cílem této práce je demonstrovat možnosti praktického využití těchto technologií na reálných případech a situacích dneška. Práce se zaměřuje na návrh a implementaci mobilní aplikace, která je schopná pro automaticky detekovaný objekt zobrazit video manuál. Vytvořená aplikace byla testována se skupinou uživatelů, která prokázala použitelnost a intuitivnost aplikace.

Abstract

This thesis deals with potential of us computer vision and augmented reality on mobile platform – OS Android. The goal of this thesis is to demonstrate practical potential of mentioned technologies on nowadays real situations. This thesis aims to design and implementation of mobile application, which is able to show video manual for automatic detected object. The created application has been tested on group of users, which approved useability and intuitivness of this application.

Klíčová slova

rozšířená realita, marker, detekce markeru, SURF, RANSAC, homografie, vuforia, android, NDK

Keywords

augmented reality, marker, marker detection, SURF, RANSAC, homografie, vuforia, android, NDK

Citace

Stanislav Nedbálek: Video-manuál jako aplikace na mobilním telefonu s rozpoznáním objektů, bakalářská práce, Brno, FIT VUT v Brně, 2013

Video-manuál jako aplikace na mobilním telefonu s rozpoznáním objektů

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Ing. Michala Španěla. Všechny literární zdroje a publikace, které jsem použil, jsou uvedeny v seznamu použité literatury

.....

Stanislav Nedbálek

14. května 2013

Poděkování

Chci poděkovat svému konzultantovi Ing. Michalu Španělovi za pomoc a cenné rady při tvorbě této bakalářské práce, Martinu Velasovi za praktické rady a zkušenosti s psaním bakalářské práce, Dominice Fabčínové za propůjčení hlasu do audio nahrávek. A také všem, kteří se účastnili testování.

© Stanislav Nedbálek, 2013.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

1	Úvod	3
2	Rozšířená realita	4
2.1	Rozšířená realita obecně	4
2.2	Rozšířená realita podrobněji	5
2.3	Historie RR	8
3	Detekce markeru pro potřeby rozšířené reality.	11
3.1	Detekce hran	11
3.2	Vyhledání markeru	12
3.3	Homografie	13
3.4	Princip použití detekce markeru v rozšířené realitě	16
4	Nástroje pro tvorbu RR	17
4.1	ARToolkit	17
4.2	Vuforia SDK	18
4.3	Srovnání	19
5	Existující aplikace	20
5.1	Aurasma	20
5.2	Wikitude	20
5.3	Layar	21
5.4	Shrnutí	21
6	Návrh aplikace VideoManual	22
6.1	Motivace	22
6.2	Cíle práce	23
6.3	Aplikace	23
6.4	Návrh testování	25
7	Implementace	26
7.1	Vuforia SDK	26
7.2	Struktura aplikace	27
8	Vyhodnocení výsledků a možnosti rozšíření	29
8.1	Průběh testování	29
8.2	Uživatelské rozhraní	30
8.3	Použitelnost	30
8.4	Shrnutí výsledků	31

8.5	Možná vylepšení	32
9	Závěr	34
A	Obsah CD	38
B	Vzorové obrázky objektů	39
C	Plakát	41

Kapitola 1

Úvod

V dnešní době se svět ubírá směrem modernizace a minimalizace. Je kolem nás spousta zařízení, k nimž máme k dispozici pouze ovládací panel, případně jiné rozhraní pro manipulaci. Většina těchto zařízení poté funguje jako černá skříňka – prostředek jímž dosáhneme výsledku bez nutnosti znalosti implementačních specifík daného zařízení. Téměř ke všemu tedy dnes potřebujeme návod k použití. Návodů jsou zpravidla psány technicky, formálně. Ani dodržení těchto obecných zásad pro psaní manuálů nemusí vést k rychlému a plnému pochopení možností ovládání.

V této práci popisují tvorbu aplikace VideoManual pro mobilní operační systém *Android* s využitím knihovny pro rozšířenou realitu (augmented reality) – *Vuforia*. Za hlavní cíl této práce kladu vytvoření intuitivního manuálu, s obrazovou a zvukovou odezvou, který bude vždy po ruce. Tedy namířením telefonu na zařízení, které se automaticky zdetekuje, a dotyku na display v místě tlačítka, pro které požadujeme manuál, se požadovaný návod přehraje. Žádaným vedlejším efektem řešení je pozitivní dopad na spotřebu zdrojů a životní prostředí. Dalším z cílů je diskuse nad možnostmi rozšiřitelnosti a využitelnosti aplikace.

Vytvořená aplikace srozumitelnou formou seznamuje uživatele s možnostmi daného zařízení, pro které je tento video manuál určen.

Práce jsem rozdělil na několik kapitol. V kapitole 2 popisují rozšířenou realitu jednak v obecnější rovině tak i trochu podrobněji. V kapitole 3 popisují používané metody detekce markeru. V kapitolách 4 a 5 popisují existující aplikace, kterými jsem se při své tvorbě inspiroval (případně ponaučil) a také dostupné technologie a knihovny vhodné k aplikaci řešeného problému. V kapitole 6 popisují cíle práce a návrh řešení. V kapitole 7 prezentují samotnou implementaci. V samotném závěru práce hodnotím dosažené výsledky a diskutuji nad rozšiřitelností do budoucna (8).

Kapitola 2

Rozšířená realita

V této kapitole vás seznámím s rozšířenou realitou. Nejprve v obecné rovině a poté i podrobněji. Popíši co ona rozšířená realita vlastně je. Poté povím něco o jednotlivých částech rozšířené reality a jejich použití především v mobilních zařízeních. Závěr věnuji historii rozšířené reality a důležitým historickým milníkům od vzniku až po současnost

2.1 Rozšířená realita obecně

Rozšířená realita (RR) je úzce spjata s virtuální realitou. Přebírá mnoho prvků a vlastností, avšak tyto prvky zasazuje do reálného prostředí – reality, ve které žijeme – namísto zasazení do prostředí fiktivního, jak je tomu u reality virtuální. V principu jsme tedy schopni *rozšířit* realitu pomocí speciálního zařízení o prvky virtuální reality. Odtud tedy název *rozšířená realita*. Rozšířená realita je někdy také nazývána smíšenou realitou¹ – mezistupněm mezi úplně reálným a úplně virtuálním světem.

Všeobecně uznávanou definicí RR jsou 3 vlastnosti RR podle Ronalda Azuma [5]:

- RR kombinuje běžnou a virtuální realitu
- RR je interaktivní v reálném čase
- RR je třírozměrná²

Virtuální svět je povětšinou spojen s nějakou hrou, která slouží jako brána do jiného světa. Virtuální svět je pak ovládán buď pomocí standardního vybavení počítače, nebo za použití audiovizuální helmy (HMD – Head Mounted Display), popřípadě speciálního oblečení reagujícího na pohyb, mající hmatovou odezvu. Uživatel je podáván fiktivní svět co nejreálněji, vlivem toho se však může – a také se to děje – uživatel cítit příjemněji ve světě virtuálním. Jedná se však o jakýsi hendikep, kdy virtuální svět uživatele doslova pohltní. Příkladem budiž hra, která dovoluje hráči létat, cestovat časem, aj.

Rozšířená realita se oproti tomu nesnaží uživatele vtáhnout do fiktivního světa. Rozšířená realita udržuje uživatele v realitě, pouze mu nabízí virtuální objekty, které se uživateli jeví jako objekty běžné reality. Často používaný příklad je zobrazení místa, kde uživatel stojí, v minulosti. Můžeme tak „být“ v prostředí, které si jinak můžeme pouze představovat (viz Obr. 2.1).

¹V orig. *Mixed reality*

²V orig. *Registered in 3-D*



Obrázek 2.1: Příklad RR – Historický pohled.

2.2 Rozšířená realita podrobněji

V této části podrobněji vysvětlím aplikaci rozšířené reality. Na diagramu znázorním z jakých komponent se systém rozšířené reality na mobilním zařízení sestavuje. Popíši jak se zpracovávají vstupní data a co je nutné k získání výstupních dat. Závěrem vysvětlím co se myslí prvky rozšířené reality. Část textu této sekce vychází z disertační práce Henryssona Anderse - *Bringing Augmented Reality to Mobile Phones* [12].

2.2.1 Diagram komponent

Diagram komponent je k nalezení na obrázku 2.2, diagram zobrazuje propojení jednotlivých komponent rozšířené reality.

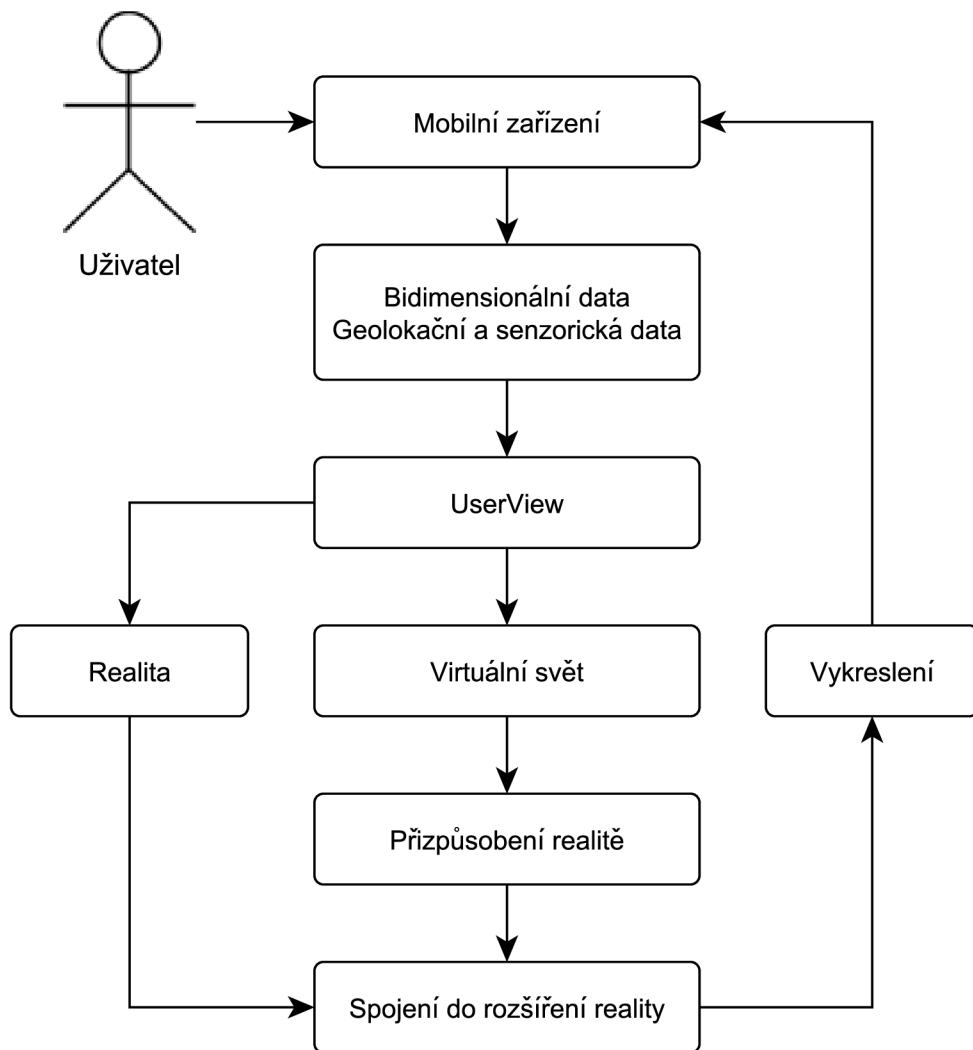
2.2.2 Aplikace rozšířené reality

V sekci 2.1 jsem zmínil 3 vlastnosti RR. Druhou vlastností byla schopnost pracovat v reálném čase. Tuto schopnost popsal ve své práci „Why do users like video?“ John C. Tang [23]. Tuto schopnost definoval počtem snímků za sekundu, přičemž 5 fps považuje za minimum a 30 fps za již velmi kvalitní a hladké vykreslování.

