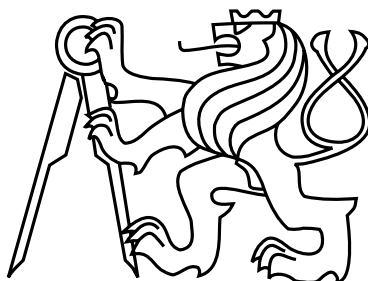


České vysoké učení technické v Praze
Fakulta elektrotechnická
Katedra počítačové grafiky a interakce



Diplomová práce

Interaktivní vizualizace zvukového výstupu

Bc. Ladislav Týč

Vedoucí práce: Ing. Roman Berka, Ph.D.

Studijní program: Otevřená informatika

Obor: Počítačová grafika a interakce

7. května 2015

Poděkování

Děkuji týmu pracovníků z Institutu intermédií za jejich ochotu a pomoc v průběhu testování instalace celého projektu. Také bych chtěl vyjádřit vděčnost za poskytnutí prostor a místního vybavení, bez čehož by dokončení práce nebylo možné.

Zvláště pak děkuji mému vedoucímu práce, panu Ing. Romanu Berkovi, Ph.D., který mi umožnil projekt realizovat dle mých představ a poskytoval mi konzultace při výběru konečných řešení. Díky němu jsem se mohl ve své závěrečné práci dostatečně realizovat a současně vytvořit projekt, který zasahuje do mnoha oblastí mých zájmů.

Prohlášení

Prohlašuji, že jsem práci vypracoval samostatně a použil jsem pouze podklady uvedené v příloženém seznamu.

Nemám závažný důvod proti užití tohoto školního díla ve smyslu §60 Zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon).

V Praze dne 6. 5. 2015

.....

Abstract

This thesis delves into the problematique of creation and deployment of a multimedia application, which combines techniques from the fields of computer graphics, computer vision and audio processing. The goal is to establish an interactive scene, which deepens the musical experience of the viewing subjects by employing a visual component in the form of audio visualization. The main focus of the projection is to create an increased spatial immersion, an illusion of a wider virtual scene, which the subject then feels to be part of.

This effect is achieved by the means of projection mapping. The visual component is projected onto the walls of the room deformed in such a way, that the subject is led to perceive the scene to be far larger than the actual physical space he is situated in. The immersion factor of this effect is achieved by a dual camera setup, which accurately tracks the observer's position. This allows the projected image to be distorted in a certain way as to evoke an illusion of depth when viewed by the observer as he freely moves around.

Abstrakt

Práce popisuje problematiku tvorby a nasazení multimediální aplikace, ve které jsou kombinovány přístupy z oboru počítačové grafiky, počítačového vidění a zpracování zvukového signálu. Cílem je vytvoření interaktivní instalace, která prohlubuje divákův zážitek z poslechu hudby doplněním o obrazovou složku ve formě vizualizace přehrávaného zvukového signálu. V rámci projekce této vizualizace je pak kladen důraz na vyvolání dojmu prostorovosti – na vytvoření iluze, že je divák součástí rozsáhlejší virtuální scény.

Tohoto efektu je docíleno pomocí techniky *Projection mapping*. Promítáme na stěny obraz zdeformovaný tak, že při pohledu z určitého místa dochází vlivem perspektivního zkreslení k optické iluzi zvětšení prostoru nad jeho fyzické rozměry. Pro celkovou uvěřitelnost výsledného efektu pak využíváme sledování pozice pozorovatele v prostoru dvojicí kamer, díky čemuž můžeme obraz deformovat v závislosti na jeho umístění a virtuální scéna při pohybu působí opravdově.

Obsah

1	Úvod	1
2	Možnosti vizualizace hudby	3
2.1	Hudba v kontextu vizualizace	3
2.2	Techniky vizualizace	4
2.3	Uzpůsobení projekce	10
3	Zpracování zvukového signálu	11
3.1	Hlasitost zvukového signálu	11
3.2	Frekvenční charakteristika zvukového signálu	12
3.3	Detekce rytmu v živé hudbě	14
4	Vizualizace a její projekce	19
4.1	Dostupné prostory a použitá technika	19
4.2	Projection mapping v Institutu Intermédií	19
4.3	Realizace aplikace po stránce grafického výstupu	21
4.4	Návrh scén pro projection mapping	23
5	Přizpůsobování projekce divákovi	27
5.1	Výběr techniky zjišťování polohy	27
5.2	Vyhledávání pozorovatele v obrazu kamery	28
5.3	Kalibrace optického systému	31
5.4	Poloha pozorovatele v souřadnicích scény	33
6	Realizovaná audiovizualizační aplikace	35
6.1	Možnosti aplikace	35
6.2	Rozbor implementace	36
6.3	Testování a hodnocení výstupu	38
6.4	Diskuse k výstupu práce	43
7	Závěr	45
A	Popis ukázek projekce z příloženého disku	51
B	Obsah příloženého DVD	57

Seznam obrázků

2.1	Šíření vzruchu způsobeného výstřelem (znázornění změn tlaku ve vzduchu) . .	5
2.2	Využití Rubenovy trubice ke znázornění frekvence přehrávaného tónu	5
2.3	Ukázka dvou variant statické hudební vizualizace	6
2.4	Morphyre - stereo zobrazení vizualizace	7
2.5	Frequency Domain - hudbou generovaný terén ve wireframe režimu	7
2.6	Cyclique - audiovizuální instalace pro SIGNAL festival 2014	8
2.7	Projection mapping na 40m vysoké nádrži v Amsterdamu [2]	9
2.8	Závislost iluze prostorovosti na pozici pozorovatele [6]	10
3.1	Úprava hlasitosti dle lidského vnímání	13
3.2	Citlivost lidského sluchu na různé slyšitelné frekvence [4]	14
3.3	Znázornění průběhu tónu v akustickém signálu [5]	16
4.1	Hlavní hala Institutu intermédií v průběhu nastavování projektorů	20
4.2	Rozdílné podání barev bočního a předního projektoru (bráno z jedné fotografie)	21
4.3	Zabarvení černých oblastí obrazu vlivem odrazu světla	21
4.4	Deformace pohledových jehlanů podle umístění pozorovatele	22
4.5	Porovnání projekcí s různým zkreslením perspektivy	23
4.6	První vizualizační scéna	24
4.7	Druhá vizualizační scéna	25
4.8	Třetí vizualizační scéna	25
4.9	Čtvrtá vizualizační scéna	26
4.10	Pátá vizualizační scéna	26
5.1	Spojené kamery pro stereo vidění (stereo rig)	28
5.2	Model kamery používané pro sledování diváka v prostoru	29
5.3	Provizorní marker využívaný k lokalizaci uživatele	29
5.4	Dílčí mezivýsledky filtrace podle <i>Hue</i> , <i>Saturation</i> , <i>Value</i>	30
5.5	Znázornění zachycení scény dvěma kamerami [3]	31
5.6	Demonstrace radiálního zkreslení a odpovídající vyrovnání obrazu [7]	32
5.7	Průběh kalibrace vnějších parametrů stereo systému	33
5.8	Znázornění soustavy souřadnic propojující reálnou scénu s virtuálními	34
6.1	Nabídka konfigurace systému pro sledování pozorovatele	35
6.2	Neformální diagram rozdělení hlavních tříd dle jejich funkcí	37
6.3	Kamery používané pro rekonstrukci polohy pozorovatele	39

6.4	Rozklad signálu do sedmi frekvenčních pásem	40
6.5	Kresba využívající úběžných bodů	41
6.6	Znázornění navazujících linií na rozhraní projekčních stěn	42
6.7	Návrh scény vhodné pro ohodnocení iluze hloubky	43

Kapitola 1

Úvod

Motivace

Výpočetní technika již prakticky od svého vzniku neslouží jen ke zpracování věcných informací, ale nachází uplatnění prakticky ve všech oborech a zasahuje do každodenního lidského života na mnoha úrovních. Jedním z nejrozšířenějších takových využití jsou počítačové hry. Ty však zdaleka nejsou jedinou možností, jak lze počítačů využít k vytváření unikátních zážitků. Neustále se hledají nové alternativy jejich využití na poli interaktivních aplikací, k prohloubení imerze v kombinaci s dalšími ději v reálném světě a vytváření zajímavých multimediálních instalací. Jednou z takových je i hudební vizualizace.

Bývá běžně udáváno, že se obrazové vjemy oproti ostatním podílejí na člověkem získávaných informacích z více jak třech čtvrtin. Dokonce se často objevuje i ne příliš podložené tvrzení, že tvoří až 90% potřebných informací a že sluch přispívá jen zlomkovým podílem. V každém případě je ale také dobře známo, jak akustická složka přispívá k vyvolání zamýšleného dojmu a odpovídající emocionální reakce v posluchačích.

Z pohledu snahy vyvolat v „uživateli“ (hráči videoher, divákovi sledujícím film atp.) hlubší psychickou odezvu je zvuk prakticky nepostradatelnou součástí prezentovaného celku. S pomocí akustické složky dokreslíme zamýšlenou atmosféru v mnohem kompletnější a intimnější podobě. Speciální úlohu pak má hudba [19]. Díky jejímu úzkému napojení na emoční ladění člověka poskytuje asi nejsnazší cestu k náladám diváka. Toho se hojně využívá v celém zábavním a zážitkovém průmyslu, kdy snad každá vizuální prezentace musí být hudbou v nějaké formě doplněna.

Sluch však zdaleka nemusí být využíván jen v doplňkové formě vůči ostatním smyslům. Hudba obecně může stát přímo v centru pozornosti bez jakýchkoliv jiných stimulů a přinášet posluchači obohacující zážitek sama o sobě. Nicméně obdobným způsobem, jakým využíváme hudbu k doplnění jiných dějů, můžeme i hudbu, jakožto hlavní prvek, rozšířit například o obrazovou složku. Tím je možné celý hudební zážitek vnímat ve více rovinách, a tak jej celý dále umocnit. Za tímto účelem nejčastěji hudební vizualizace vznikají.

Cíle práce

V rámci této práce se budu zabývat vybranými možnostmi, kterými lze v reálném čase řešit vizualizaci aktuálně přehrávané hudby. Velký důraz bude kladen na formu prezentace. Mým cílem je využít některou z technik moderní počítačové grafiky pro vyvolání dojmu prostorovosti zobrazovaného výstupu, aby měl divák pocit, že se přímo nachází ve virtuální vizualizační scéně.

Reálným produktem práce bude aplikace zpracovávající akustický signál z výstupu nějakého zařízení (mikrofon, digitální piano, hudební přehrávač a podobně), která bude připravena a vyzkoušena v Institutu intermédií [1]. Vzhledem k výše popisovaným požadavkům na výstup bude aplikace uzpůsobena na projekci pomocí třech projektorů, aby byl divák projekcí obklopen, jak nejlépe je to v prostorách IIM možné. Zároveň bude aplikace po instalaci v IIM schopna určovat polohu pozorovatele a podle ní bude celá projekce upravována. To by mělo dopomoci k vyvolání pocitu přítomnosti ve větším prostoru, než jaký je v IIM fyzicky k dispozici. Aplikace by proto měla poskytovat nějaké možnosti pro uzpůsobení zobrazování a dodatečnou kalibraci výsledné instalace.

Celý projekt by se dal rozdělit na tři části podle hlavních disciplín, kterých se dotýká. První z nich je problematika zpracování zvukového signálu s ohledem na možnost jeho vizualizace. Druhou by mohla být otázka výsledné prezentace, tedy různé techniky použitelné pro zobrazování generovaného obrazu. Třetí rozsáhlou problematikou je pak dodatečná interakce s vizualizací, v mém případě primárně sledování pozice pozorovatele v návaznosti na použitou techniku projekce. Podle těchto témat bude také celá práce strukturována.

V závěrečné části bych chtěl představit realizovanou aplikaci a nějaké experimenty, kterými by bylo možné ohodnotit její kvality. Také zde nastíním možnosti, které představené techniky otevírají. Protože projekt kombinuje tři odlišné obory v jeden celek, nabízí se otázka, jak jinak by se dalo využít třeba jen některých částí a jaké aplikace by bylo možné vytvořit nad tímto, již implementovaným základem.

Kapitola 2

Možnosti vizualizace hudby

Všechny dílčí úlohy vytyčeného projektu představují vlastní sadu komplikací s různými přístupy řešení. Zde bych se proto chtěl dotknout těch nejkřivějších otázek a problémů, které je třeba při návrhu podobné aplikace řešit.

2.1 Hudba v kontextu vizualizace

Pro vizualizaci hudby s využitím počítačové techniky můžeme rozlišit dva hlavní přístupy podle toho, kdy máme data dostupná ke zpracování. Často totiž vizualizujeme hudbu, která je uložena na paměťovém médiu. V takovém případě můžeme analyzovat nahrávku dopředu a získávat data pro vizualizaci před tím, než dojde k vygenerování samotného zvuku. Druhou možností je pak zpracovávání zvukového signálu v reálném čase, kdy předem neznáme žádné informace o jeho budoucím průběhu. V pozdějších částech textu se budu primárně zabývat právě druhou možností. Přináší sice oproti první mnohé komplikace, na druhou stranu ji ale lze využít interaktivnějším způsobem, a proto je z mého pohledu využitelnější, zajímavější.

2.1.1 Vizualizační parametry

Při vizualizaci vždy hledáme parametry daného vstupu, které bychom mohli mapovat na objekty v prostoru scény. U hudby rozlišujeme mnoho takových parametrů. Z pohledu časového (tedy *horizontálního*) průběhu hudby sem můžeme zařadit například metrum, na něho nava-
zující rytmus, tempo, styl přednesu (artikulace – tedy staccato, legato, tenuto) a dynamiku a její průběh. Z pohledu relativní výšky tónů (*vertikálního* průběhu hudby) pak melodii, harmonii, tóninu. Na vyšší úrovni můžeme zohledňovat například citové zabarvení [32]. Všechny tyto vlastnosti by bylo vhodné využívat v rámci vizualizace, ale je prakticky nemožné získat je z hudebního záznamu pomocí algoritmického zpracování. Lze jich však dobře využívat při předpřipravené vizualizaci nebo za asistence člověka, který parametry nějakým způsobem ručně určuje.

Pokud bychom realizovali vizualizaci pracující nad nějakým standardizovaným vstupem, např. využívajícím protokolu MIDI, měli bychom přesnější informace o dané skladbě a dokázali bychom odhadnout mnoho ze zmíněných parametrů. Pokud ale pracujeme na vstupu pouze se zvukovým signálem, je situace mnohem složitější a spolehlivě získáme jen hrubou

představu o analyzované hudbě. Typickým parametrem, se kterým můžeme pracovat, je intenzita signálu v danou chvíli, a tedy aktuální dynamika, případně její změny v čase a zpětný průběh. S využitím technik zpracování signálu pak můžeme získat zastoupení různých složek spektra, tedy orientační výšku tónu. Tóny reálných nástrojů jsou však složeny z mnoha harmonických frekvencí a v hudbě zpravidla hraje mnoho tónů a zvuků v jednu chvíli. Toto frekvenční zastoupení je tedy přínosné a pro vizualizaci využívané, zároveň je však informační hodnota čistě orientační a prakticky nelze zrekonstruovat hrající melodii nebo oddělit od sebe jednotlivé instrumenty. Obdobným problémem je tempo a rytmus – existují algoritmy pro odhad [28], avšak nejsou obecně dostatečně přesné pro libovolný hudební žánr. Zároveň se těžce vypořádávají s problémem „60BPM – 120BPM“.

2.1.2 Problémy vizualizace v reálném čase

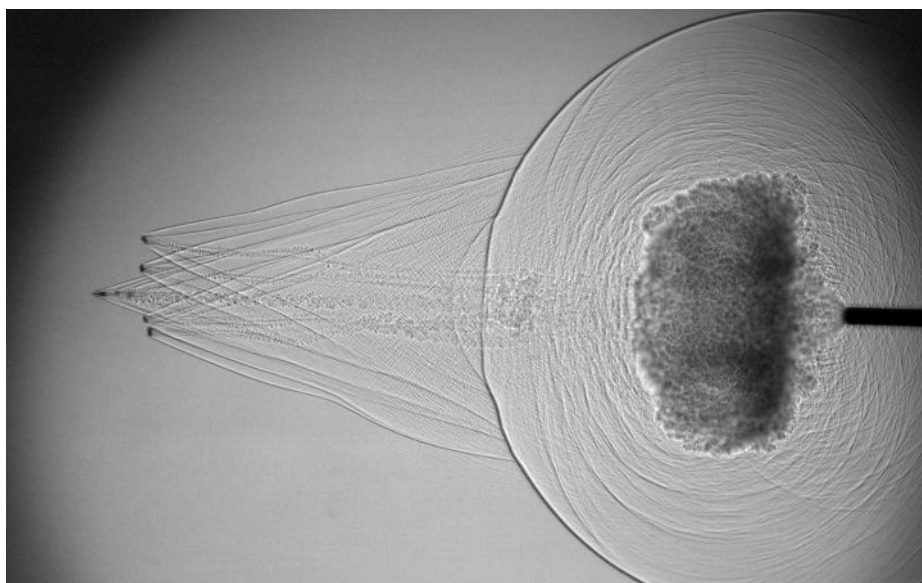
Ze zvoleného řešení, tedy vizualizace v reálném čase, plyne jeden zásadní problém, a sice, že zvuk na vstupu můžeme zpracovávat až ve chvíli, kdy je slyšet. Veškeré informace o zobrazeném zvuku tedy máme dostupné až zpětně. Lidský mozek nemá velký problém spojovat situace, kdy vizuální podnět předchází akustickému, protože je tak od přírody zvyklý. Zvuk se v běžném prostředí šíří pomaleji než světlo, a tak není nepřirozené vidět míč dopadnout a až pak jeho dopad slyšet. Takový sled událostí snadno a automaticky spojíme dohromady. Není pro nás ale přirozené, aby zvuk předcházel obrazu. V takovém případě je tolerance rozdílu času řádově menší. Proto se musí veškeré detekované změny projevit na vizualizaci okamžitě. V případě vizualizace již dostupné nahrávky bychom mohli takové informace získávat před samotným zvukovým projevem, a tak mohli využívat nějaký „vizuální náběh“ třeba půl vteřiny předem, když víme, že bude následovat nějaké razantní zesílení atp. Při vizualizaci v reálném čase to však není možné, ne se zachováním přirozeného působení tak, aby nedošlo ke „smyslovému roztržení“ informací.

2.2 Techniky vizualizace

Zvukový signál lze zpracovávat i zobrazovat různými způsoby a jednoduchou vizualizaci můžeme realizovat i bez použití počítačové techniky. Zvuk je ve své fyzikální podstatě chvěním probíhajícím v látkovém prostředí - pro člověka typicky ve vzduchu. S tím jsou pak spojeny změny tlaku, které vnímáme sluchovým ústrojím a které můžeme měřit a zaznamenávat. Na základě této znalosti lze sestavit různé pokusy či zařízení, kterými lze některé vlastnosti tohoto chvění zobrazovat. V nejjednodušší podobě stačí pozorovat přímo vibrace na zdroji zvuku, či s použitím nějaké zobrazovací techniky přímo chvění šířící se prostředím. To nám umožní například technika *Schlieren flow visualization* [33], poskytující výsledky jako na obr. 2.1.

Další zajímavou, fyzikálně založenou technikou, umožňující vizualizaci zvuku, je *Rubénova trubice*. Jedná se o princip využívající odrazů a interference zvuku v uzavřeném prostoru a následného vzniku stojatých vln [11]. Ty způsobují změny tlaku a v důsledku toho rozdílný průtok otvory, kterými uniká zapálený plyn (obr. 2.2). Podobných technik je celá řada, včetně známého osciloskopu pro znázornění časového průběhu zvukového signálu.

Vizualizace hudby je nejčastěji tvořena za účelem jejího obohacení a rozšíření zážitku z poslechu. Není to však jediný možný přístup. Existují i statické vizualizace znázorňující



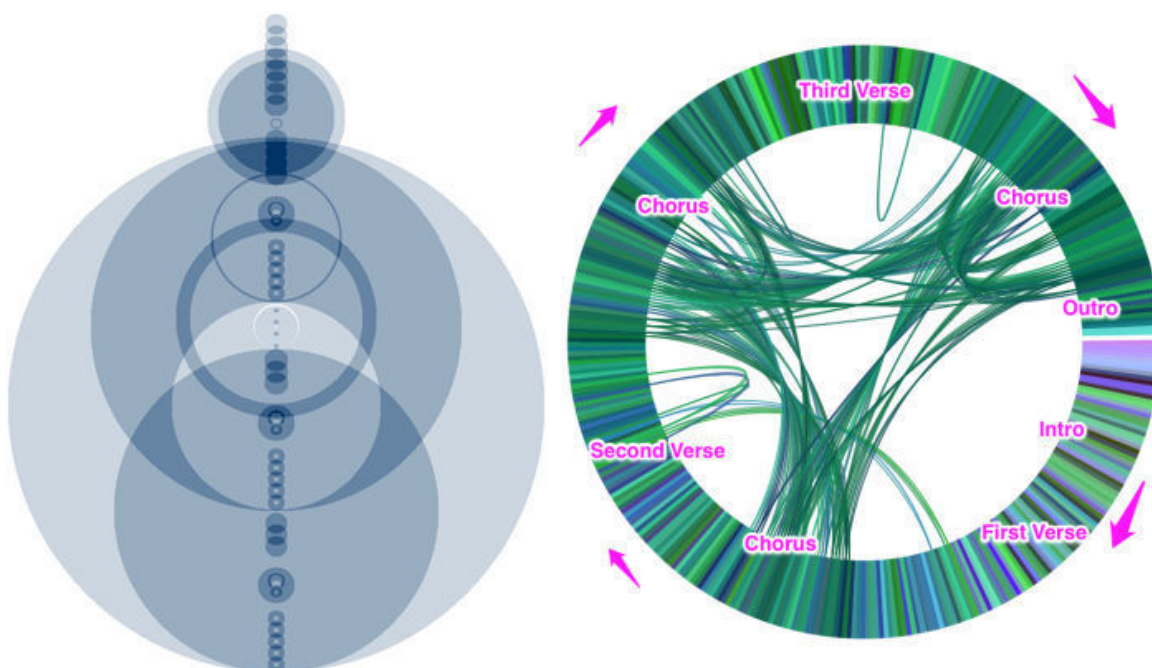
Obrázek 2.1: Šíření vzruchu způsobeného výstřelem (znázornění změn tlaku ve vzduchu)



Obrázek 2.2: Využití Rubenovy trubice ke znázornění frekvence přehrávaného tónu

některé rysy hudebních záznamů, jako na obr. 2.3. Možným příkladem je vizualizace skladeb s využitím kruhových diagramů znázorňujících jejich strukturu a opakující se části [36]. Obdobná technika je také rozebírána v pracích [20] a [30].

Zdaleka nejrozšířenější využití vizualizace hudby je v rámci hudebních přehrávačů, případně aplikacích realizovaných odděleně, avšak zamýšlených primárně jako jejich obrazový



Obrázek 2.3: Ukázka dvou variant statické hudební vizualizace

doprovod. Bývají to standardně vykreslované 2D či 3D scény, je však i pár projektů využívající technik stereoskopického zobrazení, virtuální reality a podobně. Zpravidla jsou parametry zvukového signálu mapovány na abstraktní objekty, fraktály či částicové systémy. Ty reagují změnou svojí polohy, natočení, velikosti, barvy nebo i tvaru. V rámci některých scén jsou využívány komplexnější 3D objekty, které pak mohou být animovány událostmi v hudbě – například postavy s jednoduchou animací tance, která je přehrávána podle hlasitosti a rytmu hudby. Méně často se pak lze setkat s mapováním hudby nejen na zobrazovaný obsah, ale i na řízení běhu samotného zobrazování. Příkladem může být změna globální rychlosti přehrávání animací, tedy ohýbání času v dané virtuální scéně.

2.2.1 Ukázky možných řešení

Aplikací pro vizualizaci hudby existuje poměrně velké množství. Převážně se jedná jen o klasickou počítačem generovanou grafiku zobrazovanou na standardní obrazový výstup, tedy čistě o dvourozměrné prezentace. Zde uvádím některá řešení, která prezentovanou vizualizaci dále nějak obohacují a prezentují výsledek v prostorovější podobě.

Morphyre

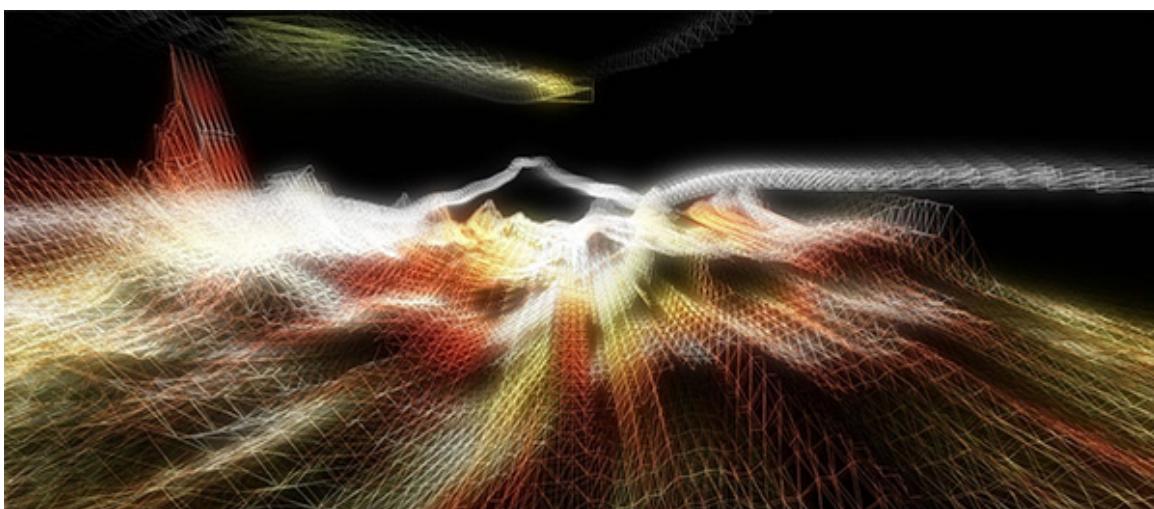
Jedná se o celkem standardní vizualizační software, který ale na rozdíl od ostatních nabízí prohlížení obsahu ve 3D (obr. 2.4). Jedná se o dnes již dobře známou a dostupnou techniku stereoskopického zobrazení [29]. Kromě toho nabízí stovky scén pro vizualizaci a do určité míry umožňuje i vytváření a prohlížení vlastních scén. Kromě toho však neumožňuje žádnou speciální interakci, jen základní nastavení jasu, barev a přiblížení kamery.



Obrázek 2.4: Morphyre - scéna mapující zvuk na chování částicového systému v side-by-side 3D zobrazení

Frequency Domain

Jde spíše o experimentální projekt, který je ještě stále ve vývoji [35]. Ačkoliv může vystupovat i samostatně, hlavní využití najde v kombinaci se zařízením typu HMD (*head mounted display*) pro virtuální realitu – Oculus Rift [24]. Divák se díky němu cítí, jako by byl přítomný v dané scéně. Může se díky vestavěným sensorům rozhlížet do různých směrů a tím také částečně s vizualizací interagovat. Samotná vizualizace probíhá jako průlet uživatele nad procedurálně generovaným terénem reprezentujícím změny v přehrávané hudbě (obr. 2.5). Uživatel je tímto světem unášen a jeho trajektorie je ovlivňována jak směrem jeho pohledu, tak i tvarem generovaného terénu.



Obrázek 2.5: Frequency Domain - hudbou generovaný terén ve wireframe režimu

Cyclique

Oproti předchozím projektům se nejedná o standardní počítačovou aplikaci, ale o rozsáhlejší uměleckou instalaci, která přehrávanou hudbu doprovází světelnými efekty na reálných objektech rozmístěnými na větší ploše [10]. Konkrétně jde o instalaci vytvořenou pro SIGNAL festival 2014, kdy je v rytmech hudby osvětlováno 256 balónků pomocí v nich instalovaných svítivých diod (obr. 2.6). To v kombinaci s jejich pohybem vyvolaným poryvy větru dělá celou multimediální prezentaci unikátní a zajímavou. Podobných instalací existuje mnoho, Cyclique uvádím jako jednoho ze zástupců pro demonstraci toho, jak lze kombinovat efekty generované počítačem s reálnými objekty.

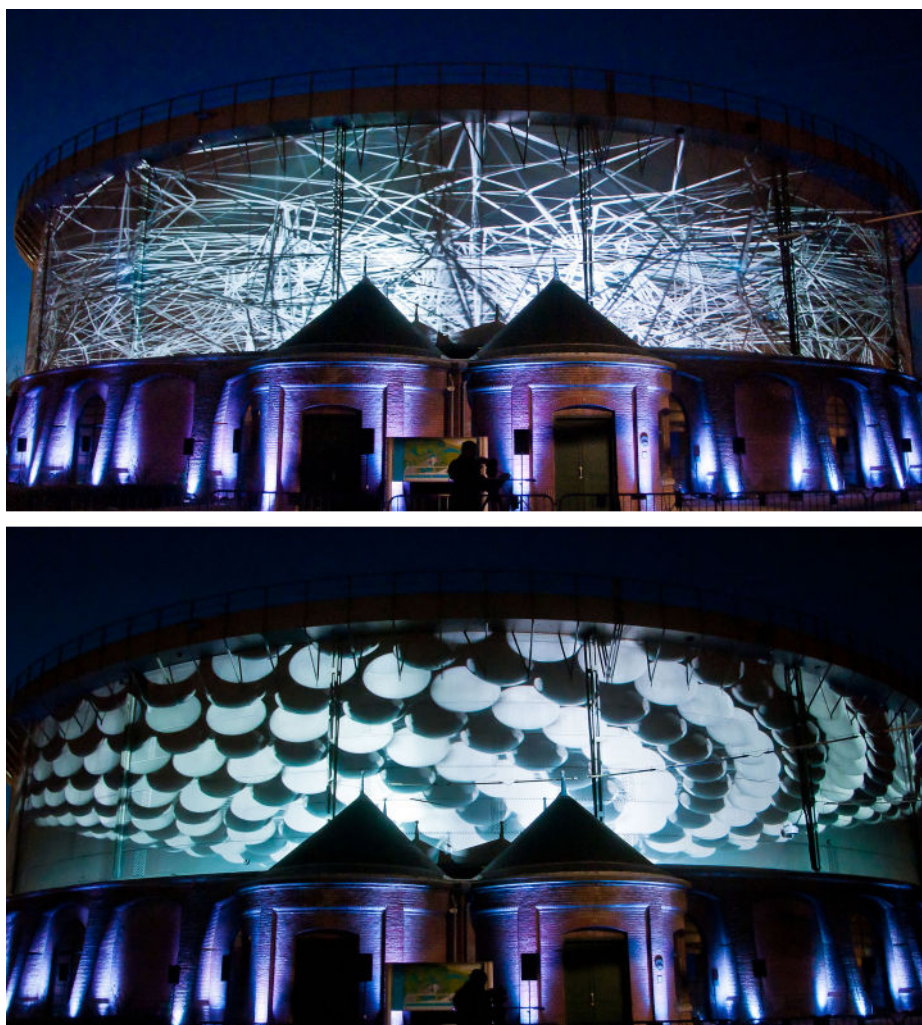


Obrázek 2.6: Cyclique - audiovizuální instalace pro SIGNAL festival 2014

2.2.2 Zvolené řešení

Techniky stereo zobrazování jsem si už osvojil na mých dřívějších projektech, stejně tak jsem již v minulosti zkoušel vytvořit aplikaci pro virtuální realitu využívající HMD. Obě tyto techniky mohou být překvapivě působivé. Osobně jsem však chtěl v rámci mého závěrečného projektu vyzkoušet některou z mnou dříve neprobádaných metod. Při vyhledávání vhodného řešení jsem několikrát narazil na techniku projekce dnes často označovanou jako *3D mapping*, někdy též *Videomapping* či *Projection mapping* (obr. 2.7). Tyto pojmy nejsou příliš jednoznačně vymezeny a někdy se využívají s mírně odlišným významem.