Aplikace RR se skládá ze třech částí (budu používat anglickou terminologii):

1. **Tracking.** Aplikace RR musí nutně znát pozici (viewpoint) uživatele. Pro získání pozice je potřeba znát vstupní data, dle kterých je jednoznačně určena pozice uživatele. Vstupní data mohou být:
 - *Dvourozměrná data (bidimensionální)*, rozumíme jimi nejčastěji vstupní obrazová 2D data³, nad kterými provádíme detekci obrazu s cílem získat 3D souřadnice.

³Vstupní data mohou být také 3D



Obrázek 2.2: Diagram komponent.

- *Geolokační a senzorká data*, skrze tato data je potřeba pouze dopočítat 3D souřadnice.

Pro rozsáhlejší systémy může být platné použití obou zmíněných typů dat.

2. **Registration.** Tracking je ve své podstatě prvním krokem k registration části. Při registraci se snažíme udržovat propojení reálného světa se světem virtuálním. Je tedy potřeba sledovat, jak se mění pozice uživatele, abychom měli stále aktuální pozici pro vykreslení obsahu.
3. **Display.** Pokud máme souřadnice 3D prostoru, do kterého promítáme, je potřeba upravit zobrazovaný model tak, aby byl ve správném měřítku, aby byl správně zrotovaný, aj. Poté provedeme vykreslení (rendering) dané scény. Uživateli tak předkládáme sjednocení virtuálních prvků s realitou – rozšířenou realitu.

2.2.3 Vstupní data a zpracování

Již jsem zmínil rozdělení vstupních dat na dvourozměrná a geolokačně-senzorická. Co tyto pojmy skrývají prozradím nyní. Pro rekapitulaci připomínám, že je možno tyto dva přístupy kombinovat. Také poznamenávám, že oba přístupy mohou využívat obrazový vstup, což však vyplývá z jádra věci, kdy se realita rozšiřuje. Je tedy potřeba realitu „získat“. Stejně tak platí, že vstup, obohacený o virtuální prvky je uživateli vrácen pomocí displeje mobilního zařízení.

Dvourozměrná data

Tento typ dat může být chápán jako 2D vstupní obraz snímáný kamerou mobilního zařízení. Samotné zpracování může mít několik kroků, jak jsou známy z počítačového vidění. Nejprve je nad vstupním obrazem provedeno předzpracování. Velmi často se používá prahování a dalších způsobů, pomocí nichž je obraz upraven tak, aby co nejlépe vystupovaly body zájmu (points of interest). Druhým krokem je potom určení těchto bodů zájmu. Záměrně používám výraz bodů zájmu, mohou to být hrany, rohy, plochy, aj. Záleží totiž vždy na použité metodě v implementovaném systému. Mezi nejčastější používané metody pro detekci těchto bodů patří *SURF*, *MSER*, *FAST*, *BRIEF*. Velmi často se jako vstupní objekt používá marker. O markeru je možné říct, že se jedná o dvourozměrnou matici černých a bílých čtverců – vysoce kontrastních barev. Takové značky jsou snadno a především velmi rychle rozpoznatelné.

Při detekci dvourozměrných dat je tedy naším zájmem jednak zjistit „co“ a druhak „kde“. Potřebujeme jednoznačně určit, zda je na vstupu nějaký objekt, pro který máme virtuální prvky k vykreslení. Pokud detekujeme objekt, musíme také určit jeho přesnou pozici ve scéně. Pozice objektu je středem scény a my se v ní pohybujeme. Potřebujeme znát přesnou polohu, abychom mohli správně pozicovat virtuální prvky a korektně je zobrazit. Příklad aplikace využívající dvourozměrného vstupu je na obrázku 2.3.



Obrázek 2.3: Příklad aplikace používající dvourozměrná data.

Geolokační a Senzorická data

Mezi geolokační a senzorická data řadíme data taková, která se nám snaží podat informaci o prostoru jinak než data dvourozměrná. Konkrétně se potom jedná o data získaná pomocí: GPS, akcelerometru, gyroskopu, kompasu (Solid state compass), RFID (Radio Frequency Identification), bezdrátových zařízení (WiFi, Bluetooth), apod. Dle těchto získaných informací, velmi často se jedná o informaci z více senzorů, určujeme uživatelův pohled (view-point). Dle geolokačních informací potom stanovíme střed scény. Samo zařízení je středem scény, které závisí na naší pozici. Do scény potom vkládáme virtuální objekty. Není zde kladen důraz na maximální přesnost zobrazovaných virtuálních objektů. Často mají tyto objekty uniformní velikost, tj. velikost nezávislou na naší vzdálenosti.

Zajímá nás spíše směr pohledu v 3D prostoru, zda je vykreslovaný objekt scény pro uživatele viditelný. Dle tohoto směru pohledu jsou potom vykresleny objekty virtuálního světa. Příklad takové aplikace je možno nalézt na obrázku 2.4.



Obrázek 2.4: Příklad aplikace používající geolokační a senzorická data.

2.2.4 Prvky rozšířené reality

Dosud jsem psal o zobrazování virtuálních prvků. Bylo by však špatné omezit se pouze na statické 2D, nebo 3D modely předmětů. Realitu můžeme stejně tak dobře obohatit o dynamické obrazové prvky, animace, videa, zvuk. Můžeme také uživateli dovolit manipulaci s objekty ve scéně, ať už na displayi pomocí prstu, nebo například snímáním gest ruky. Nebo také manipulaci futuristickým – čtením myšlenek. Je důležité mít na paměti, že realitu rozšiřujeme, omezení existují pouze realizační.

2.3 Historie RR

Historický pohled vychází z diplomové práce Jonáše Ševčíka [25], ve které popisuje postup při vytváření aplikace použitelné v knihovnictví.

První zmínka

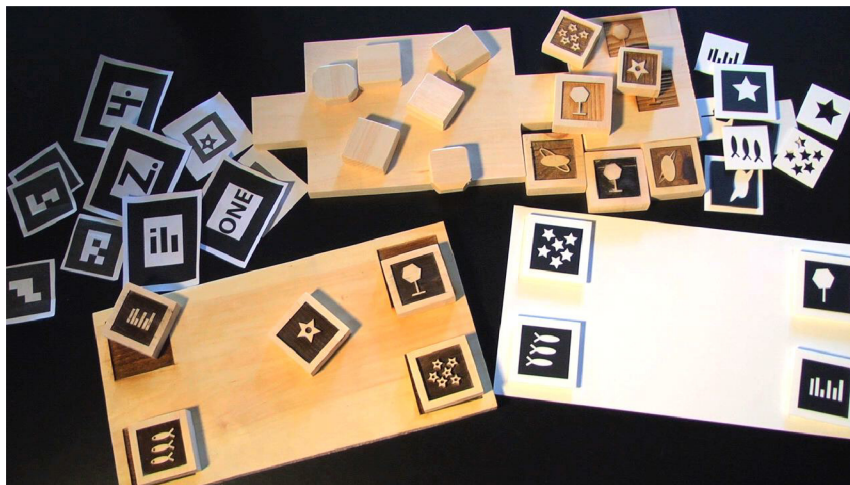
První zmínka o rozšířené realitě pochází z roku 1962, kdy Morton Heilig vytvořil simulátor motorčky. Tento simulátor simuloval zvuky, vibrace i pachy.

HMD

V roce 1965 vynalezl Ivan Sutherland první audiovizuální helmu (head-mounted display), která sloužila jako okno do virtuálního reality.

Čárové kódy

Koncem 60. let 20. století se začaly rozšiřovat čárové kódy, které se ujaly i v rozšířené realitě. Důvodem byly především nízké náklady pro výrobu zařízení snímající tyto kódy. K distribuci čárového kódu potom postačovala i obyčejná tužka. Čárové kódy byly zpravidla ohraničeny tlustým černým rámečkem. Tento způsob byl později využit J. Rekimotem při tvorbě 2D matic – markerů⁴ jak je známe dnes (viz Obr. 2.5) [20].



Obrázek 2.5: Markery jak je známe dnes.

GPS

K dalším významným milníkům patří rozšíření *GPS (Global Positioning System)*, původně určeného pouze pro armádní účely, mezi širší veřejnost. To přineslo nové možnosti použití rozšířené reality. V dnešní době tuto technologii lze nalézt téměř v každém chytrém telefonu (smartphone). Proto není divu, že rozšířená realita, našla velké uplatnění právě v navigaci.

The Touring Machine

V roce 1997 vnikl projekt *The Touring Machine* s nápadem virtuální prohlídky školního kampusu Kalifornské univerzity. Je využito přenosného počítače a poloprůhledného HMD, na který jsou zobrazovány informace o sledovaných objektech a místnostech [10].

AR Quake

O tři roky později (2000) přišel na svět projekt s názvem *AR Quake*. Jednalo se o hru využívající hybridního systému kombinujícího GPS pro určení geolokační pozice a počítačového

⁴Budu používat počeštěný originál, namísto českého překladu – *popisovače*

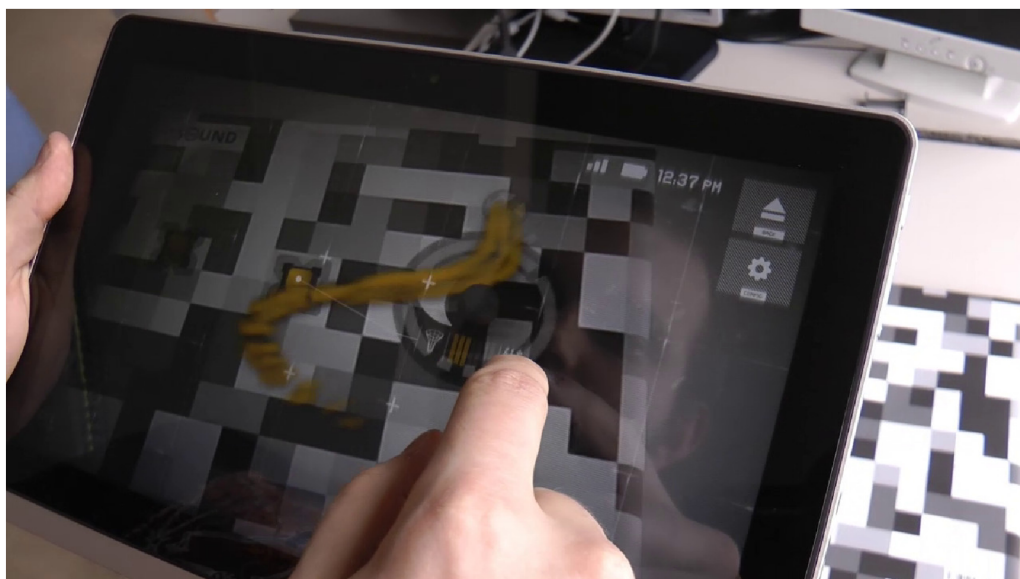
vidění (*CV - computer vision*) k detekci markerů na budovách pro zjištění přesného pohledu hráče k vykreslení virtuálního obsahu. Ačkoli projekt nebyl nikdy dokončen, podnítl spoustu zajímavých nápadů k využití rozšířené reality [24].