Hlavní podmínkou pro úspěšné nasazení projection mappingu je správná práce s perspektivou. Perspektivního zobrazení se zde využívá k vyvolání iluze objektů vystupujících z plochy, na které projekce probíhá, k vytvoření dojmu ohýbání fyzických struktur a podobným efektům. Tuto techniku lze použít při projekci na prakticky libovolný fyzický objekt,



Obrázek 2.7: Projection mapping na 40m vysoké nádrži v Amsterdamu [2]

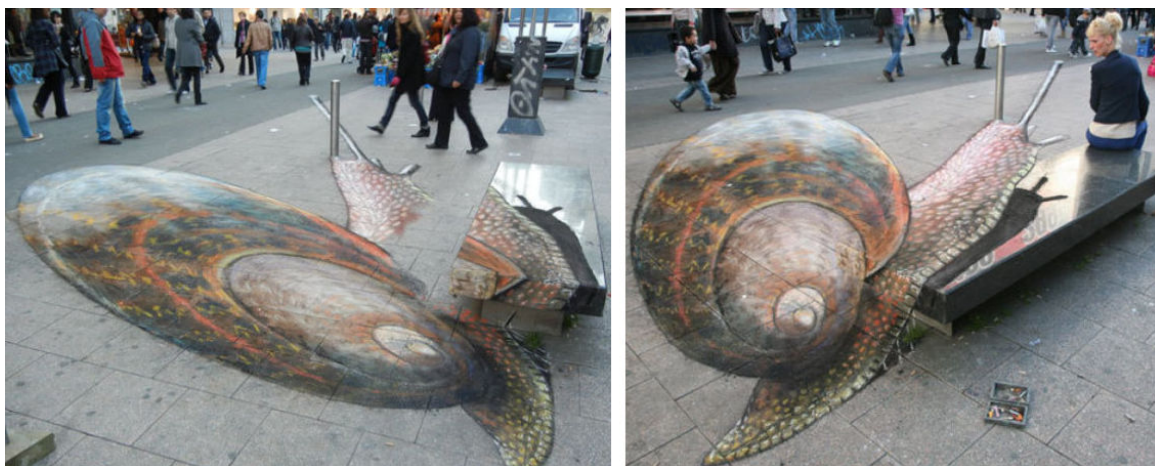
nejčastěji je však využíváno velkých povrchů jako jsou budovy a jiná architektonická díla. Velká plocha je nejvhodnější právě z důvodu uvěřitelnosti výsledného efektu. Člověk vnímá hloubku na základě mnoha indikátorů, jako je perspektivní zkreslení, ale také relativní pozice objektů, paralaxního efektu, vlastnosti binokulárního vidění a mnoho dalších. Právě binokulární vidění je důvodem, proč jsou nejvhodnější vzdálené a velké plochy.

Abychom dosáhli uvěřitelné prostorové iluze, je třeba co nejvíce eliminovat faktor vnímání hloubky na základě informací o aktuálním „nastavení“ očí. Vnímání hloubky u zdravého člověka je totiž úzce propojené s fyziologickými rysy oka. Zejména jde o akomodaci čočky v oku, vzájemné natočení (konvergenci) očí a jejich relativní posun (a s ním spojená míra paralaxního efektu). Čočka v lidském oku mění své vlastnosti při pohledu na objekty vzdálené až přibližně 6–8 metrů, což by tedy měla být minimální vzdálenost projekce, abychom mohli tento faktor zanedbat. Obdobně je tomu i s konvergencí. Ta se nemění přibližně od stejné vzdálenosti a pohledový vektor obou očí je pak téměř rovnoběžný. Posledním důležitým faktorem je to, že každé oko vidí trochu jiný obraz vlivem jejich vzájemného posunu. Tato

informace se také ve vnímání hloubky silně projevuje. Vzdálenost očí proto musí být vůči vzdálenosti projekce zanedbatelná. Pokud pracujeme s menší vzdáleností než zmíněných 8 metrů, musíme počítat s tím, že se výsledný efekt v plné míře u diváka nemůže dostavit.

2.3 Uzpůsobení projekce

Zvolená technika, projection mapping, má jedno kritické omezení. Jelikož je tak závislá na perspektivním zkreslení promítaného obrazu, je nutné takovou projekci sledovat z přesně daného místa. Při pohledu na scénu o několik stupňů odlišném se efekt hloubky vytratí, obdobně jako je tomu na obr. 2.8. S rostoucí vzdáleností projekce se zvětšuje plocha pro možné umístění diváků, na které lze výsledného efektu dosáhnout. I tak je ale potřeba nacházet se v určité vzdálenosti od místa, na které je projekce kalibrovaná.



Obrázek 2.8: Závislost iluze prostorovosti na pozici pozorovatele [6]

Na menším prostoru toto omezení kazí výsledný dojem natolik, že je třeba ho nějakým způsobem adresovat. V mém projektu proto řeším i tuto problematiku. Sleduji pohyb diváka v prostoru a celou projekci uzpůsobuji tak, aby odpovídala místu, ze kterého scénu pozoruje. Samozřejmě však není dost dobře možné ve stejné chvíli realizovat projekci pro více diváků na různých místech v prostoru, výsledná perspektiva vždy odpovídá jen pozici jednoho z nich. Teoreticky by bylo možné projekci řešit pro dva diváky a využít některé z technik pro stereoskopické zobrazování (aktivní brýle se synchronizací projektorů či polarizace projekce), pro realizovaný projekt je to však nepraktické řešení, mělo by smysl snad jen pro použití v aplikaci, kde by spolu uživatelé nějak dále interagovali – hra kombinovaná s virtuální realitou a podobně.

Ve výsledku jsem se rozhodl pro řešení určování pozice pozorovatele s využitím dvojice kamer, pomocí nichž lze zrekonstruovat informaci o tom, jak je třeba přizpůsobit perspektivu projekce. Protože však v prostorách IIM není dost dobře možné realizovat pevnou instalaci, musela být do aplikace implementována i řada funkcí sloužících pro celkovou kalibraci a konfiguraci využitého stereo systému.

Kapitola 3

Zpracování zvukového signálu

V rámci realizované aplikace využívám pro přístup k informacím o zvuku na vstupu počítače knihovny PortAudio [31]. Jedná se o volně přístupné, multiplatformní řešení, díky kterému získávám průběžně pole vzorků zvukového signálu. Ten může v určitý okamžik nabývat libovolné hodnoty mezi minimem a maximem. Pro další popis budu předpokládat, že vzorky nabývají hodnot z rozsahu -1 až 1. Při běžně používané vzorkovací frekvenci vstupního signálu je takovýchto hodnot za jednu vteřinu 44100. Nad tímto signálem pak můžeme provádět vlastní analýzu hledající v daném signálu námi požadované rysy.

Jak jsem zmiňoval dříve, je prakticky nemožné tento signál rozdělit nějakým přesným způsobem a spolehlivě získávat možné parametry pro následnou vizualizaci. Vytvořená aplikace proto pracuje se dvěma orientačními rysy. Prvním je zastoupení rozdílných frekvencí v analyzovaném signálu, tedy přibližný podíl různě vysokých tónů. Druhým pak přítomnost úderů (resp. přízvukných dob, anglicky *beat*) v daném časovém okamžiku. Mimo to je pak ještě výsledná vizualizace řízena na základě celkové aktuální hlasitosti. Aby byly dobře znatelné změny v projevu v hlasitých i v tichých pasážích, upravuji citlivost vizualizačních scén podle průměru hlasitosti za několik posledních vteřin.

3.1 Hlasitost zvukového signálu

Pokud máme k dispozici jen základní údaje o zvukovém signálu v podobě hodnot z vymezeného oboru, nemáme jak zjistit, jakou hlasitost daný signál reprezentuje v reálném světě. K tomu bychom potřebovali znát údaje o konfiguraci techniky, která signál zaznamenala a která jej přehrává. Pro účely vizualizace si však vystačíme i s hlasitostí relativní. Tu můžeme vztahovat buďto k maximální hlasitosti, jaké by mohl signál na daném oboru dosahovat, nebo k intenzitě signálu ve vymezeném předchozím časovém úseku.

3.1.1 Výpočet aktuální intenzity

Relativní hlasitost zvuku/hudby v daném okamžiku lze spočítat jako efektivní hodnotu určitého úseku signálu podělenou jeho možným maximem. Pokud pracujeme se signálem o vzorkovací frekvenci 44100 a chtěli bychom znát orientační hlasitost 30 krát za vteřinu, mohli

bychom analyzovat vždy úseky o 1470 vzorcích. Hlasitost jednoho takového úseku zjistíme výpočtem kvadratického průměru nad jeho vzorky, tedy jako druhou odmocninu aritmetického průměru jejich druhých mocnin. V anglické literatuře bývá efektivní hodnota označována jako *RMS* [16].

$$I_{rms} = \sqrt{\frac{1}{n} * \sum_{i=1}^n x_i^2} \quad (3.1)$$

Pokud pracujeme se vzorky na normalizovaném oboru hodnot (-1 až 1), můžeme na vypočtenou hodnotu rovnou nahlížet jako na procentuální hlasitost vůči maximu.

3.1.2 Vnímání hlasitosti zvuku člověkem

Z výše uvedeného výpočtu víme, jakou relativní intenzitu má zvukový signál, který aktuálně zpracováváme. Získané číselné hodnoty však vyjadřují vztahy z pohledu techniky, která se zvukem pracuje. Pokud je naším cílem lépe postihnout působení intenzity signálu na diváka/posluchače, musíme vzít v úvahu také to, jak je hlasitost vnímána na jeho straně.

Hlasitost hudby je u člověka subjektivní a neodpovídá přesně hodnotám akustického tlaku, jak je intenzita zvuku reprezentována na straně počítače. Člověk je mnohem citlivější na změny zvuku o nízké intenzitě a s postupným růstem akustického tlaku se odpovídající růst hlasitosti subjektivně zpomaluje. Této vlastnosti můžeme využít i v průběhu vizualizace. Vypočtenou efektivní hodnotu signálu zarovnanou do oboru hodnot 0 až 1 můžeme upravit pomocí logaritmické funkce podle vztahu (3.2) v závislosti na tom, jak silně chceme, aby se subjektivní faktor vnímání hlasitosti projevil. I zde značí intenzitu signálu získanou výpočtem RMS, k je koeficient pro úpravu citlivosti.

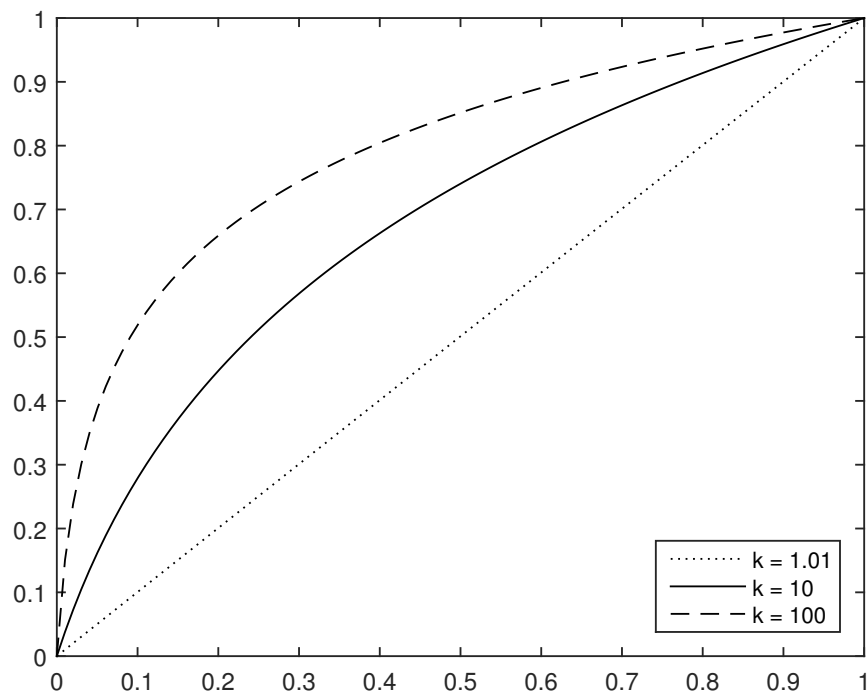
$$\log_k(I * (k - 1) + 1) \quad (3.2)$$

Této vlastnosti je využíváno v některých scénách ve finální aplikaci, v některých však působí lépe, pokud se k hlasitosti za účely vizualizace chováme jako k čistě lineární objektivní veličině.

Za zmínku také stojí fakt, že při stejné hodnotě akustického tlaku je subjektivně vnímaná hlasitost zvuku o různých frekvencích odlišná. K útlumu nedochází jen na krajích slyšitelného spektra, i frekvence v jeho mezích jsou člověkem registrovány s rozdílnou citlivostí, jak ukazuje obr. 3.2. Z pohledu vizualizace však tento jev můžeme většinou zanedbat.

3.2 Frekvenční charakteristika zvukového signálu

Vizualizaci lze realizovat za použití třeba jen jediného parametru, jako je právě hlasitost hudby. Celková hlasitost je však pouze jednorozměrná veličina, s čímž jsou také spojeny omezené možnosti pro její mapování. Za pomoci technik pro zpracování signálu však můžeme dojít k zajímavějším údajům, jako je zastoupení různých frekvenčních pásem. Zjistíme tak, jestli je analyzovaný zvuk tvořený převážně basy, středou či výškami, respektive která složka se na celku podílí jakou měrou. Přitom se nemusíme omezovat na rozdělení do pouhých třech kategorií.



Obrázek 3.1: Znázornění vztahu (3.2) pro úpravu hlasitosti dle lidského vnímání s různě voleným koeficientem citlivosti

Zastoupení tónů různých výšek (frekvencí) je možné zjistit dvěma základními způsoby – můžeme využít buďto rozkladu signálu aplikací diskrétní Fourierovy transformace (typicky za pomoci přibližných avšak mnohem rychlejších variant algoritmů známých jako FFT [18]), nebo filtrací vstupního signálu pomocí low-pass, band-pass nebo high-pass filtrů [8].

3.2.1 Vnímání výšky tónu člověkem

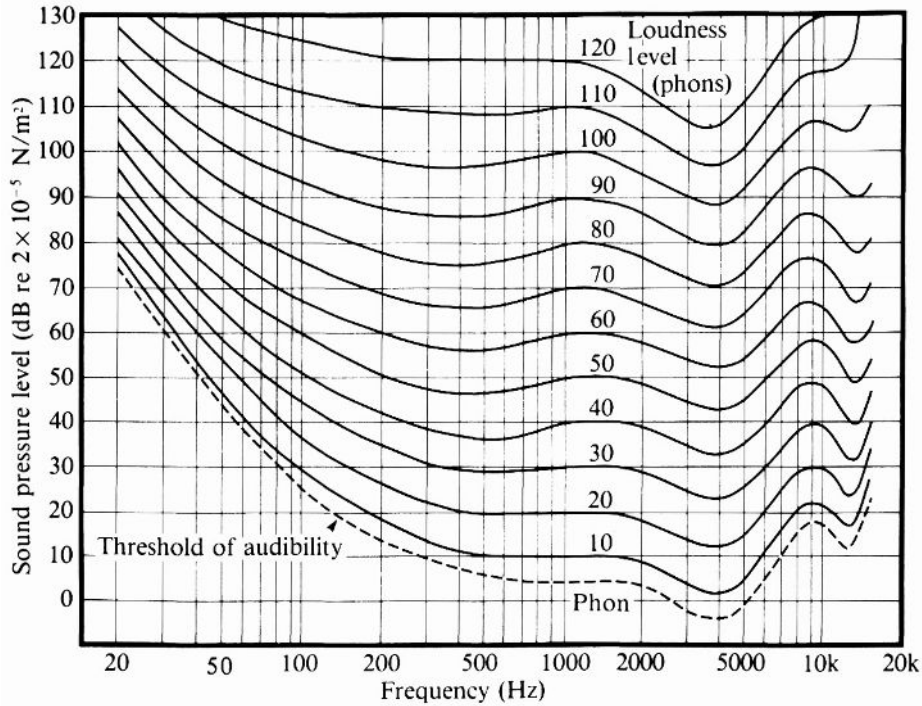
Obdobně, jako je tomu s hlasitostí, i výšky tónů člověk vnímá logaritmicky vůči objektivnímu měření – frekvenci. To se také odráží v hudební teorii. V té je zavedené uspořádání tónů do oktáv, což se z pohledu vnímání zvuku jeví jako lineární škála. Ve skutečnosti však platí, že tón o oktávu vyšší než referenční má frekvenci dvojnásobnou, tedy

$$f_1 = \frac{f_2}{2} \quad (3.3)$$

Standardní frekvenční rozsah slyšitelný člověkem se u jednotlivých jedinců liší, běžně je však udáván orientačně mezi 20Hz a 20kHz. Pokud tedy chceme rozdělit signál do N složek s ohledem na lidské vnímání zvuku, můžeme určit hranice jednotlivých intervalů podle vztahů

$$f_{Low} = t^{(t+(1-t)*\frac{i-1}{N})} \quad (3.4)$$

$$f_{High} = t^{(t+(1-t)*\frac{i}{N})} \quad (3.5)$$



Obrázek 3.2: Citlivost lidského sluchu na různé slyšitelné frekvence [4]

$$t = \log_{20000} 20 \quad (3.6)$$

Podle této formule také sestavuji band-pass filtry ve vizualizační aplikaci s tím, že využívám rozdělení zmíněného spektra do sedmi oblastí. Po této filtraci zjišťuji relativní intenzitu zvuku v jednotlivých pásmech pomocí RMS. Těchto údajů pak využívám v samotných vizualizačních scénách.