PDA a mobilní zařízení

Postupem času se začalo roširovat zařízení značené jako *PDA (Personal Digital Assistant)* mezi širší veřejnost. Bylo tedy logickým důsledkem nahrazení několikakilového zařízení nošeného na zádech, spolu s přilbou na hlavě, o mnoho kompaktnějším PDA. Ve prospěch PDA kromě hmotnosti a rozměrů mluvila také cena. V roce 2004 vznikla aplikace *The Invisible Train* [19], která ukázala obecné principy pro tvorbu rozšířené reality na mobilních zařízeních, platící dodnes [12].

Mobilní zařízení

K masovému rozšíření RR pak došlo s nástupem tabletů, mobilních zařízení, smartphonů⁵ chcete-li, tak jak je známe dnes. Mezi vývojáři postupně vznikaly frameworky a toolkity pracující s rozšířenou realitou. Na obrázku 2.6 můžeme vidět jednu z nejnovějších aplikací - hru *re-Sound*, využívající *Uniform Marker Fields* [22] původem z VUT FIT.



Obrázek 2.6: Rozšířená realita dneška - aplikace re-Sound.

⁵Jako smartphone se dříve označovalo mobilní zařízení s hardwarovou klávesnicí

Kapitola 3

Detekce markeru pro potřeby rozšířené reality.

V této kapitole popisují některé často používané techniky k detekci markeru. Nyní uvedu jeden z možných postupů detekce markeru [13].

3.1 Detekce hran

Nejprve je potřeba vstupní obraz rozdělit na menší regiony, například čtverce se stranou 40 px. Tyto čtverce potom poslouží k urychlení dalších kroků. V následujícím kroku se budou detekovat hrany v každém regionu. K detekci hran je možno použít například *Sobelův gradientní operátor* [4] a to vždy ve směru osy x , tak ve směru osy y . Sobelův operátor má tvar:

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (3.1)$$

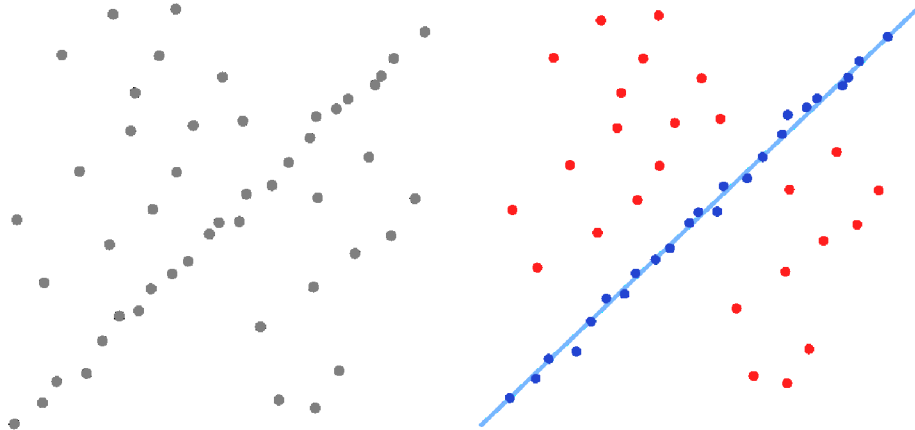
3.1.1 RANSAC

Vlivem šumu a dalších negativních obrazových jevů, je však možno dostat pouze přibližnou informaci o horizontálních a vertikálních hranách v obraze. Je potřeba tyto přibližné informace zpřesnit. Na to se hodí algoritmus *RANSAC* (RANdom SAmple Consensus) [11]. Je to iterativní metoda pro odhad parametrů matematického modelu obecně obsahujícího body, které do modelu patří i nepatří. RANSAC je nástrojem pro vyloučení dat, které do modelu nespadají a naopak určení dat, které do modelu spadají. V našem případě nám tato metoda dokáže vést přímku skrze nejvíce korespondujících bodů v každém regionu. Ilustraci vstupu a výstupu metody RANSAC je možné nalézt na obrázku 3.1. Algoritmus RANSAC má však obecnější charakter, není to jen algoritmus pro hledání přímek.

Zkrácený zápis algoritmu RANSAC pro hledání přímky:

1. Náhodně vyber 2 body ze stejného regionu, jejichž umístění odpovídá hypotetické přímce.
2. Pokud vybrané body splňují hypotézu, jsou přidány do množiny pozitivních nálezů.

3. Opakujeme dokud nenalezneme dostatečný počet bodů přímky (uvažujme alespoň 4).
4. Kroky 1 – 3 potom opakujeme, dokud jsou v daném segmentu nalezené přímky.



Obrázek 3.1: RANSAC: vlevo vstupní data, vpravo model s nalezenou přímkou [2].

3.2 Vyhledání markeru

Ze zjištěných přímek je poté potřeba určit markery. K tomu je však potřeba dalšího předzpracování. Je potřeba propojit odpovídající přímky napříč segmenty. Tyto přímky „prodloužit“ takovým způsobem, aby se přesně protly v rozích, které spolu sdílejí v realitě. Následně odstraníme všechny hrany, které nemají „roh“. Tímto krokem se podařilo detekovat markery v obraze. Není to však úplně pravda, bylo detekováno pouze počítačovým viděním rozeznatelných čtyřúhelníků (Viz obr 3.2). Je tedy potřeba dalšího zpracování. Je vhodné použít některý z algoritmů pro popis klíčových bodů, například často používaný *SURF* [6].

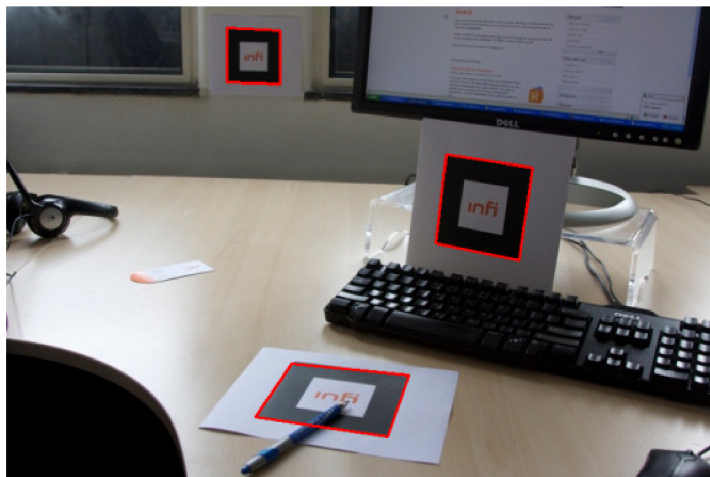
3.2.1 SURF

SURF (Speeded Up Robust Features) [6], je nástroj, pro popis a detekci významných bodů v obraze. Hlavní výhodou této metody je invariantnost k translacím, rotacím, změnám velikosti markeru a také schopnost běžet v reálném čase. Metoda vychází z metody *SIFT - Scale-Invariant Feature Transform*. Metoda používá následující 3 bloky:

- **Integrální obraz.** Jedná se o reprezentaci každého bodu obrazu hodnotou, která je součtem všech hodnot jednotlivých pixelů ve čtyřúhelníku, jež začíná v levo nahoře a jehož diagonálním rohem je právě počítaný bod. Vyjádřeno matematicky:

$$I_{\Sigma}(x) = \sum_{i=0}^{i \leq 0} \sum_{j=0}^{j \leq 0} I(i, j) \quad (3.2)$$

- **Fast-Hessian detektor.** Je založen na Hessově matici (Čtvercová matice druhých parciálních derivací funkce), která se počítá z integrálního obrazu. Z matice jsou



Obrázek 3.2: Ukázka detekovaných markerů [17].

poté určeny minima a maxima, což jsou dané význačné body. Pomocí nejlepší odezvy Haarových vlnek na nejbližším okolí – kružnice s poloměrem $6b$ – zkoumaného bodu b je určena orientace tohoto bodu.

- **Deskriptor významných bodů.** Pro výpočet deskriptoru se vezme čtvercové okolí (region) bodu b se stranou $20b$ a natočí dle orientace bodu b . Dané okolí se rozdělí na 16 menších čtverců (podregionů) o velikosti 5×5 , pro které je určena odezva haarových vlnek (ilustrováno na obrázku 3.3). Vznikne nám tedy 4-dimenzionální vektor:

$$v = (\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|) \quad (3.3)$$

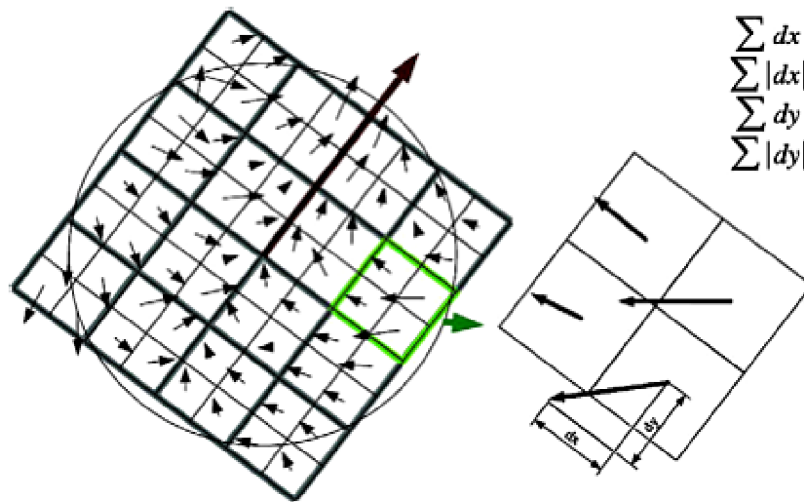
Spojením všech 16 podregionů potom získáme výsledný 64-dimenzionální deskriptor.

Shrnutí Popsal jsem detekci markeru pro potřeby rozšířené reality. Popsal jsem některé významné metody, které jsou klíčové pro detekci markerů. Jedním z cílů detekce je také získání hraničních bodů markeru. Obvykle jsou důležité hranice markeru, které poslouží pro určení projektivní transformace. Dalším z cílů detekce markeru je určení virtuálního obsahu, jež bude vykreslen.

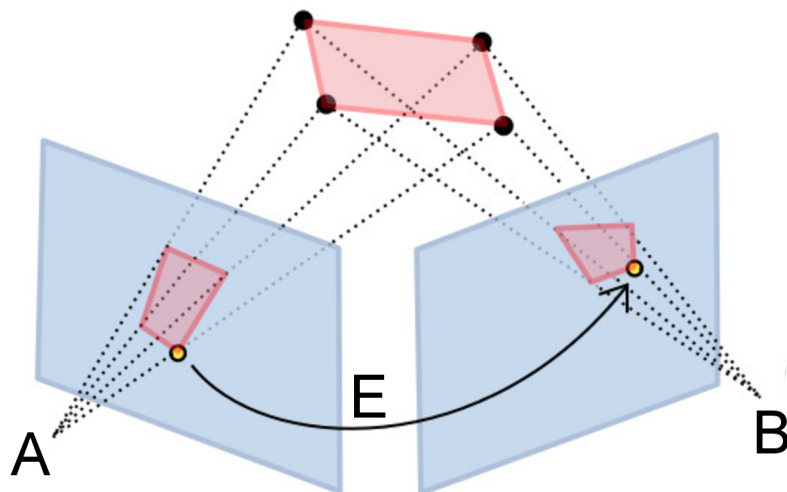
3.3 Homografie

Nutnost homografie pramení v principu 6DOF [18], uživateli je poskytnuta možnost ovládat mobilní zařízení v prostoru, který obecně nekoresponduje s prostorem virtuálního světa. Tuto situaci ilustruje obrázek 3.4. Objekty z virtuálního světa je tedy potřeba transformovat do polohy, které odpovídá světu reálnému.