3.3 Detekce rytmu v živé hudbě

Teoreticky je možné ze vstupního signálu odhadnout rychlost, respektive rytmus hudby. To typicky probíhá sledováním výrazných pozitivních změn v hlasitosti (primárně v nižších frekvenčních pásmech). Problematika je však mnohem komplikovanější a spolehlivý princip použitelný pro všechny hudební styly zatím neexistuje. Ba dokonce například u klasické hudby ani pravidelný a stálý rytmus být nemusí.

Přístupy, které tento problém řeší, pracují nad velkými úseky skladeb a používají sofistikovanějších statistických přístupů (*comb filter* [15] atp.) a multiagentních algoritmů. Jsou proto výpočetně značně náročné a pro aplikace pracující v reálném čase nepoužitelné.

Odlehčenou variantou tohoto problému, respektive jeho podproblémem, je detekce rázů (hlavních dob, anglicky *beat*) v hudbě. V tomto případě není kladen tak silný důraz na absolutní pravidelnost v průběhu celé skladby, zaměřujeme se vždy jen na aktuální události. Ačkoliv ani tento problém nemá univerzální řešení a jeho přesnost je velice citlivá na vstup,

je pro něho možné navrhnout řešení pracující v reálném čase, které dosahuje dostatečně zajímavých výsledků pro účely vizualizace.

3.3.1 Algoritmus detekce rázů v reálném čase

Pro účely vizualizace není tak klíčová přesnost, jaká by byla nezbytná například pro analýzu struktury hudebních děl pro jejich následné porovnávání. V našem případě je důležitější zanedbatelná výpočetní náročnost vyhodnocování a vzhledem k tomu, že celý zvuk zpracováváme zpětně, také co nejmenší zpoždění samotného vyhodnocení. Pokud budeme schopni rozhodnout o přítomnosti hlavní doby až o tři snímky dále než kdy nastala, nebude už mít smysl efekt mapovaný na tuto událost zobrazovat.

S ohledem na tyto potřeby jsem po analýze stávajících (přesnějších avšak náročnějších) řešení navrhl vlastní algoritmus, který určí, zda je v daném rámci zpracovávaného signálu přítomna přízvučná doba (*beat*). Pracuji s velikostí jednoho rámce jako s 1024 vzorky signálu, tedy analyzuji 43 rámců za sekundu.

Klíčové u podobných algoritmů bývá vyhledávání lokálních maxim v hlasitosti signálu nebo některých frekvenčních pásmech – *peak detection*. Z počátku jsem proto také pracoval s tímto jako rozhodujícím ukazatelem. Výsledky byly samozřejmě závislé na konkrétní skladbě, avšak většinou byly hlavní úhozy v hudbě zachyceny. Tato metoda však má jedno důležité úskalí, a sice její včasnost. Abychom odhalili přítomnost takového lokálního maxima, musíme buďto znát průběh signálu dopředu, nebo ho můžeme analyzovat až se zpožděním. Pro odstranění šumu je třeba celý signál vyhlazovat (průměrovat), tedy znát ještě alespoň hodnoty dvou nebo tří rámců za tím, ve kterém byl *beat* ve skutečnosti přítomný. Výsledný efekt se proto ve vizualizaci může projevit až se značným zpožděním a jím evokovaný dojem se z něho v kombinaci s posunutím vůči hudbě silně vytrácí. Proto jsem později od tohoto přístupu ustoupil.

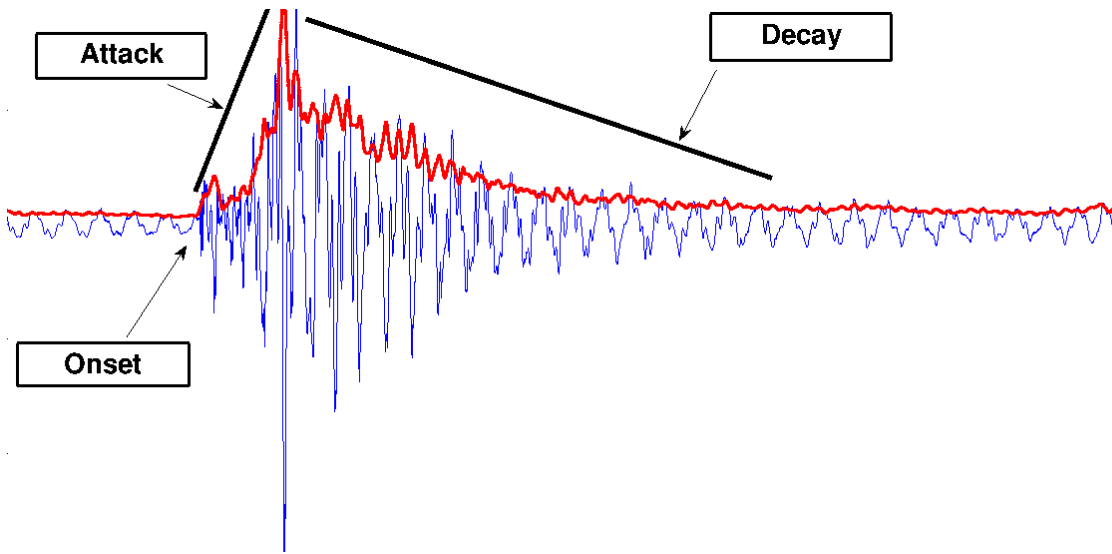
Jako základ pro ve výsledku použitou metodu jsem zvolil jiný ukazatel – vyhledávání nástupních hran takovýchto vychýlení v hlasitosti (*onset detection*, obr. 3.3). Ne každé razantní zesílení musí v hudbě být součástí hlavní doby, avšak jde-li nám o vyhodnocování v kontextu vizualizace, a tedy o okamžitou odezvu, jsou výsledky založené na tomto přístupu divácky mnohem líbivější.

Jednoduché zjištění přítomnosti těchto náběhů lze realizovat jako sledování poměru intenzity aktuálního rámce s rámci předchozími. Pokud je intenzita aktuálního vyšší než C -krát průměr předchozích, může se jednat o námi hledaný a potenciálně hudebně zajímavý úsek. Hodnota C je volena v závislosti na konkrétní skladbě či žánru, například 1,5. Musíme tedy znát intenzitu dříve analyzovaných rámců, např. za předchozí dvě vteřiny, se kterými aktuální hodnotu porovnáváme.

$$I_{act} > C * \frac{1}{n} \sum_{i=1}^n I_i \quad (3.7)$$

Abych v algoritmu upřesnil, že jde o první část nástupní hrany a ne jen o celkově silnější oblast v rámci daného úseku skladby, srovnávám aktuální intenzitu také s váženým průměrem několika předchozích rámců, vážených podle jejich pozice v historii.

$$I_{act} > C * \frac{1}{\sum_{i=1}^n 0.5^{i-1}} * \sum_{i=1}^n (I_i * 0.5^{i-1}) \quad (3.8)$$



Obrázek 3.3: Znázornění průběhu tónu v akustickém signálu [5]

3.3.2 Rozšíření podmínek onset detection algoritmu

Výše zmíněná jednoduchá metoda funguje jen pro velice omezené množství skladeb resp. žánrů, protože je silně závislá na volbě konstanty C . Postupným zkoušením jsem hledal, kdy je jaká hodnota nejvhodnější. Ukázalo se, že pro hudební žánry se silně zvýrazněným rytmem, jako je rap či techno, je vhodné volit C okolo 1,6. Oproti tomu u žánrů, kde vystupuje mnoho nástrojů po celou dobu a intenzity hlavních dob se tolik neliší (metal, rock), taková hodnota téměř nikdy nenastane a je třeba C volit jako přibližně 1,2. Samozřejmě cílem je, aby vizualizační systém nebylo nutné pro každou skladbu ručně upravovat. Hledal jsem proto cestu, jak volit hodnotu C v závislosti na aktuálně analyzovaném signálu.

Pokud zaznamenáváme historii hodnot z předchozích analyzovaných rámců, je nejvhodnější cestou analýza „rozkolísanosti“ přehrávaného signálu, kterou můžeme určit pomocí rozptylu (variance) získaných hodnot.

$$var = \frac{1}{n} \sum_{i=1}^n (I_i - \bar{I})^2 \quad (3.9)$$

Pokud je rozptyl menší, tedy signál je více zašuměný (metal, rock), volíme menší i rozhodovací konstantu. Při větší varianci naopak předpokládáme, že jsou rázy v hudbě výraznější a konstanta C bude volena okolo 1,6. Konkrétní hodnoty, pro jaký rozptyl volit jaké C , záleží na hodnotě maximální možné intenzity (hlasitosti) signálu.

Obzvláště u hudby s elektrickými nástroji můžeme využít toho, že je rytmus nejčastěji udáván v nejnižších oblastech frekvenčního spektra. Samozřejmě toto pravidlo neplatí vždy, ale jde nám v první řadě o pokrytí netypičtějších případů. Této znalosti můžeme využít v našem algoritmu. Namísto porovnávání intenzity vzorků jednotlivých rámců můžeme porovnávat vážené intenzity s ohledem na akustickou výšku. Pokud tedy chceme klást větší důraz na basové polohy, můžeme celkovou intenzitu pro účely *onset detection* vnímat jako

součet intenzit jednotlivých pásem násobených jejich vahou.

$$I_w = \sum_{f=0}^{n_f-1} (I_f * 0.6^f) \quad (3.10)$$

Vzhledem k tomu, že stále porovnáváme intenzitu (ačkoliv váženou) s předchozími, je běžné, že je jedna hudební doba hlášena několikrát krátce po sobě, protože je dána zvukem trvajícím nějaký čas. Pokud tomu chceme předejít, můžeme přidat *koeficient necitlivosti*. Ten při nově detekovaném rázu nastavíme na nějakou téměř nedosažitelnou hodnotu (100) a v krátkém čase přibližujeme k hodnotě 1. Podmínku pro detekci nástupu hlavní doby pak upravíme jako

$$I_{act} > t * C * \bar{I} \quad (3.11)$$

kde t je nově zavedený koeficient pro dočasné znecitlivění detekce.

Dále jsem do vyhodnocovacího algoritmu zařadil práci s intenzitou předchozího nalezeného rázu. Je typické, že po sobě jdoucí rázy mají podobnou váženou intenzitu. Můžeme proto udržovat v proměnné intenzitu předchozího nalezeného rázu a intenzitu aktuálního rámce porovnávat s ním. To eliminuje velké množství chyb 2. typu (*false positive*). V mém případě jsem určil podmínku tak, že aktuální intenzita musí být vyšší než hodnota dopočtená lineární interpolací od intenzity předchozího rázu po hodnotu nula, za dobu 4 vteřin. Za 1 vteřinu je tedy požadována intenzita alespoň 75% předchozí, za 4 vteřiny podmínka zcela vymizí.

Posledním posílením algoritmu je v mém případě něco, co by bylo možné nazvat jako *beat prediction*. Zaznamenávám časy mezi předchozími nalezenými rázy a na základě rozdílu posledních dvou zvyšuji vypočtenou váženou intenzitu aktuálního rámce, pokud je v blízkém okolí času, kdy by se na základě předchozího rozdílu měl objevit nový. Díky tomuto je celá detekce rytmu o něco stabilnější v případě, že odhalí dvě po sobě následující hlavní doby. Bohužel to má i svoji nevýhodu - pokud je dvakrát po sobě detekovaný nějaký náběh na špatném místě, mírně se znesnadní zotavení z tohoto stavu. V praxi však častěji popisovaná výhoda převyšuje nad druhou možností.

3.3.3 Napojení rytmu na vizualizaci

Pro účely vizualizace poskytuje třída zprostředkovávající funkcionalitu *beat detection* metody pro zjištění přítomnosti pravděpodobné hlavní doby v aktuálním snímku. Tím lze získat binární informaci, na základě které se spouští ve scéně nějaký vizuální efekt. Navíc pak využívám metody pro zjištění pravděpodobnosti pravidelnosti aktuálního detekovaného rytmu, která probíhá na základě srovnání posledních třech nalezených rázů v signálu. Člověk totiž očekává v hudbě pravidelný rytmus a pokud je vynechána jedna hlavní doba mezi ostatními, nebo je navíc detekována některá vedlejší, je lepší tento fakt ve vizualizaci přiznat, aby byl vizuální projev projekce v souladu s očekáváním posluchače. Při detekované nepravidelnosti rytmu proto např. změním směr efektu (viz ukázky z aplikace na příloženém DVD).

Kapitola 4

Vizualizace a její projekce

Technika pro samotnou vizualizaci a její zobrazování byla volena s ohledem na dva základní požadavky. Nejdůležitější bylo navrhnout takové řešení, které by diváka lépe vtáhlo do celého audiovizuálního zážitku. Z tohoto pohledu je důležité co největší obklopení pozorovatele, mimo to pak snaha o vyvolání dojmu celkové prostorovosti a tím zdánlivé „monumentálnosti“ projekce. Druhým aspektem byla možnost realizace navrhovaného řešení v prostorách Institutu intermédií, od čehož se pak odvíjejí určité limitace a vlastnosti realizované aplikace.

4.1 Dostupné prostory a použitá technika

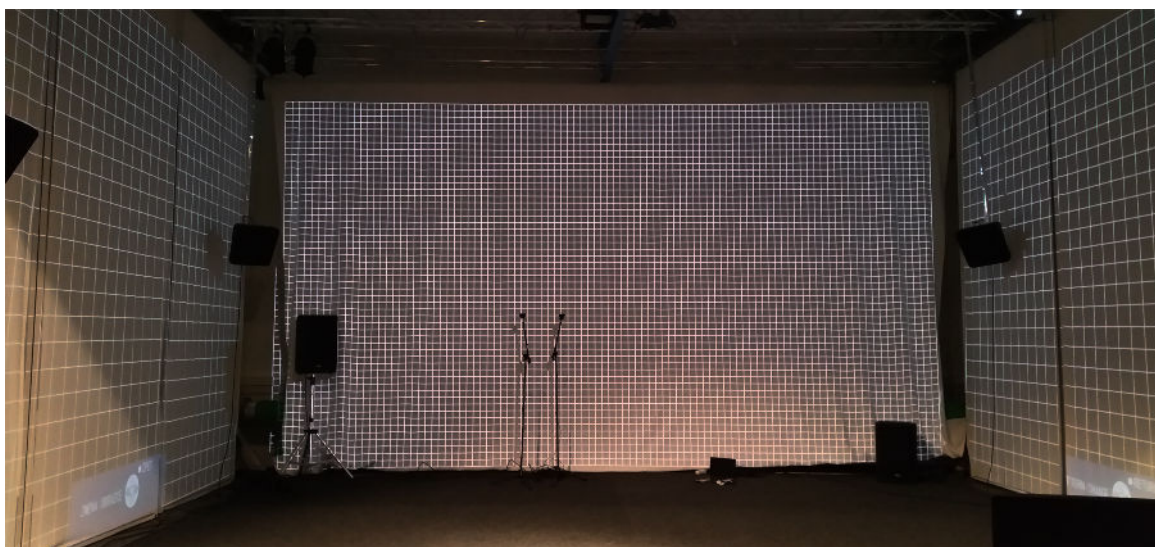
Hlavní část Institutu intermédií (IIM) je tvořena volným prostorem (halou) o velikosti přibližně 9 na 8,5 metrů, něco přes 5 metrů vysokým. Nad tímto prostorem jsou umístěny pojezdové vozíky na stropních kolejnicích se zavěšenými rámy pro možnou rekonfiguraci haly. Na ně lze připevnit nejrůznější techniku jako jsou reflektory pro osvětlení představení, v našem případě pak dodatečné projektory.

Přes většinu přední stěny je natažené plátno vhodné pro projekci z hlavního projektoru, který lze uzpůsobit tak, aby svým obrazem pokryl celou jeho plochu. Postranní stěny jsou tvořeny z bíle potažených pojízdných panelů, které lze přesunout dle potřeby a vytvořit tak celistvou bílou plochu. Obraz na postranní stěny promítají projektory umístěné na protilehlé straně na konstrukci nad samotným prostorem haly.

Použité projektory v této konfiguraci jsou schopny na výšku pokrýt téměř celé postranní stěny, horizontálně pak tvoří obraz o šířce 6 metrů. Celkově tedy probíhá projekce na plochu přibližně 8,5*5 metrů v případě přední a 6*5 metrů v případě postranních stěn (obr. 4.1).

4.2 Projection mapping v Institutu Intermédií

Projection mapping, u kterého je cílem vytvoření dojmu ohýbání v reálném světě, je technika náročná a citlivá na prostor. Pokud celou projekci sledujeme naživo, je důležité, aby byla promítací plocha vzdálena dostatečně daleko na to, abychom mohli zanedbat vzdálenost očí, jejich přirozené ostření a konvergenci. V opačném případě mozek celkem spolehlivě identifikuje obraz jako dvourozměrný. Aby tomu tak nebylo, bylo by nutné, aby obě oči směřovaly *do*

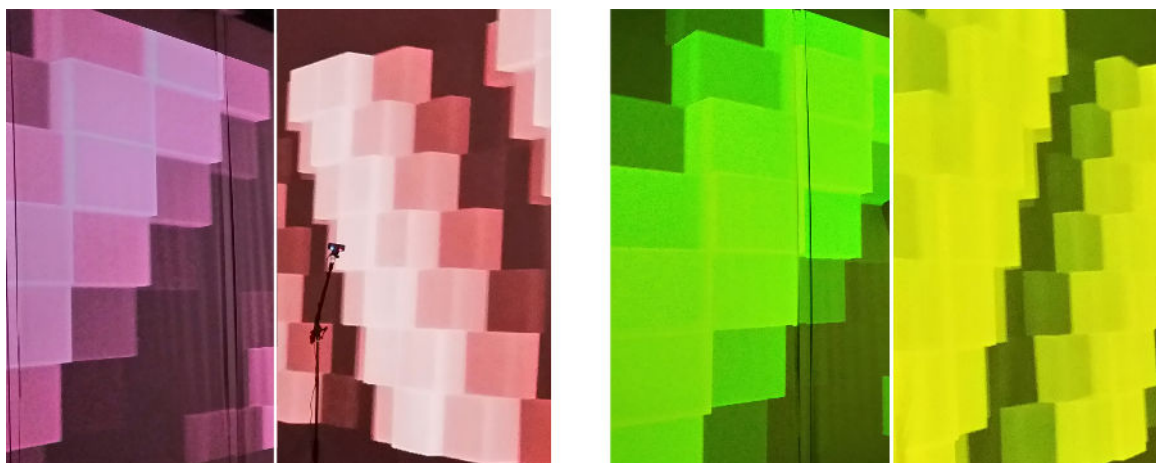


Obrázek 4.1: Hlavní hala Institutu intermédií v průběhu nastavování projektorů

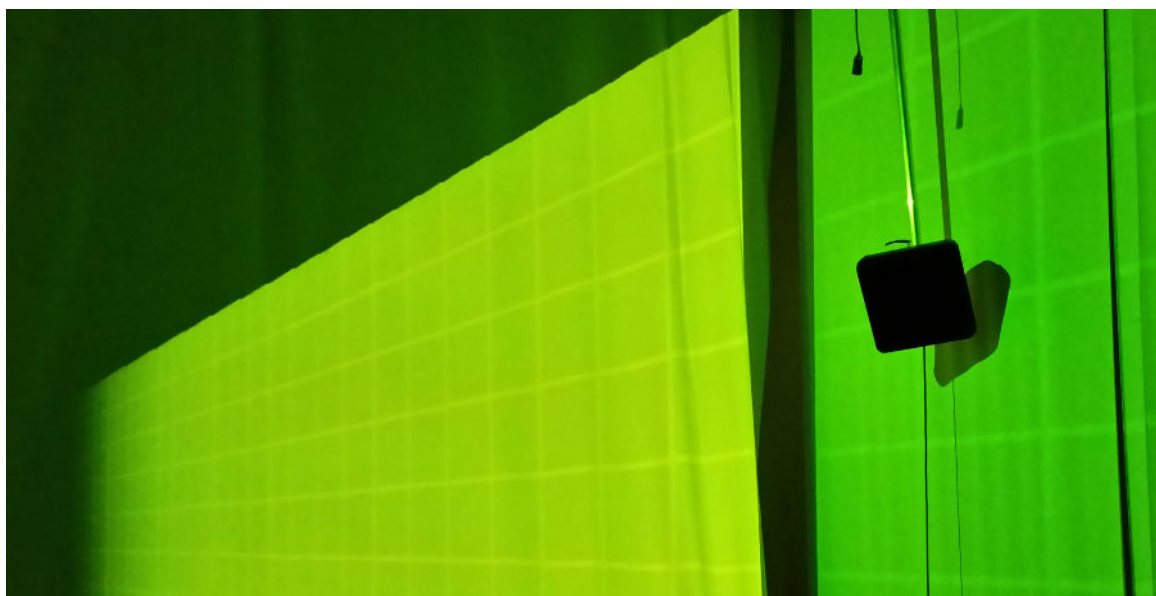
nekonečna (rovnoběžně, jako při sledování horizontu v otevřené krajině) a také byly *ostřeny na nekonečno*. Takový stav (*emmetropia*) nastává pro objekty vzdálené nejméně 6–8 metrů, proto by bylo pro vytvoření věrohodného dojmu potřeba mít zobrazovací plochy alespoň takto daleko, samozřejmě ideálně dvakrát tak dále. V rámci projektu v prostorách IIM je ale největší možná vzdálenost od bočních stěn jen 4 metry, kdy má ostření oka stále velký dopad, navíc v této vzdálenosti nelze zanedbat vzájemnou polohu a natočení očí. Výsledná iluze proto na takovém prostoru nemůže dosáhnout plného potenciálu.

Druhým faktorem ovlivňujícím výsledný dojem je doléhání projekcí na jednotlivé stěny. Plátno pro přední projekci v našem případě nezasahuje až do krajů celého prostoru a vzniká zde mezera, která prezentaci opticky roztrhává na tři části, viz obr. 4.1. Objekty, které se při projekci nacházejí právě na rozhraní dvou stěn, jsou tudíž vnímány, jako by zde docházelo k nějaké zdánlivé chybě. Mimo to také hrají značnou roli optické vlastnosti jednotlivých projektorů a vlastnosti povrchů, na které projekce probíhá. Materiál předního plátna má odlišné difuzní vlastnosti oproti postranním, stejně tak barevný projev hlavního projektoru je jiný než u bočních (obr. 4.2). Barevné ladění lze na projektorech mírně korigovat, lidské oko je však na rozdíly v jasů a barevnosti dosti citlivé a to se projevuje i na celkovém dojmu, zejména u světlejších scén.

Posledním výrazným neduhem při projekci na menším prostoru daných vlastností je odraz světla mezi jednotlivými stěnami (obr. 4.3). Pokud bude na levou stěnu promítáno větší množství zelené, značná část tohoto světla se odrazí také na přední plátno a jinak černá oblast bude mít nazelenalý odstín. Tím dojde k narušení obrazu před pozorovatelem. Tento jev, v počítačové grafice známý pod pojmem *color bleeding*, se nejvíce projevuje právě na rozhraní jednotlivých stěn a tím ještě zdůrazňuje jejich nedoléhání.



Obrázek 4.2: Rozdílné podání barev bočního a předního projektoru (bráno z jedné fotografie)



Obrázek 4.3: Zabarvení černých oblastí obrazu vlivem odrazu světla

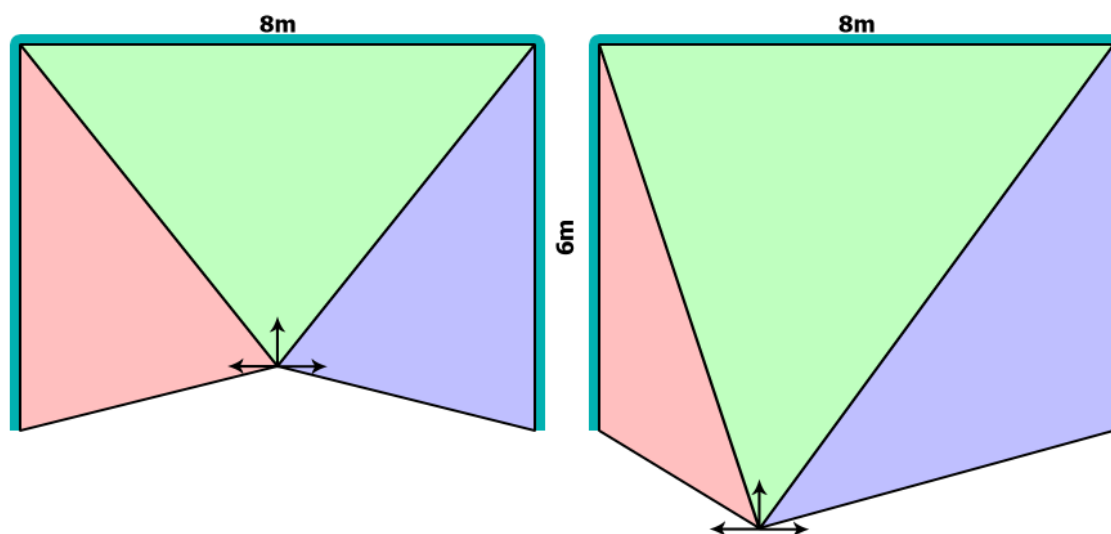
4.3 Realizace aplikace po stránce grafického výstupu

Vzhledem k tomu, že kvůli vytíženosti prostor IIM nemůžeme realizovat celou instalaci se vším potřebným napevno, používáme jako hlavní řídicí jednotky přenosného počítače. Tato zařízení obvykle mají limitovaný počet portů sloužících pro grafický výstup. Institut intermédií však disponuje zařízením TripleHead2Go společnosti Matrox [23], díky kterému postačí, aby měl notebook alespoň jeden výstup. Obraz z tohoto výstupu je pak rozdělen na tři stejné části, každému připojenému projektoru je pak zasílán odpovídající výřez. Při použití s tímto zařízením tedy musí být aplikace schopna vykreslovat všechny tři obrazy do jednoho okna.

Na druhou stranu při individuálním zapojení projektorů se každý hlásí pro operační systém odděleně. V takovém případě musíme vytvořit pro každý projektor speciální okno, které

půjde maximalizovat a upravovat nezávisle na ostatních. Za účely správy oken a vykreslování v nich využívám knihovny FreeGLUT [14], která slouží jako nadstavba pro práci s knihovnou OpenGL [26], využívanou pro finální rendering grafických prvků.

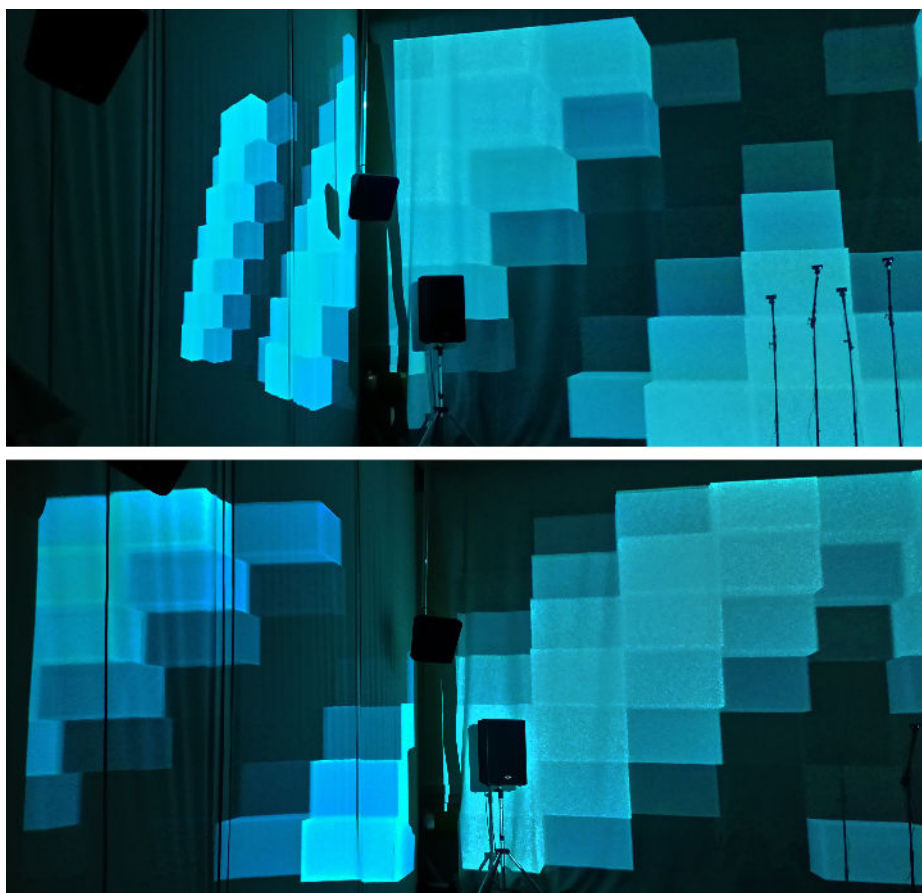
Z pohledu techniky Projection mapping je důležité to, jak odpovídajícím způsobem deformovat promítaný obraz, abychom kompenzovali perspektivní zkreslení při pozorování rovné plochy z určitého bodu v prostoru uživatelem. Pokud by projekce probíhala ze stejného místa, ze kterého je pozorována, žádná kompenzace zkreslení by nebyla potřeba brát v úvahu, protože by k ní docházelo přirozeně na základě promítání na šikmý povrch (vzhledem ke směru promítání). To je však možné jen u velkých projekcí jako je 3D mapping na budovy atp., v našem případě taková situace není realistická.



Obrázek 4.4: Deformace pohledových jehlanů (pohled shora) při různém umístění bodu pozorování. Šipky značí pohledový vektor kamer odpovídající směru promítání

Musíme tedy vyřešit problém, kdy projekce probíhá na zobrazovací plochu kolmo, avšak pozorovatel ji vidí pod určitým úhlem. To jsem zpočátku řešil ručně s využitím homografie, kdy známe pozici, kam se body promítají, a víme, kam bychom chtěli, aby se promítaly. Podle toho provedeme na základě maticových operací dodatečné zkosení. Nakonec jsem však využil toho, že OpenGL umožňuje vytvářet již zdeformované pohledové jehlany, které nejsou příliš realisticky modelovatelné čočkami v reálném světě. Jedná se však stále o lineární zobrazení, proto je odpovídající přepočít pro danou perspektivu v počítači snadno realizovatelný. Takto lze docílit korekce perspektivy, aby odpovídala pohledu pozorovatele i při promítání z libovolného jiného úhlu. Nastínění konfigurace kamer v daném prostoru ukazuje obr. 4.4 (pohled shora), při pohledu z boku je situace obdobná. Je vidět, jak je třeba pohledový jehlan deformovat v závislosti na zvoleném bodu, ve kterém se nachází pozorovatel. Při pohledu z jiného místa bude celá scéna působit *rozlomeně* a prostorový efekt se nedostaví (obr. 4.5).