Homografie (projektivní transformace, projektivita, kolinearita) je relace mezi dvěma objekty (mezi projekcemi objektu, tedy obrázky), kde jeden bod z jednoho objektu je v relaci s bodem z druhého objektu. Homografie je tedy způsob pro popis transformace takové, jejíž aplikací na první objekt získáme objekt druhý. Tato část práce je čerpána



Obrázek 3.3: Tvorba SURF deskriptoru.



Obrázek 3.4: Ukázka různé projekce jednoho objektu [16].

z disertační práce Geoga Kleina[15] a článku *Homografie a epipolární geometrie* Martina Benedy [7] publikovaného v časopise *Trilobit*.

Objekty v 3D prostoru jsou reprezentovány jako homogenní souřadnice $(x \ y \ z \ 1)^T$ přičemž pro označení objektu A se používá vektoru $(x_A \ y_A \ z_A \ 1)^T$. Obdobně potom pro objekt B . Pro transformace se používá transformační matice s rozměry 4×4 značenou E_{BA} pro vyjádření transformace A do B , přičemž je známo, že platí vztah $E_{AB}^{-1} = E_{BA}$. Tedy $B = E_{BA}A$:

$$\begin{bmatrix} x_B \\ y_B \\ z_B \\ 1 \end{bmatrix} = E_{BA} \begin{bmatrix} x_A \\ y_A \\ z_A \\ 1 \end{bmatrix} \quad (3.4)$$

Transformační matice má tvar:

$$E_{BA} = \begin{bmatrix} R & t \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.5)$$

kde R je matice rotace ($|R| = 1, R^T R = 1$) v prostoru a t je vektor posunutí. Lze také říci, že matice E reprezentuje transformaci v \mathbb{R}^3 . Pro výpočet transformační matice je potřeba znát alespoň 4 body ve 2D prostoru. Jasně se tedy nabízí použití rohů detekovaného markeru

$$\begin{bmatrix} w_1 u_1 & w_2 u_2 & w_3 u_3 & w_4 u_4 \\ w_1 v_1 & w_2 v_2 & w_3 v_3 & w_4 v_4 \\ w_1 & w_2 & w_3 & w_4 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \\ y_1 & y_2 & y_3 & y_4 \\ 1 & 1 & 1 & 1 \end{bmatrix} \quad (3.6)$$

kde u_n, v_n, x_n, y_n jsou známé, proto je možné pro u_1 vyjádřit

$$u_1 = \frac{w_1 u_1}{w_1} \quad (3.7)$$

$$= \frac{h_{11} x_1 + h_{12} y_1 + h_{13}}{h_{31} x_1 + h_{32} y_1 + h_{33}} \quad (3.8)$$

přeuspořádáním vzniká

$$h_{11} x_1 + h_{12} y_1 + h_{13} - h_{31} x_1 u_1 - h_{32} y_1 u_1 + h_{33} u_1 = 0 \quad (3.9)$$

podobně pro v_1

$$h_{21} x_1 + h_{22} y_1 + h_{23} - h_{31} x_1 v_1 - h_{32} y_1 v_1 + h_{33} v_1 = 0 \quad (3.10)$$

spojením těchto poznatků vznikne matice 8 lineárních homogenních rovnic 9-ti neznámých, tedy matice C pro kterou platí $Ch = 0$ – řešením je nulový prostor.

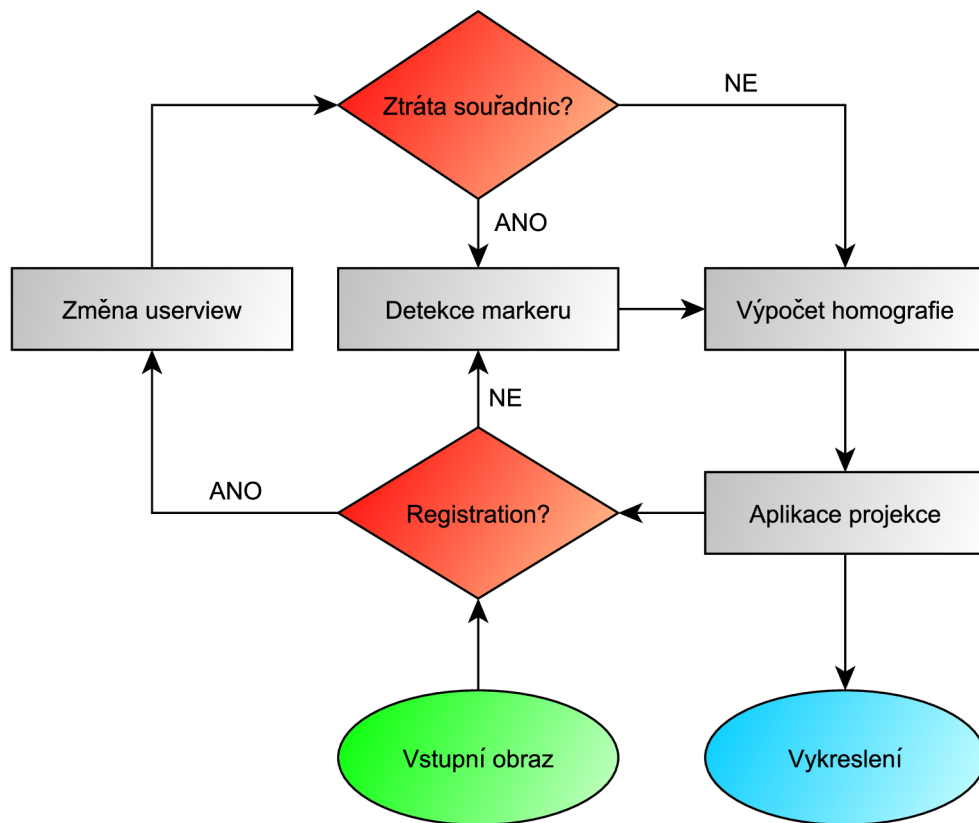
$$\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x_1 u_1 & -y_1 u_1 & -u_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -x_1 v_1 & -y_1 v_1 & -v_1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -x_2 u_2 & -y_2 u_2 & -u_2 \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -x_2 v_2 & -y_2 v_2 & -v_2 \\ x_3 & y_3 & 1 & 0 & 0 & 0 & -x_3 u_3 & -y_3 u_3 & -u_3 \\ 0 & 0 & 0 & x_3 & y_3 & 1 & -x_3 v_3 & -y_3 v_3 & -v_3 \\ x_4 & y_4 & 1 & 0 & 0 & 0 & -x_4 u_4 & -y_4 u_4 & -u_4 \\ 0 & 0 & 0 & x_4 & y_4 & 1 & -x_4 v_4 & -y_4 v_4 & -v_4 \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \\ h_{33} \end{bmatrix} = 0 \quad (3.11)$$

Vypočítáním rovnic se získá homografie h , pomocí níž lze transformovat virtuální svět do světa reálného. Stojí za zmínku, že výpočet homografie je spíše aproximací, než přesným řešením. K řešení je možné využít již zmíněného algoritmu RANSAC.

3.4 Princip použití detekce markeru v rozšířené realitě

Vykreslení rozšířené reality je poslední krok k dosažení výsledku. Již se podařilo detekovat marker, vypočítat projektivní matici. Zbývá tedy aplikovat tuto matici na virtuální svět a výsledkem aplikace potom rozšířit vstupní obrazový proud o projekci virtuálního světa. Občas se ještě před aplikací projekční matice násobí maticí popisující parametry daného zařízení. Tato matice popisuje vztah mezi souřadnicemi v projekci a souřadnicemi pixelů na displayi.

Protože celý systém rozšířené reality musí interagovat s uživatelem v reálném čase, je neustále snímán vstupní obrazový proud. Pro práci v reálném čase však není možné v každém snímku detekovat marker, systém tedy pro zjednodušení sleduje pouze hranice markeru. K tomu se využívá prediktivních technik, jako například *Kalmanova filtru*¹, případně obecnějších částicových filtrů, jako například *Monte Carlo localization*². V každém kroku je pak případně nutné přepočítat projektivní matici, případně ji poupravit dle odezvy již zmíněných filtrů. Celou situaci ilustruje obrázek 3.5.



Obrázek 3.5: Vývojový diagram.

¹Viz http://en.wikipedia.org/wiki/Kalman_filter

²Viz http://en.wikipedia.org/wiki/Monte_Carlo_localization

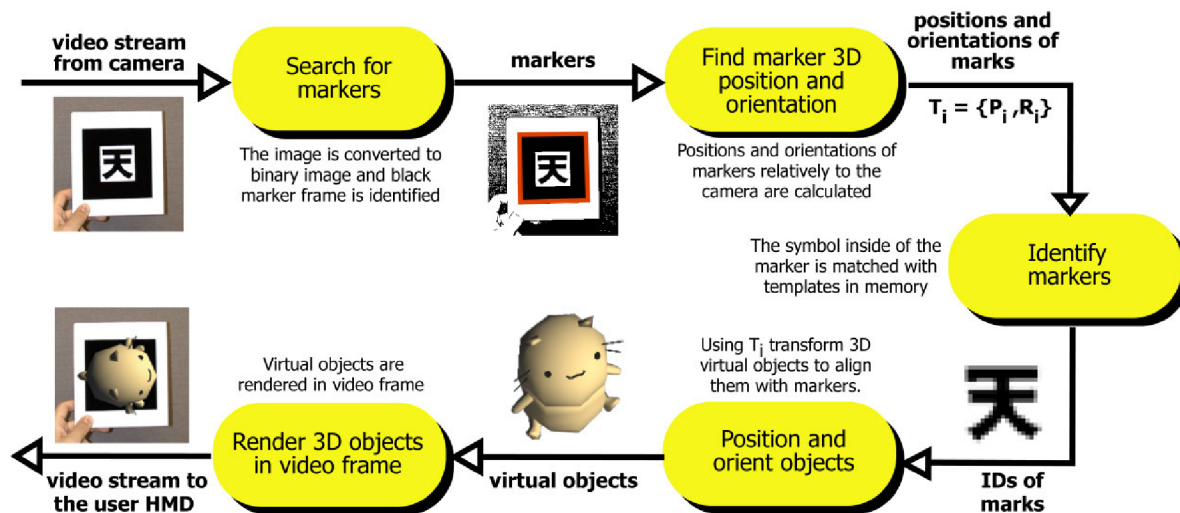
Kapitola 4

Nástroje pro tvorbu RR

V této kapitole čtenáře stručně obeznámím s nástroji pro tvorbu RR *ARToolkit*, přesněji portované verze pro OS android – *AndAR*, a *Vuforia SDK*. Závěrem tyto knihovny srovnám.