V neposlední řadě je také nutné si uvědomit, že čím menší je dostupný prostor a tudíž čím blíže pozorovateli projekce probíhá, tím větší je potřeba přesnosti měření při nastavování konfigurace výsledné projekce. Realizovaná aplikace umožňuje v tomto ohledu nastavit prakticky všechny používané parametry pro uzpůsobení projekce celému prostoru pomocí jedno-



Obrázek 4.5: Srovnání projekce na rozhraní projekčních stěn z bodu mimo střed promítání a v něm

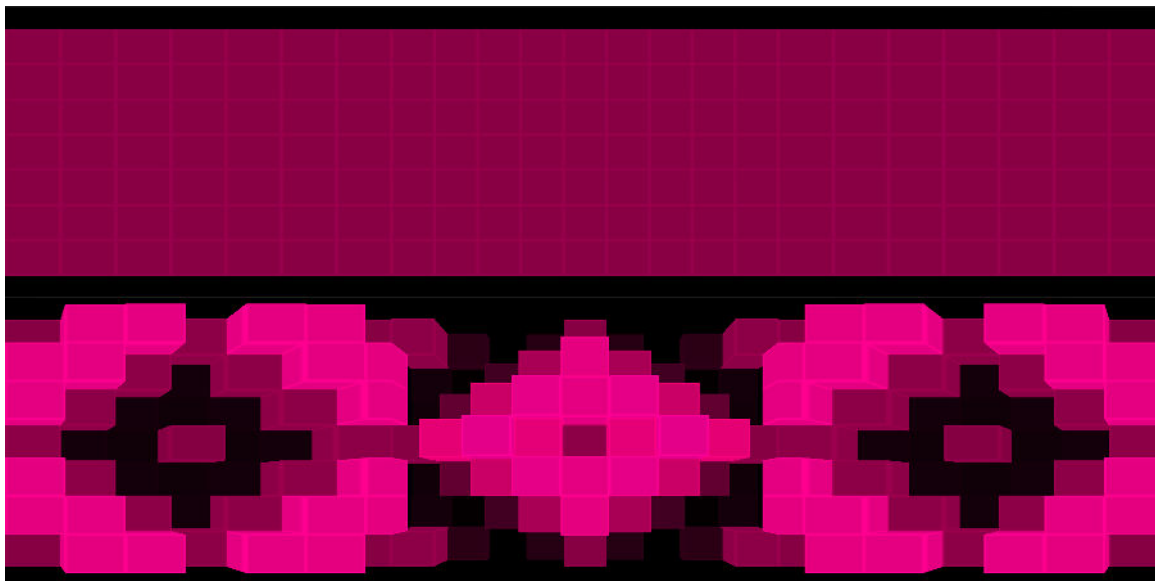
duchých textových konfiguračních souborů. Je třeba zabývat se například odsazením spodní hrany obrazu od fyzické podložky, na které se divák nachází, ale také možnými nepřesnostmi při nastavování pozic jednotlivých projektorů. Ve zmíněných konfiguračních souborech tak lze nastavovat fyzicky využívanou plochu pro projekci, velikost rámečků (*viewport*) v obrazu projektorů, pozitivní i negativní odsazení obrazu v horizontálním i vertikálním směru od okrajů pro jednotlivé projektory a mnoho dalšího. To umožňuje eliminovat nepřesnosti vzniklé při montáži projektorů v hale. Na obr. 4.1 můžeme vidět rozdíl ve vertikálně využitém prostoru jednotlivými projektory a horizontální mezery, které je třeba kompenzovat.

4.4 Návrh scén pro projection mapping

Vizualizační scény realizované v rámci práce měly za hlavní cíl demonstrovat využití a efektivnost rozebírané projekční techniky. S ohledem na toto kritérium byly také navrhované. Scény proto umožňují vedle audioreaktivního módu prohlížení také v podobě demo ukázek, na kterých lze vyhodnocovat výsledky samotné projekční techniky. Za tímto účelem byl volen i obsah scén. Jako ústřední objekty jsou zde použity virtuální krychle a kvádry. Jedná se totiž

o tvary, které jsou člověku nejdůvěrněji známe a tak poskytují nejlepší příležitost pro ohodnocení, do jaké míry působí umístění kostky ve virtuálním prostoru za promítacími stěnami realisticky a perspektivně korektně.

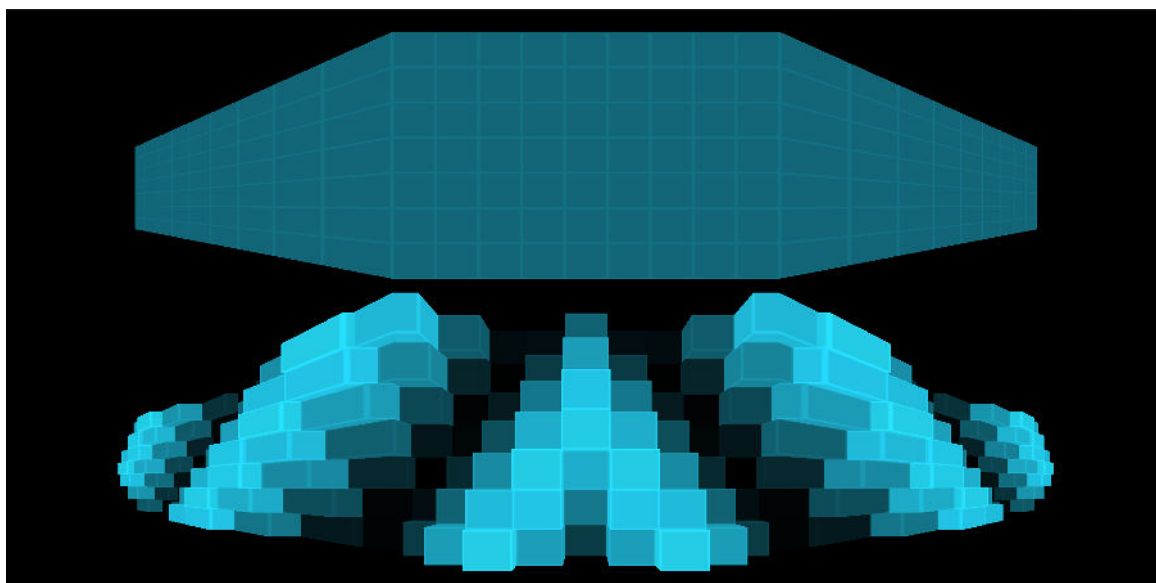
Úvodní scéna je nejjednodušší. Jejím cílem je navození dojmu, že jsou projekční plochy okolo uživatele tvořeny z velkých kvádrů (obr. 4.6). Ty vystupují do popředí respektive se prohlubují za fyzicky přítomné stěny a tvoří tak dojem jejich plasticity. Ve statické podobě je tato scéna ideální pro konfiguraci posunu a natažení obrazů jednotlivých projektorů, aby linie na všech stěnách doléhaly na sebe.



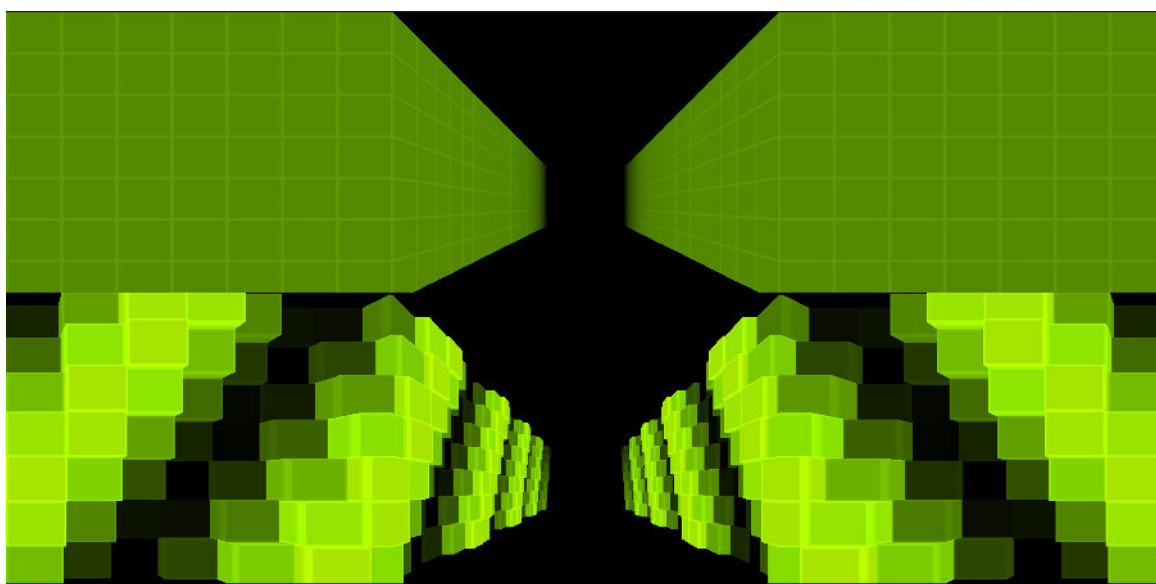
Obrázek 4.6: První scéna ve statické a demo variantě – při projekci na stěny vytváří iluzi vystupujících a zanořujících se kvádrů z jednotlivých stěn

Následující dvě scény se také opírají o fyzický prostor, ve kterém vizualizace probíhá, ale na rozdíl od předchozí jej opticky protahují daleko za rámec samotných projekčních ploch. V první z nich je cílem navození iluze, že přední stěna je stále přítomna tak, jak je v reálném světě, avšak namísto postranních je právě přední stěna protažena do prostoru jdoucího daleko za boční hranice haly (obr. 4.7). Ve druhé využíváme podobné myšlenky, ale v opačném smyslu. Vycházíme z levé a pravé stěny v reálném světě, snažíme se však vyvolat dojem, že zde není přítomné přední plátno a že postranní stěny pokračují daleko do prostoru před pozorovatelem (obr. 4.8).

U všech výše zmíněných scén se jedná primárně o znázornění průběhu hudby s ohledem na její rozdělení do jednotlivých výškových pásem. Intenzity v těchto pásmech jsou mapovány na vystoupení kostek ze stěny/za stěnu v odpovídající vertikální hladině. Zajímavé jsou také scény zohledňující jen detekovaný rytmus hudby. První je tvořena na rytmus reagujícími sloupy pokračujícími za prostorem haly v blízkosti jejích okrajů, jak ukazuje obr. 4.9. Ve druhé se jedná o průběžně se objevující krychle ve virtuálním prostoru za stěnami místnosti, které se pomalu pohybují a rotují. U této scény jsem však namísto standardního mapování na barvu, velikost či umístění zvolil originálnější variantu, při které na detekovaný ráz v hudbě reaguje scéna změnou průběhu plynutí času v ní. Pokud je tedy identifikována potenciální



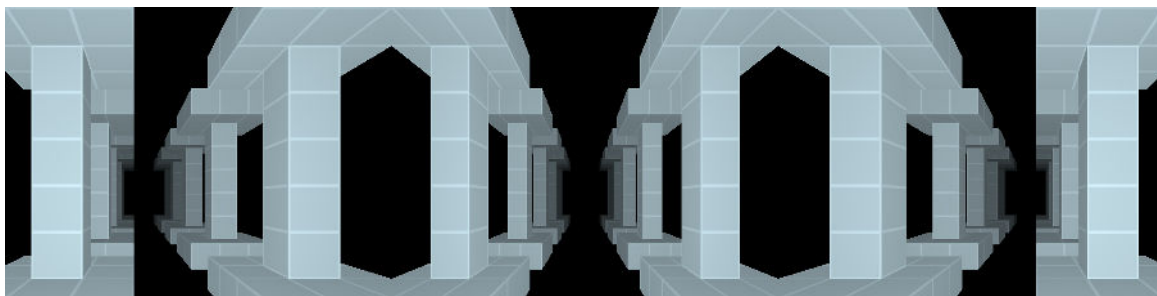
Obrázek 4.7: Druhá scéna ve statické a demo variantě – při projekci na stěny vytváří iluzi, jako kdyby přední stěna pokračovala dále do stran za fyzické postranní stěny



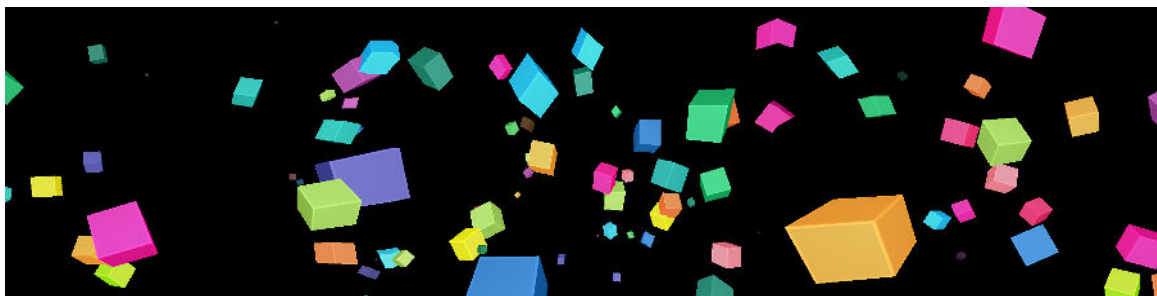
Obrázek 4.8: Třetí scéna ve statické a demo variantě – z nastaveného místa pozorovatele vytváří iluzi pokračujících postranních stěn do dálky, za fyzicky přítomnou přední stěnu

hlavní doba v analyzovaném signálu, jsou animace letu kostek přehrávány rychleji. Navíc zohledňují pravidelnost detekovaného rytmu a v případě vynechání nějaké doby či detekce doby vedlejší měnící směr, v jakém animace probíhá (obr. 4.10).

Za účely testování výsledného dojmu podporují scény mnoho variant a přizpůsobení. Těmi jsou inverze hlavního efektu (směr letu či růstu), aktivace či deaktivace vlastního vyzařování materiálu objektů, odezvy na rázy v hudbě, změna barevného ladění a další.



Obrázek 4.9: Čtvrtá vizualizační scéna s řadovými sloupy ve všech směrech



Obrázek 4.10: Scéna využívající detekce rytmu k řízení rychlosti animací

Kapitola 5

Přizpůsobování projekce divákovi

Ačkoliv už díky použití samotné techniky 3D mapování můžeme získat zajímavé výstupy, výsledek stále nebude zcela přesvědčivý z pohledu uvěřitelnosti celkového efektu. Hlavním problémem je, že dosud popisovaná aplikace bude schopna vyvolat zamýšlený efekt jen při pohledu z jednoho konkrétního bodu v sálu. Pokud se divák od tohoto místa vzdálí o více jak čtvrt metru, nebude odpovídat perspektivní zkreslení, linie na rozhraní jednotlivých projekčních pláten nebudou opticky navazovat a celá iluze hloubky se vytratí (obr. 4.5).

Abychom tomuto neduhu zabránili a umožnili pozorovateli volný pohyb se zachováním zdánlivého rozšíření prostoru, musíme celou projekci uzpůsobovat s ohledem na jeho aktuální polohu. Hlavním problémem tedy je, jak v dostupných prostorách identifikovat, kde přesně se pozorovatel nachází.

5.1 Výběr techniky zjišťování polohy

V rámci návrhu celého projektu jsem rozmýšlel volbu techniky, pomocí které by bylo možné polohu pozorovatele určovat. Většinu možných přístupů jsem však musel zavrhnout na základě omezení, jako je pořizovací cena potřebných zařízení či vysoká nepřesnost. Úvodní ideou byla lokalizace s využitím intenzity signálu, například ze spárovaného mobilního telefonu. Bylo by možné určit napevno vertikální polohu očí dle výšky pozorovatele a v rozích haly umístit dva vysílače WiFi či bluetooth signálu [9]. S přibývající vzdáleností přijímače od takovýchto vysílačů dochází k úbytku síly signálu. Doufal jsem, že by bylo možné takovýto systém zkalibrovat na několika určených bodech, zjistit vlastnosti úbytku na přijímači s přibývající vzdáleností a přibližnou vzdálenost pak dopočítávat a následně určit i horizontální polohu diváka v hale. Tato metoda je experimentálně využívána v obchodních centrech či na stadionech. Provedl jsem proto sérii vlastních pokusů pro obě zmíněné bezdrátové technologie. Bohužel se ukázalo, že i při přímé viditelnosti mezi přijímačem a vysílačem touto metodou nelze dosáhnout přesnosti vyšší než 3 metry. I tato hodnota je ale silně optimistická.