4.1 ARToolkit

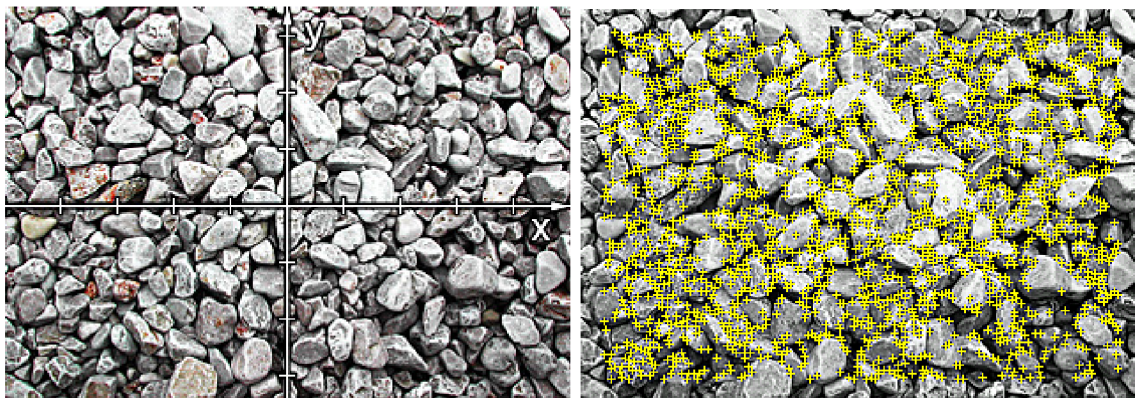
Tato knihovna je první knihovnou svého druhu, vytvořena Hirokazu Kato-em v roce 1999 [14]. Celý toolkit slouží pro všechny tři části aplikace rozšířené reality (Tracking, Registration, Display), jak jsem popsal dříve. Knihovna je založená na dvourozměrných vstupních datech (viz. 2.2.3) při zachování principu 6DOF. Jako značky pro vykreslení je zde využito markeru. Pro jednodušší popis je vše znázorněno na diagramu 4.1.



Obrázek 4.1: Diagram aplikace rozšířené reality pomocí ARToolkitu [21].

4.2 Vuforia SDK

Qualcomm, výrobce procesorů do mobilních zařízení, nabízí řešení s názvem *Vuforia* (Dříve také QCAR – knihovna se tak jmenuje stále), které je optimalizováno pro běh na architektuře *ARM*. Toto řešení vychází z použití libovolného obrázku – nejen markeru – s dostatkem význačných bodů. Dle mých doměnek jsou detekovány rohy v obrázku. Přičemž obrázky, které chceme použít pro vykreslení rozšířené reality, je potřeba pomocí *TargetManageru* nahrát na web vuforie, kde jsou převedeny do interní prezentaci – databázi objektů. Tuto databázi je pak možné stáhnout lokálně a připojit do mobilní aplikace¹, nebo používat online (samozřejmě za cenu zpomalení vlivem přenosu pomocí mobilního internetu)². Na obrázku 4.2 lze vidět ukázkou dobře rozpoznatelného obrázku. Každému obrázku je také nutno definovat rozměry ve virtuálním světě³, tedy velikost scény, kterou bude detekovatelný obrázek zabírat ve světě virtuálním nezávisle na vzdálenosti pozorovatele. Je třeba podotknout, že množství bodů na obrázku nemusí nic znamenat, jelikož jejich pravidelnost může bránit v lokaci přesného pohledu uživatele při částečném překrytí markeru.



Obrázek 4.2: Ukázkou obrázků [3]: vlevo nahrávaný obrázek včetně os, vpravo detekované body – features.

Vuforia nabízí možnost vývoje pomocí nástroje *Unity3D*, které nabízí možnost vytváření zobrazovaných modelů a výsledné aplikace na jednom místě – tato varianta je však placená. Druhou variantou je vytváření modelů pro *OpenGL ES* v grafickém editoru a tvorbu aplikace pomocí *Android SDK* a *Android NDK*.

Zajímavostí řešení od qualcommu je použití virtuálních tlačítek. Funkčnost spočívá v zakrytí části pozorovaného obrázku, což má za následek nedostatek klíčových bodů v daném místě. Pomocí této funkčnosti je možné dále rozšiřovat funkčnost rozšířené reality o nové ovládací prvky.

Mezi další vlastnosti pak patří možnost definovat namísto obrázku krychli, nebo kvádr. Takový objekt je složen ze 6-ti obrázků. Objekt je poté možno použít pro různé rotace objektů virtuálního světa okolo definovaných objektů reálného světa.

Vuforia také nabízí možnost vytváření markerů přímo pomocí aplikace, není tedy potřeba využívat webového rozhraní pro tvorbu markeru.

¹Limit je však 80 obrázků.

²S možností více než 1 000 000 obrázků, avšak toto řešení je zpoplatněno

³Můžeme chápat jako jednotku *em*, jejíž velikost je relativní – udává spíše jakési měřítko

4.3 Srovnání

Obě knihovny jsem testoval se zařízením Motorola RAZR s OS Android ve verzi 4.1.2. Srovnání jsem zaznamenal formou tabulky (4.1):

Úkon	AndAR – ARToolkit	Vuforia
Doba detekce markeru	Až 3 vteřiny	Méně než vteřinu
Detekce částečně překrytého markeru	Nedetekován	Detekován v pořádku
Velký pozorovací úhel	Bez problému	Bez problému
Rychlé pohyby se zařízením	Znatelné zpomalení, ztráta detekovaného markeru	Bez zpomalení, občas ztráta markeru
Rychlost vykreslování	Znatelné zpomalení, při rychlých pohybech	Bez zpomalení
Detekce 5-ti markerů simultánně	Prakticky nepoužitelné	Lehké zpomalení

Tabulka 4.1: Srovnání AndAR a Vuforie.

Z tabulky je zřejmé, že podstatně použitelnější je řešení od Qualcommu. Technologii AndAR jsem díky nutnosti viditelnosti celého markeru, s přihlédnutím k nutnosti použití samotného markeru shledal jako nepoužitelnou. Většímu testování se tedy dostalo Vuforii, kde jsem narazil na další negativa.

Při postupném vzdalování od sledovaného objektu reálného světa se stalo, že objekt virtuálního světa doslova zmizel. Bližším zkoumáním problému jsem narazil na jisté omezení vzdálenosti od sledovaného objektu. Při velikosti snímaného markeru – papíru velikosti A4 – na hranici přibližně 60-ti centimetrů, selhává výpočet orientace osy z , což má za následek promítnutí objektu virtuálního světa do záporných hodnot osy z . Nastává tedy situace kdy je objekt virtuálního světa překryt realitou.

Další nedostatek jsem našel u použití kvádrů namísto obrázku. Vlivem zakřivení obrazu, který je focen fotoaparátem telefonu, není možné vytvořit přesný geometrický útvar jako je obdelník, jelikož zakřivení zkreslí reálný poměr stran objektu. Tento způsob je tedy vhodný pro vytvoření grafiky, která se vytiskne a nalepí na nějakou krabici. Jiným postupem bohužel nelze dosáhnout, byť jen nějakého, výsledku.

Kapitola 5

Existující aplikace

V této kapitole popíši nejznámější aplikace pro mobilní platformu android. Přínos této kapitoly vidím především ve vyzkoušení aplikací a vyvarování se chyb. Stojí za zmínku, že většina aplikací vznikla pro operační systém iOS, čili vývoj pro platformu android není úplně perfektní, jelikož se nejedná o primární cílovou platformu. Ani jedna z aplikací nemá dostupné zdrojové kódy. Všechny aplikace používají své vlastní knihovny a frameworky, která dále nabízejí jako službu, výměnou za finanční prostředky.

Všechny aplikace lze nalézt na obvyklých distribučních kanálech aplikací pro android.

5.1 Aurasma

Tato aplikace využívá dvourozměrných vstupů, pro rozpoznávání stejně jako Vuforia nevyužívá markeru, ale libovolného obrazu. Problémem celé aplikace je rychlost. Vše běží na vzdáleném uložišti. Věškeré detekce probíhají na vzdáleném uložišti. Pro větší rychlost je obraz zmenšen, bohužel je tento obraz také zpětně vykreslován uživateli. Celá aplikace tedy působí velmi nepěkným a nereálným dojmem. Pohyb s telefonem okamžitě způsobí ztrátu spojení virtuálního světa s reálným.

Aplikace však dovoluje vytváření vlastních markerů a přiřazování virtuálního obsahu z databáze. Detekce objektu v tomto případě probíhá bez použití internetu. Avšak uživatel je omezen pouze na již existující databázi objektů.

Celá aplikace na mě působí snahou cílit na co nejširší spektrum uživatelů, bohužel je však celý komplex neoptimalizovaný a působí nepoužitelně.

5.2 Wikitude

Jednou z prvních opravdu použitelných aplikací pro rozšířenou realitu je právě aplikace *Wikitude*. Aplikace využívá geolokační a senzorická data. Slouží k zobrazování míst z online databáze, které jsou něčím zajímavé. Aplikace působí hezky, plynule. Problémem je však ona databáze, kterou vytváří uživatelé. V našich končinách jsou možnosti obsahu velmi omezené. Aplikace tedy skrze málo obsahu nemá valné využití.

5.3 Layar

Aplikace *Layar* je ve své podstatě spojením již jmenovaných aplikací v této kapitole. Používá dvourozměrných vstupních dat pro zobrazování virtuálního obsahu pro partnerské časopisy. Tyto časopisy jsou placené a navíc v našich končinách nedostupné. Aplikace používá také geolokačních a sensorických dat k zobrazení bodů zájmu z databází třetích stran. Existují české databáze jako například *Zlaté stránky*, různé realitní kanceláře, apod. Avšak ani vytváření obsahu třetími stranami nepomohlo ke zkvalitnění obsahu oproti wikipedii a využití je tedy mizivé.

5.4 Shrnutí

Shrnu-li poznatky z testovaných aplikací, můžu říci, že použití dvourozměrných vstupních dat ve spojení s rozpoznáváním na vzdáleném uložišti je v dnešní době opravdu pomalé. Stejně tak nejsou úplně vhodné aplikace, které se snaží zacílit na široké spektrum uživatelů a aplikace vážnoucí na nedostatku obsahu. Z těchto poznatků budu dále vycházet.

Kapitola 6

Návrh aplikace VideoManual

„The best way to predict the future is to invent it“

—Allan Kay, 1971, *meeting of PARC*

V této kapitole objasním nápad mé aplikace *VideoManual*. Popíši návrh aplikace pro mobilní OS Android s použitím Vuforia – knihovny pro rozšířenou realitu. Začátek kapitoly však věnuji motivaci k řešenému problému, která může alespoň nastínit některé informace, které nemusejí být na první pohled zřejmé.

6.1 Motivace

Při testování aplikací zabývajících se rozšířenou realitou, jsem narazil na některé vlastnosti, kterým se chci vyvarovat. Mým cílem je vytvořit aplikaci, která bude skutečně použitelná, uživatelsky přívětivá a bude dělat pouze to, co od ní uživatel očekává. Ukázka toho, jak by mohla aplikace vypadat, je na obrázku 6.1.



Obrázek 6.1: Ilustrační obrázek aplikace.

Na začátku toho však vždy stojí nápad. Jedním z mých¹ nápadů byla aplikace pro intuitivní a interaktivní seznámení uživatele s neznámým zařízením, ke kterému by jinak

¹Inspirovalo mě především „motivační“ video ze zadání této bakalářské práce.

musel číst papírový návod k použití. Možná se ptáte proč nepoužívat papírové manuály. Jsou přece lidé, kteří čtou rádi. Jsem však názoru, že vyhledat informaci o nějakém tlačítku v manuálu prakticky znamená přečíst celou kapitolu popisující ovládání. Proč tedy neudělat manuál jako aplikaci, kde stačí ukázat na tlačítko a ihned víme vše co nás zajímá a nic navíc, co nepotřebujeme?

Další motivací může být například životní prostředí. Stačí si vzpomenout, kolikrát jste četli návod k lednici, kterou máte doma. Většinou manuály přečteme při otevírání krabice se zařízením a pak již jen v případě, že si s něčím opravdu nevíme rady. Po zbytek času je manuál někde schovaný, občas už ani netušíme kde. A když ho náhodou potřebujeme na dovolené, jsme v téměř neřešitelné situaci. Věřím, že knihy jsou informačním kanálem, který lze předávat po generacích. Co však děláme s manuály především je to, že je vyhazujeme. Proč tedy plýtvat lesy na hromady papíru, které v lepším případě přečteme maximálně dvakrát?

Nesmíme zapomenout, že samotný manuál fyzicky zabírá nějaké místo a také něco váží. Cena za manuál samotný, logistiku a expedici se také promítá do výsledné ceny produktu. Nebylo by lepší vymyslet lepší způsob šíření manuálů?

6.2 Cíle práce

- vytvořit demonstrační aplikaci na mobilním telefonu, která by dokázala nahradit papírové manuály,
- vytvořit aplikaci s obrazovou i zvukovou odezvou pro snadnější pochopení manuálů, zapojením více smyslů,
- aplikaci s minimálistickým uživatelským rozhraním, které bude dostatečně intuitivní pro většinu věkových kategorií,
- aplikaci, které nebude třeba markerů k rozpoznávání,
- aplikaci, rozšiřitelnou na velké množství objektů.

6.3 Aplikace

Obecně jsem činnost aplikace popsal již v 2.2.2. První dva kroky aplikace rozšířené reality (tracking a registration) řeší vuforia, na mně zbývá vytvoření sledovaných objektů, tvorba audio manuálů (zvuků), 3D objektu k zobrazení včetně textur a funkčního uživatelského rozhraní.

6.3.1 Sledovaný objekt

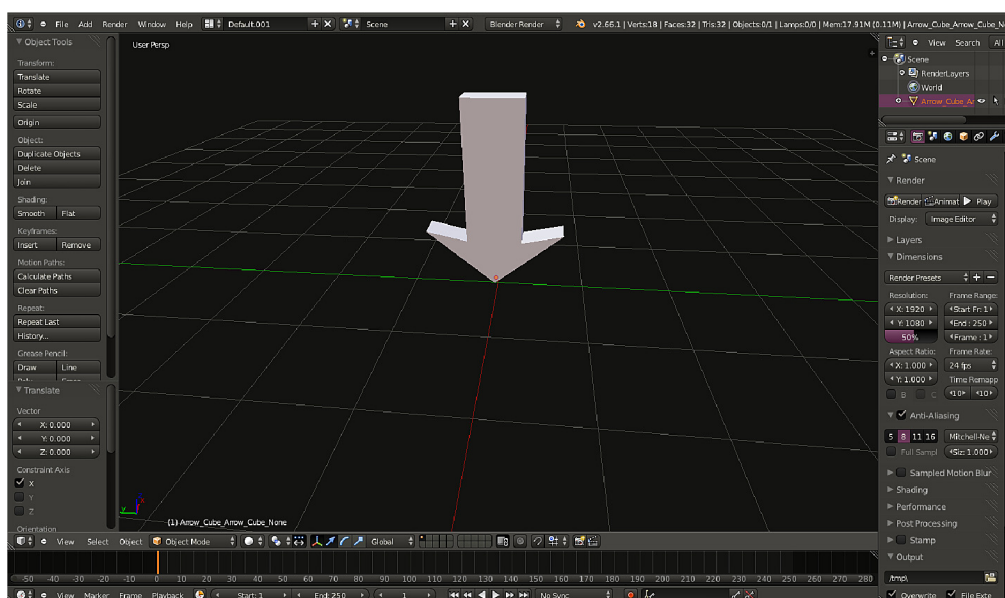
Aplikace jsem navrhl tak, aby zde byla možnost použitelnosti na již existujících zařízeních. Z tohoto důvodu se snažím nepoužít markery pro detekci objektů, ale využít složitosti ovládacích panelů, které využívám k detekci. V tom však výrazně brání konstrukce ovládacích panelů zařízení. Ovládací panely jsou totiž často symetrické a také se snaží působit sladně, nejsou tedy používány příliš kontrastní barvy. Dalším problémem při použití vuforie je nemožnost detekovat kulatá tlačítka. Záměr je tedy vybrat takový objekt, který má dostatek význačných bodů v obraze. Použité obrazové materiály se nachází v příloze B.

6.3.2 Struktura aplikace

Celou strukturu aplikace do jisté míry diktuje Vuforia. Je nutné použít Android SDK (java) pro tvorbu aplikace jako takové, Android NDK² (C/C++) pro tvorbu nativního kódu, který bude k výsledné aplikaci linkován jako knihovna. Práce programátora je vytvořit obě části aplikace, které se propojí do výsledného celku. Pro vytvoření databáze bylo potřeba vytvořit vývojářský účet na stránkách vuforia.

6.3.3 Virtuální prvky

Jedním z cílů je vytvoření aplikace se zvukovou a obrazovou odezvou. Je tedy potřeba vytvořit audio nahrávky pro popis funkčních tlačítek. Je potřeba vybrat člověka jehož hlas bude příjemný a srozumitelný. Dále je potřeba vytvořit model pro OpenGL ES, který bude zobrazován (Viz obr. 6.2) včetně textury, která nebude moc rušivá, ale dostatečně viditelná.



Obrázek 6.2: Šipka – ukázka tvorby modelu v programu Blender [1].

6.3.4 Uživatelské rozhraní

Protože tvořím manuál, je kontraproduktivní vytvářet aplikaci, která by potřebovala také návod k ovládání. Uživatelské rozhraní je v mém případě jen jakési okno do rozšířené reality. Uživatel musí mít možnost zvolit video manuál pomocí stisknutí displeje telefonu v oblasti tlačítka, jehož funkčnost chce znát. Zároveň musí mít uživatel možnost stisknout tlačítko pro znovupřehrání videomanuálu. Spodní část displeje je překryta malým informačním řádkem, který zobrazuje červeným textem informaci o nenalezeném objektu a zeleným textem informaci o konkrétním detekovaném objektu. Ukázka rozhraní je na obrázku 6.3.

²Native Development Kit



Obrázek 6.3: Uživatelské rozhraní aplikace VideoManual

6.4 Návrh testování

V aplikaci mi jde především o intuitivnost a použitelnost. Dvě ne příliš měřitelné veličiny.

Při tvorbě uživatelského rozhraní použiji inkrementálního testování. Pro měření intuitivnosti budu postupně předkládat aplikaci stejným i různým uživatelům. Budu především sledovat, jaké vlastnosti daný uživatel od aplikace bez rozhraní očekává. Dle těchto výsledků budu postupně směřovat další vývoj.

Pro testování použitelnosti povedu diskusi s uživatelem na téma, co mu v aplikaci schází k tomu aby byl ochoten ji v případě potřeby použít a co vše by měla aplikace mít, aby ji bylo možno považovat za plnohodnotný manuál.

Kapitola 7

Implementace

Aplikaci jsem implementoval dle specifikací, které jsem uvedl v kapitole 6. Popisují zde podrobněji některé důležité třídy a strukturu výsledné aplikace.

7.1 Vuforia SDK

Pro implementaci jsem použil SDK vuforia ve verzi 2.0.32, jež bylo aktuálně dostupné na stránkách výrobce [8]. SDK vuforia poskytuje knihovnu QCAR jež poskytuje rozhraní pro jazyk C++, ve kterém byla část aplikace implementována.

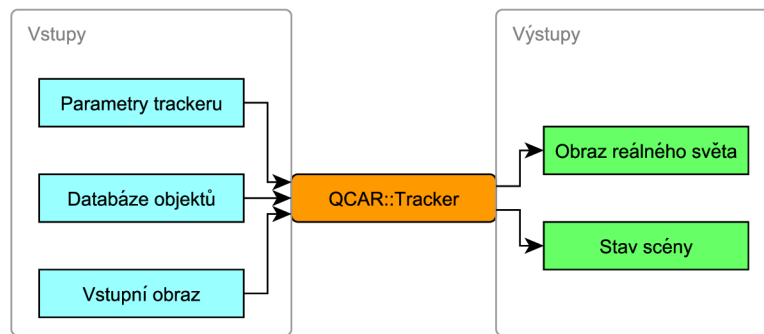
K implementaci jsem také využil některých souborů z volně stažitelných vzorových aplikací. Především pak souborů, které slouží k načtení a inicializaci všech komponent a také soubory pro práci s maticemi apod.

7.1.1 Target database

Pod tímto pojmem se skývá název pro databázi objektů k detekci, uložené ve vnitřní reprezentaci vuforia. V mém případě jsou to soubory `Panels.dat` pro binární reprezentaci databáze a `Panels.xml` pro rozměry virtuální scény pro daný objekt. Pro vytvoření databáze bylo potřeba použít webového rozhraní vuforia, dostupné po přihlášení k vývojářskému účtu a následného nahrání obrázků k rozpoznání, ze kterých je vytvořena targets datáze. Databázi je dále potřeba stáhnout a přibalit do výsledné aplikace do složky `assets`. Použité obrazové materiály se nachází v příloze B.

7.1.2 Tracker

Tato komponenta knihovny QCAR obsahuje veškeré algoritmy počítačového vidění k detekci a sledování objektů reálného světa ve vstupním obrazu. Pro mě je však především důležitý objekt stavu (state object), ze kterého získávám informaci o detekovaných objektech a také jejich umístění ve scéně. Na obrázku 7.1 jsem vyobrazil tuto komponentu. Připomínám, že zde existuje možnost připojení více databází objektů, avšak v jeden moment může být aktivní pouze jedna.



Obrázek 7.1: Vstupy a výstupy komponenty `QCAR::Tracker`

7.2 Struktura aplikace

V této sekci popisují nejdůležitější třídy, potřebné pro chod aplikace. Závěr je věnován rychlému shrnutí tvorby virtuálního obsahu.

7.2.1 Android aplikace

Android SDK tvoří základní strukturu aplikace, která slouží jako transformační rozhraní mezi operačním systémem a nativním kódem. Mimo jiné se stará o propojení nativní části aplikace s OpenGL, správu datového úložiště a přístup k souborům, přehrávání zvuku, poskytnutí informací o doteku na display a samozřejmě také o distribuci obrazu z kamery do nativní části aplikace pro další zpracování. Aplikace byla vytvářena pro OS android ve verzi 4.1.2 (API verze 16).

Základ aplikace jsem rozčlenil do dvou tříd `VideoManual` a `VideoManualRenderer`, přičemž první třída provádí veškerou inicializaci aplikace, knihoven, načtení textur – třída `Texture`, vytvoření uživatelského rozhraní, třídy `SoundManager` aj. Třída `VideoManualRenderer` je přístupovým bodem pro nativní část aplikace a také slouží jako příjemce zpráv z nativní části aplikace. V neposlední řadě je zde třída `QCARSampleGLView` pro komunikaci s *OpenGL ES*, kterou jsem převzal ze vzorových aplikací. Všechny zmíněné třídy lze nalézt v adresáři `src` v souborech stejnojmenných názvů s příponou “.java”.

7.2.2 Nativní část aplikace

V této části aplikace probíhají veškeré inicializace knihovny `QCAR`¹. Nativní část aplikace slouží ke všem třem částem aplikace rozšířené reality, jak bylo popsáno v 2.2.2. Pomocí knihovny získám údaje o vztahu pozorovaného objektu vzhledem ke kameře telefonu – projekční matici a `modelView` matici. Aplikací těchto maticí na objekt poté dosáhnu správného umístění virtuálních prvků ve scéně. Pro vývoj nativní části aplikace bylo využito NDK ve verzi *r8c*.