Existují i další metody lokalizace [37], například na základě vibrací (ve chvíli, kdy se uživatel pohybuje), akustické metody, kdy má divák připevněný ultrazvukový vysílač a v prostoru jsou rozmístěny přijímače, nebo i přístupy založené na působení magnetického pole. Na rozdíl od výše zmíněného přístupu však neposkytují dostatečný komfort a nebo vyžadují speciální vybavení a vlastnosti prostoru. Nakonec jsem tedy zvolil nejběžněji používanou techniku –

optickou lokalizaci využívající dvojice kamer. Tato technika má největší spolehlivost, zároveň není třeba žádné drahé vybavení a při správném návrhu aplikace ani příliš složitá instalace.



Obrázek 5.1: Spojené kamery pro stereo vidění (stereo rig)

Běžně bývají kamery napevno spojené ve vzdálenosti v jednotkách až desítkách centimetrů, orientované stejným směrem. Tento přístup (tzv. *stereo rig*, viz obr. 5.1) je výhodný v tom, že lze jednorázově snadno změřit parametry celého systému a při budoucím použití už není třeba dalších úkonů. Pokud ale chceme kamerami pokrýt větší prostor s dostatečnou přesností i pro vzdálené objekty, potřebujeme, aby byly kamery relativně daleko od sebe a také v obecném natočení. Nemůžeme-li realizovat instalaci napevno, musíme počítat s tím, že bude třeba při každém použití celý systém nanovo kalibrovat.

Jako kamery pro realizovaný projekt jsem zvolil Sony PlayStation Eye [34] (obr. 5.2). Jedná se o kamery v dnešní době nejrozšířenější pro projekty počítačového vidění. O to se zasloužily svojí dostupností a zejména pak rychlostí snímání obrazu. Každá tato kamera je schopna zachytit až 120 snímků za vteřinu v QVGA rozlišení, 60 při rozlišení VGA. Můžeme tak vyhodnocovat dění před nimi s násobně větší frekvencí a není problém, ani když se po prostoru pohybujeme velkou rychlostí. Oproti konkurenčním produktům mají celkově citlivější senzor a navíc poskytují obraz bez komprese. Díky tomu z nich získáme překvapivě kvalitní obraz i za špatných světelných podmínek, jako je právě zatemněný sál.

5.2 Vyhledávání pozorovatele v obrazu kamery

5.2.1 Omezení daná podmínkami projekce

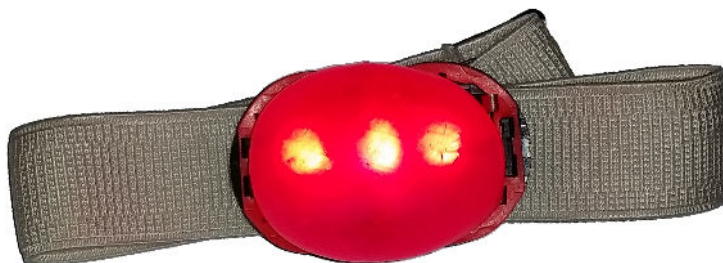
Abychom s pomocí navrhovaného systému dokázali zjistit, kde přesně se uživatel nachází, musíme být schopni určit jeho polohu v obrazech jednotlivých kamer. Pro výslednou projekci



Obrázek 5.2: Model kamery používané pro sledování diváka v prostoru

lze počítat s tím, že prostory budou zatemněné. Nemůžeme proto spoléhat na techniku vyhledávání obličeje v obrazu, protože pozorovatel pravděpodobně nebude vůbec vidět. Pokud nelze realizovat přímo vyhledávání obličeje, je nejjednodušším a zároveň nejspolehlivějším řešením vytvoření *markeru*, který bude mít uživatel připevněný na sobě. Vzhledem k zatemnění celé scény je pak téměř jediná možnost aktivně vyzařující marker, tedy nějaké menší světlo.

Ideálním by pro marker bylo využití takového záření, které lidské oko nevnímá, aby na uživatele působil co nejméně invazivně/rušivě. K tomu se nejvíce hodí infračervené diody. V použitých kamerách by se odstranil filtr blokující toto spektrum a vložil jiný, k potlačení snímání člověkem viditelného světla. V rámci realizace však velice dobře posloužilo zadní světlo na jízdní kolo, které bylo zatlumené, aby přímo neosvětlovalo okolí (obr. 5.3). Variantu s využitím infračerveného záření jsem také nerealizoval proto, že je při každé úpravě kamer vyžadována celková recalibrace a odstranění filtru v kamerách by tento proces zkomplikovalo.

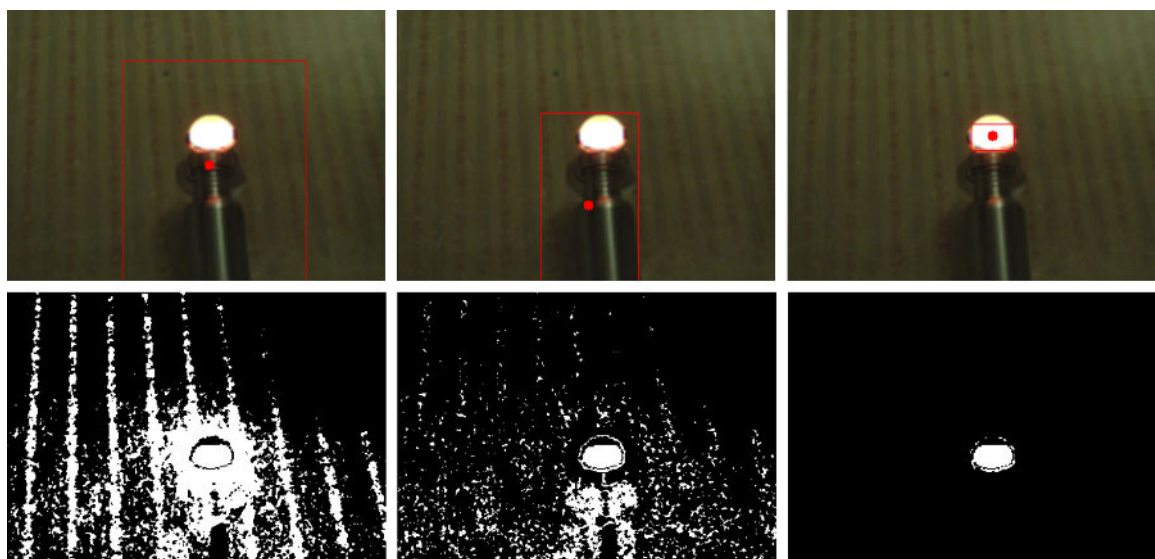


Obrázek 5.3: Provizorní marker využívaný k lokalizaci uživatele

5.2.2 Vyhledávání markeru v obrazu

Každý snímek pořízený kamerou lze reprezentovat dvourozměrnou maticí, složenou z vektorů představujících zastoupení jednotlivých RGB složek, ze kterých se skládá výsledná barva daného obrazového bodu. Naším cílem pak je určit jeden z těchto bodů (nebo i rozhraní bodů), kde se v obrazu nachází námi zvolený marker. Proces vyhledávání tohoto objektu funguje v několika krocích.

Důležitým mezikrokem je rozhodnutí, jestli jsou pro nás jednotlivé obrazové body (pixely) nějakým způsobem zajímavé a chceme tedy každému z nich přiřadit binární hodnotu, jestli tomu tak je či není. Tento proces bývá označován z anglického *thresholding* jako prahování obrazu. Nejprve převedeme obraz z RGB do HSV reprezentace, abychom měli lepší kontrolu nad barvami bez ohledu na světelné podmínky. Poté si zvolíme intervaly hodnot pro barevný odstín, saturaci a intenzitu jasu podle námi zvoleného markeru v zachyceném obrazu. Vyhledáme v HSV obrazu takové body, jejichž hodnoty spadají do odpovídajících intervalů. Podle této podmínky můžeme vytvořit kopii původního obrazu s tím, že body, které testem prošly, budou mít hodnotu 1 (bílá barva), ostatní hodnotu 0 (černá barva). Do výsledné aplikace jsem přímo zakomponoval interaktivní nástroj, kterým lze určovat rozsahy těchto hodnot a sledovat odfiltrovaný mezivýsledek (obr. 5.4). Hodnoty se po dokončení uloží do konfiguračního souboru pro další použití.



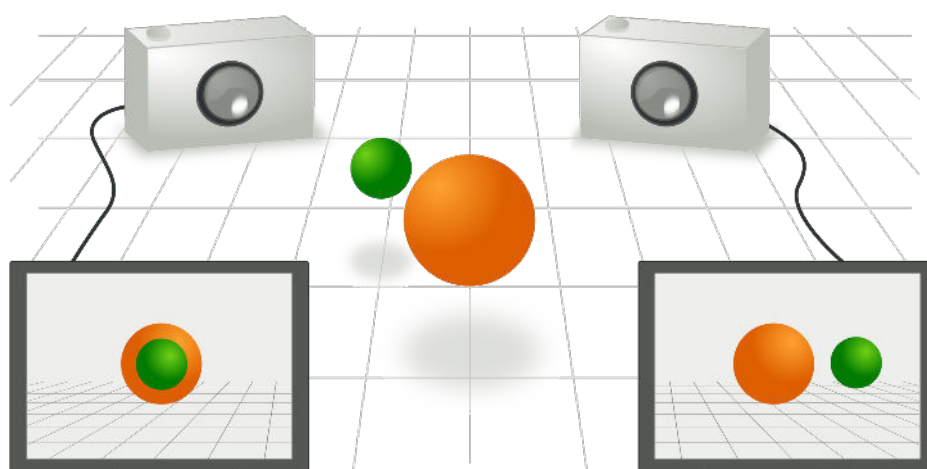
Obrázek 5.4: Dílčí mezivýsledky filtrace podle *Hue*, *Saturation*, *Value*

V získaném binárním obrazu vyhledáme kontury okolo skupin bílých pixelů. Pro tuto funkcionalitu využívám knihovny OpenCV [25]. Vybereme největší z nalezených kontur a pokud máme vhodně zvolené parametry pro filtraci původního obrazu, získáme představu o tom, kde se nachází náš marker. Jeho konkrétní pozici danou dvěma hodnotami můžeme získat buďto jednoduše jako střed ohraničujícího obdélníku (AABB) této kontury, nebo přesněji jako její obrazový moment (*image moment* [13]).

5.3 Kalibrace optického systému

5.3.1 Princip 3D rekonstrukce z dvojice 2D obrazů

Nahlíží-li dvě kamery na scénu z různých bodů v prostoru, zobrazí se objekty ve scéně v obrazu obou kamer do různých míst, jako je tomu na obr. 5.5. Při této projekci do dvou dvourozměrných obrazů však platí mnohá pravidla a omezení, která musí být zachována pro všechny body ze zachycované scény. Pokud známe polohu určitého objektu v prostoru obrazu obou kamer a pro tyto kamery známe některé charakteristické vlastnosti, jsme schopni aplikací těchto pravidel zpětně dopočítat, kde v prostoru se mohl objekt zachycený do daných obrazů nacházet. Oblast zabývající se touto problematikou se nazývá epipolární geometrie [17].



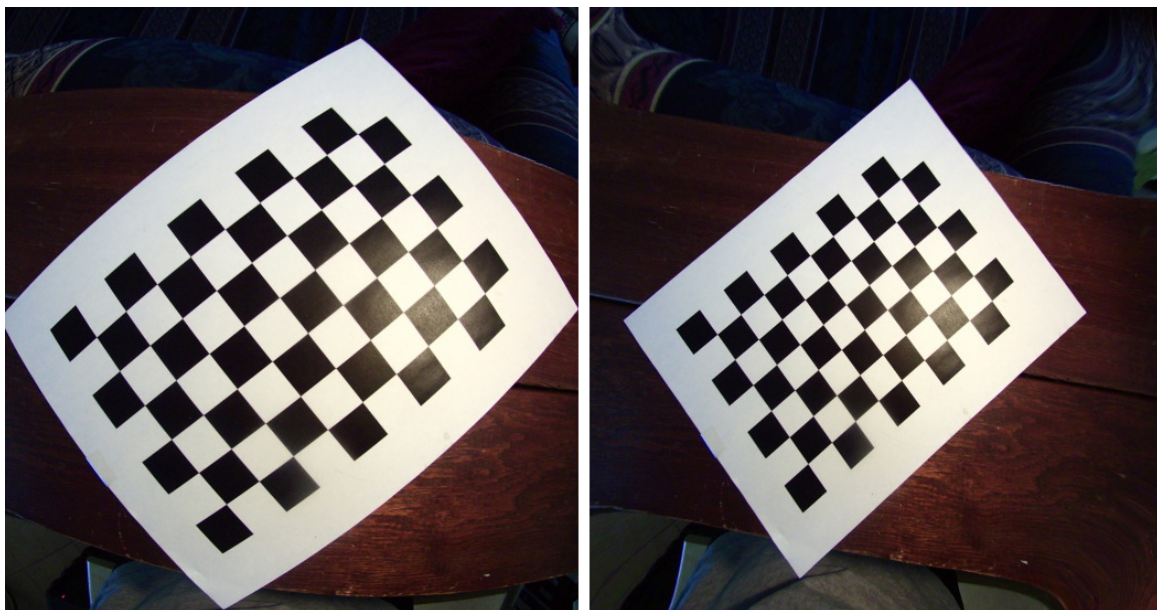
Obrázek 5.5: Znázornění zachycení scény dvěma kamerami [3]

V praxi je nejdůležitější znát relativní transformaci kamer, které snímky scény pořídily. Tuto transformaci lze vyjádřit pomocí vektoru symbolizujícího relativní posun středů modelových kamer a matice, který charakterizuje jejich vzájemné natočení. Z toho lze sestavit možné projekční matice těchto kamer. Ty vyjadřují zobrazení, které mapuje body z 3D prostoru do 2D obrazů, můžeme je ale využít i opačným směrem k rekonstrukci původní pozice objektu. Z těchto myšlenek vychází stereo rekonstrukce s využitím epipolární geometrie, která je názorně rozebírána v knize Ing. Tomáše Pajdly, Ph.D. [27].

5.3.2 Kalibrace jednotlivých kamer

Epipolární geometrie pracuje s modelem ideální kamery, tedy předpokladem, že lze reálné kamery aproximovat tzv. dírkovou (*pinhole*) kamerou [27]. V praxi však dochází při snímání reálnými kamerami ke zkreslení obrazu vlivem lomu světla při průchodu čočkami objektivu. Abychom dosáhli při prostorové rekonstrukci co největší přesnosti, je třeba nejprve toto zkreslení kompenzovat (obr. 5.6). Knihovna OpenCV [25] využívaná v rámci projektu poskytuje funkce, které v tomto pomáhají. Počítá se zde s radiálním a tangenciálním zkreslením. To

je z pohledu fyzikálních principů v kamerách vyvoláno, pokud nejsou čočky zcela paralelní s obrazovou rovinou.



Obrázek 5.6: Demonstrace radiálního zkreslení a odpovídající vyrovnaní obrazu [7]

Parametry pro kompenzaci zkreslení pořízeného obrazu jsou vnitřními parametry jednotlivých kamer a nemění se jejich přemísťováním v prostoru. V realizované aplikaci je připravena procedura, která tyto parametry dopočítá na základě série snímků pořízených při držení a natáčení šachovnice před kamerou. V těchto obrazech nalezne klíčové body a se znalostí skutečné relativní pozice těchto bodů určí parametry dané kamery. Ty pak aplikace uloží do konfiguračních souborů pro budoucí použití.