Tato část aplikace se nachází v adresáři `jni`, pro překlad nativního kódu je třeba v kořenovém adresáři zavolat překlad pomocí terminálového příkazu `ndk-build`. Vznikne nám tak přeložená dynamicky linkovaná knihovna v adresáři `libs`.

¹Vuforia poskytuje knihovnu s názvem `QCAR`

V adresáři `jni` lze nalézt několik zdrojových souborů. Většina z nich jsou soubory, které jsem převzal ze vzorových aplikací. Je zde soubor `VideoManual.cpp`, ve kterém lze nalézt zdrojový kód, který zajišťuje vykreslení a mnoho obslužných rutin pro inicializaci nativního prostředí. Důležitá je však především metoda `renderFrame`, kde je obslužný kód pro vykreslení virtuálního obsahu.

V metodě `renderFrame` jsou zpracovány výstupy z trackeru. Nejprve je vykreslena realita a poté je ověřen objekt stavu, který udává informace o detekovaných položkách. Je třeba ověřit souřadnice dotyku, abychom správně zvolili manuál – tedy umístění a vykreslení objektu (`arrow.h` – modelu pro openGL ES 2.0) ve scéně (na předem určeném absolutním umístění) a také přehrání audio nahrávky pro zvolené tlačítko je-li to vynuceno. Po aplikaci projekční a modelView matice je virtuálním prvkem překreslena realita.

7.2.3 Virtuální obsah

Pro označení přehrávaného manuálu je využito objektu (šipky) ukazující na místo (tlačítko), pro které zobrazujeme video manuál a audio nahrávek. Objekt jsem vytvořil pomocí grafického programu Blender [1] a následně převeden na hlavičkový soubor – model pro openGL ES – pomocí skriptu `mt12opengl` od Ricarda Rendona Cepeda [9].

Kapitola 8

Vyhodnocení výsledků a možnosti rozšíření

V této kapitole hodnotím dosažených výsledků. Na začátku popisuji jednotlivé verze aplikace. Jednotlivé verze potom hodnotím separovaně v podkapitolách o uživatelském rozhraní a použitelnosti. Závěr kapitoly jsem věnoval diskusi nad možnostmi vylepšeními do budoucna.

8.1 Průběh testování

Jak jsem již naznačil v *návrhu testování* (6.4), aplikaci jsem podroboval testování s cílovou skupinou v několika iteracích. A to v následujících pěti:

1. První prototyp aplikace obsahoval databázi jednoho objektu, pro který byl po zdetekování automaticky přehráno video manuál.
2. Do druhé verze jsem přidal možnost znovupřehráni manuálu a animaci vizuálního prvku – šipky.
3. Třetí verzi jsem obohatil o možnost výběru manuálu pomocí dotyku na display telefonu a ochudil o automatické přehráni manuálu.
4. Ve čtvrté verzi jsem aplikaci rozšířil o další objekt, tentokrát reálný objekt. Dále jsem přidal panel se stavem aplikace.
5. Označení „klikatelných“ tlačítek sledovaného objektu očekávajících uživatelskou interakci.

Vzhledem k problémům při testování 4. verze aplikace jsem bohužel nestihl implementovat další verzi. Pátá verze je tedy spíše možností rozšíření do budoucna.

Všechny iterace testování jsem prováděl se skupinou 22 lidí. Každému uživateli jsem sdělil co má aplikace řešit a bez instrukcí jsem uživateli poskytl zařízení s aplikací. Sledoval jsem především intuitivnost, tedy co uživatele napadne udělat jako první. Po vyzkoušení jsem vedl rozhovor o pocitu z celé aplikace a možných vylepšeních do budoucna.

Při testování jsem u starších věkových kategorií narážel na problémy s ovládním . Následnou diskusí jsem zjistil jeden omezující fakt. Lidem, kteří neměli z dřívější zkušenosti

s dotykovými telefony, připadá ovládání zprvu neintuitivní. Avšak upozornění na možnost dotyku vedlo k dalšímu bezproblémovému použití.

8.2 Uživatelské rozhraní

Vyhodnocení testování z hlediska uživatelského rozhraní.

8.2.1 Verze 1

V této verzi žádné uživatelské rozhraní nebylo. Stačilo pouze držet mobilní telefon namířen na sledovaný objekt. Diskuse s uživateli ukázala, že chybí možnost výběru manuálu a také možnost znovupřehrání daného manuálu.

8.2.2 Verze 2

Přidání tlačítka pro znovupřehrání bylo pochopitelné pro všechny, bez výjimky. Animace šipky (pohyb nahoru a dolů) byla přijata kladně. Avšak většina uživatelů tuto změnu spíše nezaregistrovala.

8.2.3 Verze 3

Ve třetí verzi přibylo „neviditelné“ rozhraní. Jelikož telefon již automaticky nepřehrával manuál, uživatelé se snažili vynutit nějakou akci dotykem displaye. Možnost výběru byla přijata kladně a většina uživatelů toto hodnotila jako intuitivní. 6 z 22 uživatelů se o dotyk nepokusilo a zhodnotili aplikaci jako nefunkční. Při rozmluvě jsem hledal důvod nepochopení. Uživatelé si zvykli na automatické přehrávání a nyní jim chybělo. Po sdělení nových možností těchto 6 uživatelů změnilo svůj dojem na pozitivní.

8.2.4 Verze 4

Přidáním panelu informujícím o stavu aplikace se uživatelské rozhraní opět vylepšilo. Ukázalo se však, že chybí označení tlačítek, pro které má aplikace manuál k přehrání.

8.3 Použitelnost

Vyhodnocení testování z hlediska použitelnosti.

8.3.1 Verze 1

Z hlediska použitelnosti uživatelé shledali aplikaci jako nepoužitelnou. V tak ranné fázi aplikace se však nebylo čemu divit. Mým záměrem testování použitelnosti byla především kvalita detekce a plynulého vykreslování rozšířené reality. Celá detekce a sledování objektu s použitím vuforie však fungovala naprosto výtečně, proto nebylo třeba pomýšlet na vlastní implementaci metod.

8.3.2 Verze 2

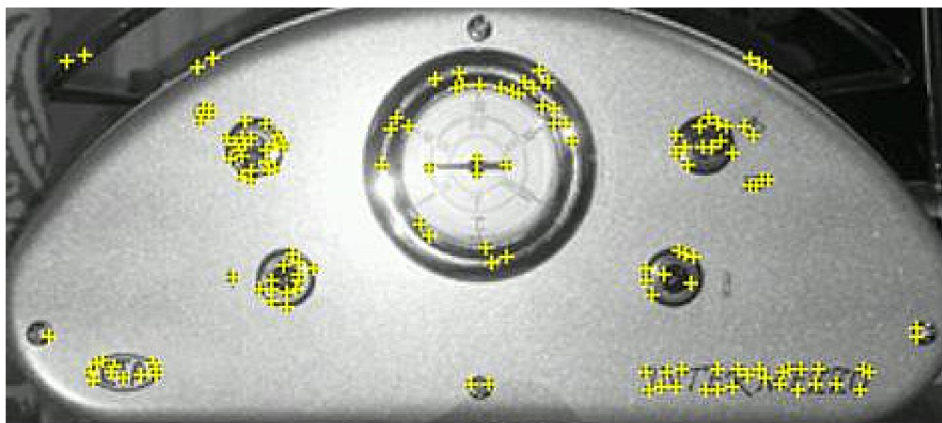
Tlačítko pro znovupřehrání mělo kladný ohlas, avšak použitelnosti to mnoho nepřidalo. Diskuse naznačila, že možnost výběru z více manuálů pro zařízení je dobrým krokem k lepší použitelnosti. Animace se na rychlosti nikterak nepodepsala a vše fungovalo rychlostně beze změn.

8.3.3 Verze 3

Co do použitelnosti byla tato verze velkým krokem kupředu. Uživatelé zhodnotili možnost výběru velmi kladně. 10 z 22 uživatelů aplikaci ohodnotilo jako použitelnou. Zbytek uživatelů vytýkal především použitelnost na jeden objekt.

8.3.4 Verze 4

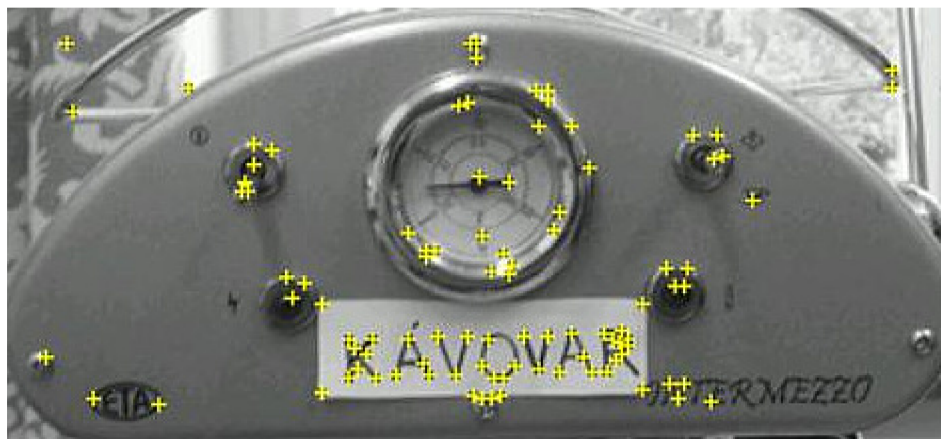
Do této verze jsem přidal druhý objekt – kávovar. Při testování se detekce objektu téměř nedařila, začal jsem tedy hledat důvody. Problémy byly hned dva. Objekt byl vyfocen jako obrázek avšak detekce reálného objektu vyžadovala shodný úhel s úhlem pod kterým byl obrázek focen. Totiž pokud není objekt rovinný, úhlem se mění geometrie a je nerozpoznatelný. Druhým problémem bylo cca 80% detekovaných bodů, které byly na lesklé části objektu a tak reflektovali místnost a člověka detekujícího objekt (Viz obr 8.1). Tuto situaci jsem vyřešil pomocí markeru, který na sebe „přitáhnul“ většinu klíčových bodů a tak již nebylo problémem objekt spolehlivěji detekovat (Viz obr 8.2). Testování opravené verze s přidaným markerem byla zhodnocena 18 uživateli z 22 jako opravdu použitelná. 4 lidé však projevili osobní preference k papírovému manuálu.



Obrázek 8.1: Obrázek ilustrující detekované odlesky.

8.4 Shrnutí výsledků

Cílem této aplikace bylo vytvořit manuál, který by dokázal nahradit papírové manuály. Potvrdily se mé domněnky o existenci skupiny lidí, kteří preferují tištěný manuál, respektive odmítají použití mobilního telefonu k těmto účelům. 18 z 22 uživatelů ohodnotilo aplikaci jako použitelnou.



Obrázek 8.2: Obrázek ilustrující změnu detekovaných bodů po přidání markeru.

Rozhraní aplikace je opravdu jednoduché a tak není divu, že bylo vnímáno jako intuitivní. Věřím však, že dokončení páté iterace by ještě více přispělo k dobré ovladatelnosti.

Vyskytly se problémy při detekci neplanárního objektu a problémy s reflexivitou částí detekovaného objektu. Obě situace se podařilo vyřešit přidáním markeru, avšak urpěl design zařízení.

Aplikaci jsem úspěšně otestoval na čtyřech zařízeních. Zařízení včetně technických specifikací jsou v tabulce č. 8.1. Navzdory odlišnému hardwaru a různým verzím systému, neoficiálních ROM verzí systému Android, však aplikace fungovala totožně na všech zařízeních.

Zařízení	Verze systému	Kamera/CPU/GPU/RAM
Motorola RAZR	Android 4.1.2	8 MPX / Dual-core 1.2 GHz Cortex-A9 / PowerVR SGX540 / 1 GB
Motorola Defy+	CM ¹ 10 (Android 4.2.1)	5 MPX / 1 GHz Cortex-A8 / PowerVR SGX530 / 512 MB
Sony Ericsson Xperia mini	CM 9 (Android 4.0.3)	5 MPX / 1 GHz Scorpion / Adreno 205 / 512 MB
LG Optimus 2X	Android 2.3.6	8 MPX / Dual-core 1 GHz Cortex-A9 / ULP GeForce / 512 MB

Tabulka 8.1: Technické specifikace testovaných zařízení.

8.5 Možná vylepšení

Omezujícím faktorem je použití aplikace pouze na planární objekty. Pokud by dnešní telefony nabízely možnost 3D obrazového vstupu, aplikace by byla jistě použitelnější. Bohužel telefonů s 3D kamerou je nedostatek a trh mobilních telefonů tímto směrem prozatím nejde.

Druhým omezujícím faktem použitelnosti je, že aplikace by měla zvládnout schraňovat mnoho manuálů. Bylo by tedy nutné vytvořit distribuční systém pro manuály od výrobců.

Tedy rozšířit aplikaci o vhodnou reprezentaci video manuálů pomocí další databáze. A vytvořit desktopovou aplikaci pro tvorbu těchto manuálů. S tím také souvisí nutnost zájmu ze strany výrobců zařízení.

Další vylepšení se týká uživatelského rozhraní. Odstraněním automatického přehrávání manuálu klesla intuitivnost aplikace. Jistě by tedy pomohlo po detekci objektu označit místa, která očekávají uživatelskou interakci dotykem.

Rychlost detekce pomocí vuforia je opravdu rychlý, avšak všechny virtuální prvky jsou pozicovány relativně. Jistě by se vyplatilo mít možnost detekovat v rámci vstupního obrazu samostatné tlačítko. Není však zaručeno že by mělo tlačítko dostatek bodů pro přesné pozicování obrazu a tak by bylo pravděpodobně nutné označit samotné tlačítka pomocí markerů. S tím však bohužel souvisí narušení designu zařízení.

Kapitola 9

Závěr

Svou prací jsem se snažil demonstrovat možností rozšířené reality na dnešních mobilních zařízeních s platformou OS Android. Hlavním cílem této práce bylo vytvořit demonstrační aplikaci na mobilním telefonu, která by dokázala nahradit papírové manuály. Za druhotné cíle považuji především použitelnost a intuitivnost navržené aplikace.

Ve své práci jsem zkoumal aplikace a porovnal některé knihovny zabývající se rozšířenou realitou. Výsledky porovnání existujících knihoven ukázaly na použitelnost jednoho z dostupných řešení – vuforia. To mělo za následek maximálního využití dostupných prostředků vuforia, které pracovaly velmi spolehlivě a nebylo tak třeba vytvářet vlastní implementace metod.

Podařilo se mi navrhnout a implementovat aplikaci VideoManual, která je uživateli vnímána jako intuitivní a použitelná. Je však dobré zmínit, že aplikace bojuje se zažitým standardem papírových manuálů, a tak ani inovativní a srozumitelnější seznámení s manuálem zařízení pomocí mé aplikace nemění zaběhnuté stereotypy osob.

Dále jsem splnil hlavní cíl – možnost nahrazení papírových manuálů aplikací na mobilním zařízení. Avšak vždy zde bude omezení dané nutností vlastnit zařízení s OS Android.

Literatura

- [1] Blender. 2013, [online]. [cit. 2013-05-05].
URL <<http://www.blender.org/>>
- [2] RANSAC. 2013, [online]. [cit. 2013-05-05].
URL <<http://en.wikipedia.org/wiki/RANSAC>>
- [3] Vuforia Dev Guide: ImageTargets. 2013, [online]. [cit. 2013-05-05].
URL <<https://developer.vuforia.com/resources/dev-guide/image-targets>>
- [4] Acharya, T.; Ray, A.: *Image Processing: Principles and Applications*. Wiley, 2005, ISBN 9780471745785, 135–136 s.
URL <http://books.google.cz/books?id=smBw4-xvfrIC>
- [5] AZUMA, R. T.: A Survey of Augmented Reality. *Presence: Teleoperators and Virtual Environments 6*, 1997: s. 355–385, [online]. [cit. 2013-04-04].
URL <<http://www.ronaldazuma.com/papers/ARpresence.pdf>>
- [6] Bay, H.; Ess, A.; Tuytelaars, T.; aj.: Speeded-Up Robust Features (SURF). *Computer Vision and Image Understanding 110*, 2008: s. 346–359, ISSN 1077-3142, [online]. [cit. 2013-04-20].
URL <http://www.sciencedirect.com/science/article/pii/S1077314207001555>
- [7] Beneda, M.: Homografie a epipolární geometrie. In *Trilobit - elektronický odborný vědecký časopis*, 2010, ISSN 1804-1795, [online]. [cit. 2013-04-27].
URL <<http://trilobit.fai.utb.cz/homografie-a-epipolarni-geometrie>>
- [8] Center, Q. A. R.: Android – Vuforia SDK. 2013, [online]. [cit. 2013-05-05].
URL <<https://developer.vuforia.com/resources/sdk/android>>
- [9] Cepeda, R. R.: mtl2opengl. 2013, [online]. [cit. 2013-05-05].
URL <<http://www.rendoncepeda.com/home/projects/mtl2opengl/>>
- [10] FEINER, S.; MACINTYRE, B.; HOLLERER, T.; aj.: A touring machine: prototyping 3D mobile augmented reality systems for exploring the urban environment. In *Wearable Computers, 1997. Digest of Papers., First International Symposium on*, 1997, s. 74–81, doi:10.1109/ISWC.1997.629922, [online]. [cit. 2013-04-08].
URL <<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=629922>>
- [11] Fischler, M. A.; Bolles, R. C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography.

- Communications of the ACM* 24, 1981: s. 381–395, ISSN 0001-0782, [online]. [cit. 2013-04-20].
 URL <<http://doi.acm.org/10.1145/358669.358692>>
- [12] HENRYSSON, A.: *Bringing Augmented Reality to Mobile Phones*. Linköping studies in science and technology. dissertations, Linköping University Linköping University, Visual Information Technology and Applications (VITA), The Institute of Technology, 2007, [online]. [cit. 2013-04-08].
 URL <<http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-10204>>
- [13] HIRZER, M.: *Marker Detection for Augmented Reality Applications*. Technická zpráva, Institution for Computer Graphics and Vision, Graz University of Technology, Austria, 2008, [online]. [cit. 2013-04-25].
 URL <http://studierstube.icg.tugraz.at/thesis/marker_detection.pdf>
- [14] Kato, H.; Billinghurst, M.: Marker Tracking and HMD Calibration for a Video-based Augmented Reality Conferencing System. In *Proceedings of the 2nd International Workshop on Augmented Reality (IWAR 99)*, San Francisco, USA, oct 1999, [online]. [cit. 2013-04-29].
 URL <<http://www.hitl.washington.edu/artoolkit/Papers/IWAR99.kato.pdf>>
- [15] Klein, G.: *Visual Tracking for Augmented Reality*. Dizertační práce, University of Cambridge, 2006, [online]. [cit. 2013-04-27].
 URL <<http://www.robots.ox.ac.uk/~gk/publications/Klein2006Thesis.pdf>>
- [16] Moulon, P.: openMVG: open Multiple View Geometry. 2013, [online]. [cit. 2013-05-05].
 URL <<http://imagine.enpc.fr/~moulonp/openMVG/coreFeatures.html>>
- [17] den Otter, C.: Marker Detection for Augmented Reality Applications. 2010, [online]. [cit. 2013-05-05].
 URL <<http://infi.nl/blog/view/id/56/>>
- [18] Paul, R.: *Robot manipulators: mathematics, programming, and control : the computer control of robot manipulators*. MIT Press series in artificial intelligence, MIT Press, 1981, ISBN 9780262160827, [cit. 2013-04-20].
 URL <http://books.google.cz/books?id=m99TAAAAMAAJ>
- [19] PINTARIC, T.; WAGNER, D.; LEDERMANN, F.; aj.: Towards Massively Multi-User Augmented Reality on Handheld Devices. In *Proceedings of the 3rd International Conference on Pervasive Computing (PERVASIVE 2005)*, Volume 3468: Lecture Notes in Computer Science / Springer, 2005, ISBN 3-540-26008-0, s. 208–219, [online]. [cit. 2013-04-08].
 URL <http://www.ims.tuwien.ac.at/media/documents/publications/wagner05_pervasive.pdf>
- [20] REKIMOTO, J.: Matrix: a realtime object identification and registration method for augmented reality. In *Computer Human Interaction, 1998. Proceedings. 3rd Asia Pacific*, 1998, s. 63–68, doi:10.1109/APCHI.1998.704151, [online]. [cit. 2013-04-08].
 URL <<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=704151>>

- [21] Sinclair, P.: ARToolkit. 2001, [online]. [cit. 2013-05-05].
URL <<http://www.metade.org/research/phd/artoolkit/>>
- [22] SZENTANDRASI, I.; ZACHARIAS, M.; HAVEL, J.; aj.: Uniform Marker Fields: Camera localization by orientable De Bruijn tori. In *Mixed and Augmented Reality (ISMAR), 2012 IEEE International Symposium on*, 2012, s. 319–320, doi:10.1109/ISMAR.2012.6402593, [online]. [cit. 2013-04-08].
URL <<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6402593>>
- [23] Tang, J. C.: *Why Do Users Like Video? Studies of Multimedia-Supported Collaboration*. Technická zpráva, Mountain View, CA, USA, 1992, [online]. [cit. 2013-04-20].
URL <<http://www.izix.com/pubs/Tang-WhyLikeVideo.pdf>>
- [24] THOMAS, B.; CLOSE, B.; DONOGHUE, J.; aj.: ARQuake: an outdoor/indoor augmented reality first person application. In *Wearable Computers, The Fourth International Symposium on*, 2000, s. 139–146, doi:10.1109/ISWC.2000.888480, [online]. [cit. 2013-04-08].
URL <<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=888480>>
- [25] ŠEVČÍK, J.: *Rozšířená realita na mobilní platformě Android a její aplikace v knihovnictví*. Diplomová práce, Masarykova univerzita, Fakulta informatiky, 2011, [online]. [cit. 2013-04-07].
URL <http://is.muni.cz/th/255493/fi_m/>

Příloha A

Obsah CD

Příložené CD obsahuje:

- PDF verzi této práce,
- \LaTeX verzi této práce,
- zdrojové kódy/projekt android aplikace,
- přeloženou aplikaci pro OS android,
- použitou verzi vuforia SDK,
- sadu testovacích obrázků,
- soubor `README.txt` s popisem zprovoznění prostředí,
- plakát prezentující tuto práci.

Příloha B

Vzorové obrázky objektů



Obrázek B.1: Ovládací panel šaliny.



Obrázek B.2: Ovládací panel kávovaru s markerem.



Obrázek B.3: Ovládací panel kávovaru bez markeru.

Příloha C

Plakát



Obrázek C.1: Ukázka plakátu.