5.3.3 Kalibrace stereo systému

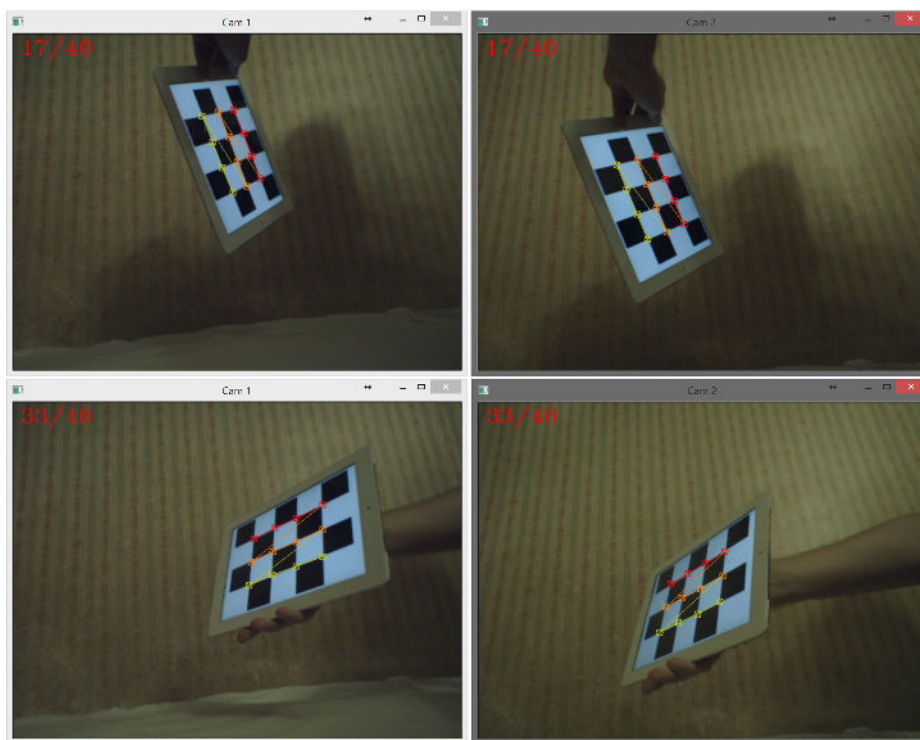
Obdobná procedura, jaká slouží pro kalibraci vnitřních parametrů jednotlivých kamer, je v aplikaci přítomna i pro zjištění vnějších parametrů kamer reprezentujících jejich vzájemné umístění v prostoru (obr. 5.7). Při ní musí šachovnice v jeden okamžik být viděna oběma kamerami. Následně jsou identifikovány odpovídající body v obou obrazech. Z těchto korendencí pak na základě pravidel daných epipolární geometrií dopočítáme posun a rotaci kamer relativně vůči sobě. Z těchto údajů sestavíme projekční matice kamer tak, že

$$P1 = [E|O] \quad (5.1)$$

$$P2 = [R|C] \quad (5.2)$$

kde R je matice symbolizující vzájemnou rotaci kamer a C je vektor udávající posun druhé kamery vůči první. Lze určit i jednotky, které mají být použity pro posun, avšak v našem případě na nich nezáleží. Naším cílem v této fázi je jen získat nástroje potřebné k převodu do libovolného trojrozměrného prostoru, nehledě na jeho jednotky či poměry. S použitím

takovýchto matic $P1$ a $P2$ pro rekonstrukci získáme souřadnice hledaných objektů v souřadném systému první kamery, ve kterém směr osy X a Y odpovídá souřadnému systému 2D obrazu, který pořizuje, osa Z pak vzdálenosti od kamery ve směru jejího pohledového vektoru. Pokud ale nechceme být limitováni umístěním kamer v rámci scény, musíme ještě nalézt transformační matici pro převod ze souřadného systému první kamery do reálných souřadnic scény.

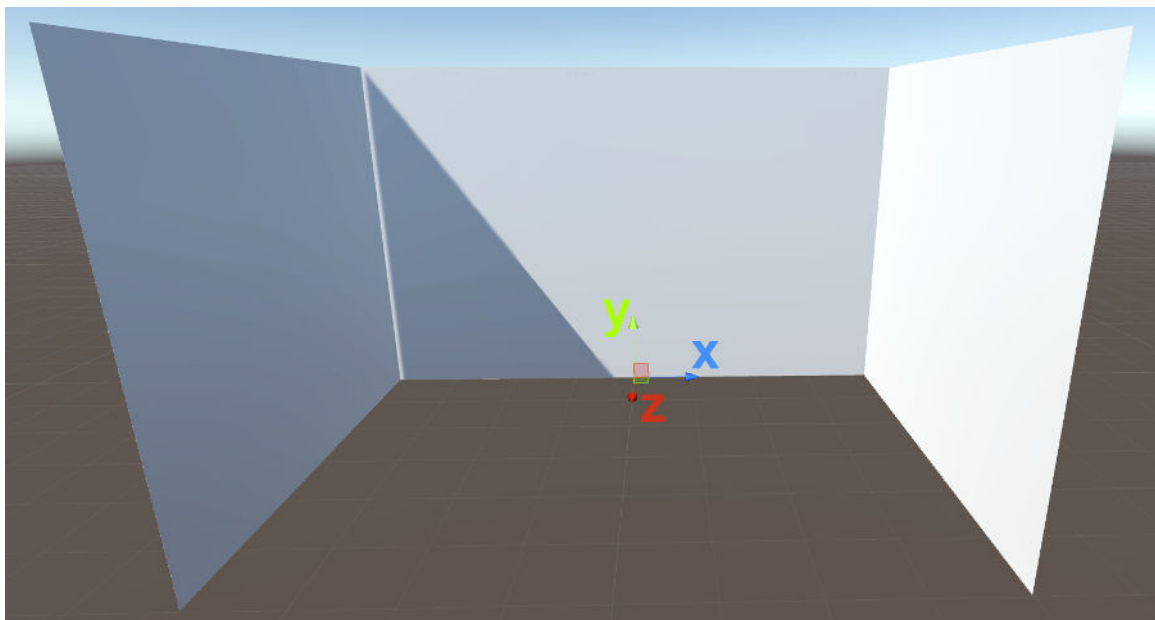


Obrázek 5.7: Průběh kalibrace vnějších parametrů stereo systému

5.4 Poloha pozorovatele v souřadnicích scény

Pokud jsme schopni získat souřadnice pozorovatele (respektive markeru) v souřadném systému určeném umístěním a natočením první kamery, stačí najít takovou matici zobrazení, která tyto souřadnice převede do souřadného systému scény. Ten je ve virtuálním prostoru aplikace určený s počátkem na podlaze přesně uprostřed předního promítacího plátna, s jednotkami v metrech, osou X směřující při pohledu na plátno doprava, osou Y směřující vzhůru a osou Z směřující od předního plátna směrem do prostoru, jako na obr. 5.8. Abychom takovou transformaci našli, stačí nám znát jen čtyři korespondence mezi těmito soustavami. V aplikaci v průběhu procedury hledající tuto transformaci tedy stačí postupně umístit marker na určené čtyři body v prostoru (volitelné v konfiguračním souboru kalibrace), které neleží v jedné rovině. Zjistíme, jaké souřadnice mají z pohledu první kamery. Na základě jejich reálných a zrekonstruovaných souřadnic dopočítáme, jaký obraz má počátek souřadné soustavy scény v prostoru kamery. Oba souřadné systémy posuneme tak, aby jejich počátky

splývaly. Vzhledem k tomu, že translace není lineární operací a nelze ji přímo vyjádřit maticovými operacemi, musíme ode všech bodů v souřadném systému kamery odečítat tento vektor. Pomocí pseudoinverze pak nalezneme matici A , která splňuje podmínku, že posunuté body v prostory kamery jejím přenásobením co nejlépe odpovídají určeným souřadnicím v reálném světě.



Obrázek 5.8: Znázornění soustavy souřadnic propojující reálnou scénu s virtuálními

Abychom mohli korektně uzpůsobovat perspektivní deformaci projekce, je možné v konfiguraci aplikace nastavit relativní posun markeru vůči očím pozorovatele. Pokud má například marker připevněný v oblasti hrudníku, je třeba k vypočítané pozici markeru po převodu do souřadného systému reálného světa přičíst orientačně 35 centimetrů v ose Y , pokud je marker umístěný na hlavě, naopak 15 centimetrů odečíst a podobně.

Tímto postupem tedy přesně víme, kde se nachází oči pozorovatele v reálném světě, a můžeme tomu podřídít výslednou projekci, aby uživatel nebyl omezován v pohybu a mohl zkoumat celý prostor a promítané virtuální scény z libovolného úhlu. Stačí napojit umístění diváka na pozici virtuálních kamer ve vizualizační aplikaci a jednotlivým pohledům, respektive jejich odpovídajícím pohledovým jehlanům, upravovat zkosení, jak bylo popisováno v předchozí kapitole, viz obr. 4.4.

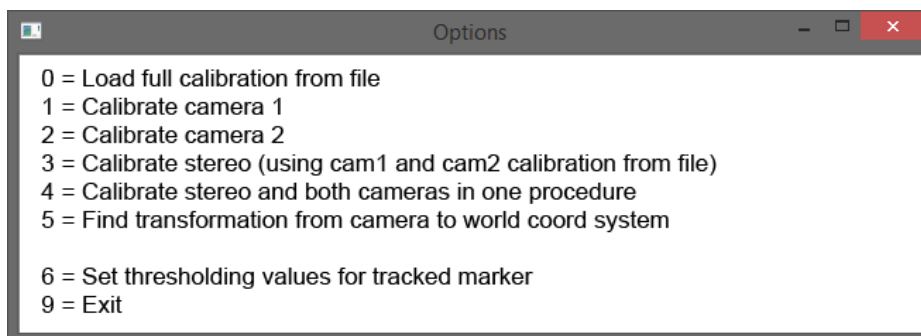
Máme-li ale k dispozici tak klíčovou informaci, jako je umístění uživatele v prostoru, nemusíme pracovat jen s přizpůsobováním projection mappingu na okolní stěny. Můžeme rovnou vytvořit scény, které přímo svým děním v nich reagují na umístění uživatele, generují objekty v závislosti na jeho poloze nebo i na jeho aktuální rychlosti, kterou se po prostorech pohybuje. V audiovizualizační aplikaci například využívám této informace k dodatečné interakci, kdy lze ovládat některé parametry vizualizačních scén, nebo jednoduše přepínat mezi jednotlivými scénami rychlým pohybem uživatele (resp. markeru) do stran.

Kapitola 6

Realizovaná audiovizualizační aplikace

6.1 Možnosti aplikace

Vzhledem k povaze a vytíženosti prostor IIM je nezbytné, aby aplikace nabízela možnosti, které celou instalaci systému usnadní, umožní její rekalicaci a úpravy klíčových nastavení. Jedná se především o hodnoty pro vyhledávání markeru v obrazu kamery, kalibraci vnějších parametrů stereo systému, kalibraci vůči souřadnému systému scény a uzpůsobení projekce pro kompenzaci nepřesností instalace projektorů. Pokud jsou kamery správně nainstalované a připojené k řídicímu počítači, zobrazí se po spuštění aplikace jednoduchá nabídka procedur, pomocí kterých lze celý systém patřičně nakonfigurovat, viz obr. 6.1. Vlastnosti spojené s fyzickými rozměry prostoru se nastavují ručně v konfiguračních souborech ve složce aplikace.



Obrázek 6.1: Nabídka konfigurace systému pro sledování pozorovatele

Po nastavení všeho potřebného pro sledování pohybu diváka v prostoru přejdeme k samotným vizualizačním scénám. Scény reagují na hudbu přivedenou na vstup počítače, aplikace je tedy orientována na vizualizaci výstupu druhého zařízení. Lze ji však snadno využít i pro vizualizaci zvukového výstupu přímo na daném počítači pomocí duplikování výstupního signálu na vstup. Některé systémy toto umožňují automaticky, v případě OS Windows se pak osvědčil program VB-Audio Virtual Cable. Vizualizační scény pracují s informacemi vytěženými z tohoto signálu, konkrétně s jeho intenzitou, s podílem jednotlivých frekvenčních

pásem (výška tónů v hudbě) a také s odhadem rytmu dané skladby. Na základě pohybu uživatele je pak celá perspektiva projekce ohýbána tak, aby docházelo k iluzi rozšíření prostoru sálu a propojení reálného světa s virtuálním.

Ideální využití vidím v kombinaci s živým vystoupením. Osobně bych rád vyzkoušel hru na nějaký sólový hudební nástroj napojený na tuto vizualizaci, zatímco bych se s ním mohl pohybovat po prostoru místnosti a kolem mě by reagovaly objekty na mnou generovaný zvuk. Kupříkladu pro hudebníka hrajícího na akordeon, elektrické housle či příčnou flétnu by mohl být zážitek vskutku unikátní. Sám však hraji na klavír, se kterým by takový výkon asi nebyl možný.

Fotografie ze samotné projekce generované aplikací jsou k vidění v příloze v závěru této práce. Vzhledem k povaze projektu, u kterého jde především o animaci, polohově závislou projekci a zpracování zvuku, jsou však zdaleka nejdůležitější videa z přiloženého DVD. Na těch je patrná souhra všech zde představených složek, které se na výsledné prezentaci podílejí a poskytují tak alespoň hrubou představu o iluzi, kterou projekce vytváří v prostorách instalace.

6.2 Rozbor implementace

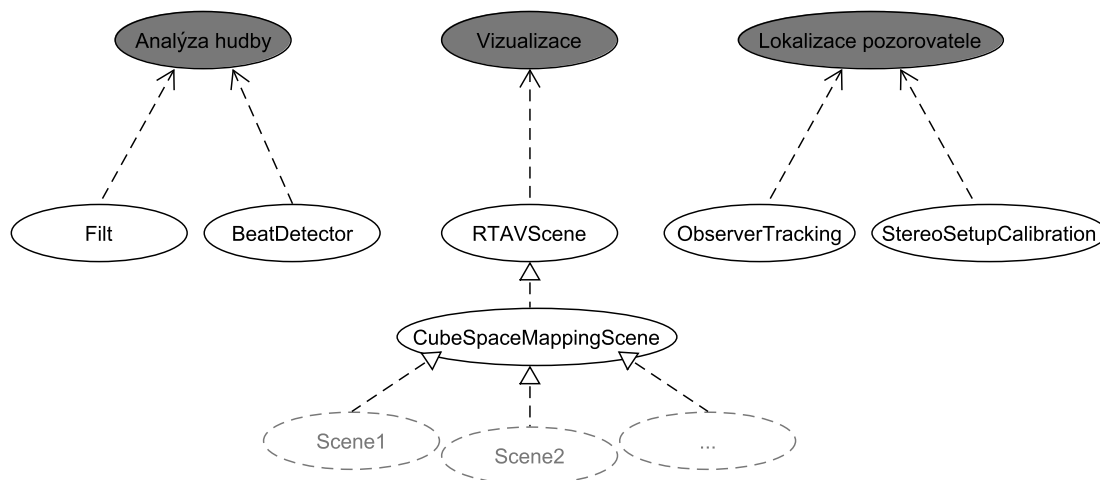
Po softwarové stránce byl projekt realizovaný v jazyce C++ a jako cílová platforma byl zvolen OS Windows. Až na přístup k využívaným kamerám, pomocí speciální knihovny pro ně určené, je však kód platformě nezávislý a zbylé využívané knihovny by měly být přenosné mezi všemi nejpoužívanějšími operačními systémy. Z externích nástrojů využívám knihoven OpenCV [25], FreeGlut [14], PortAudio [31] a CLEyeMulticam [21].

Systém je rozvržený do tříd obstarávajících nějakou klíčovou funkcionalitu, případně předepisujících rozhraní využívané pro předávání informací mezi nimi. Orientační znázornění těchto vztahů je pro ty nejdůležitější třídy znázorněno na obr. 6.2. Dělení se opět odráží od struktury projektu. Hlavní skupiny proto jsou třídy orientované na práci se zvukovým signálem, na rekonstrukci polohy diváka v prostoru a na výsledný grafický výstup. Mimo to projekt obsahuje mnoho mých vlastních pomocných struktur pro vektorové operace, práci s barvami a matematickými funkcemi obecně.

6.2.1 Zpracování zvuku

Srdcem pro analýzu akustického signálu je knihovna PortAudio. Ta abstrahuje práci se vstupním a výstupním zvukovým zařízením a poskytuje nástroje pro pohodlnou práci s ním bez potřeby znát specifika cílové platformy. Po iniciaci v kódu aplikace v pravidelných intervalech volá předanou funkci (*callback*). V té získáváme průběžně jednotlivé vzorky vstupního signálu, typicky 44100 vzorků za vteřinu.

Nad těmito vzorky provádím výpočet efektivní hodnoty signálu, tedy jeho relativní hlasitosti. Zároveň každý nový vzorek posílám k analýze třídě *Filt* implementující lineární filtry propouštějící pásma z předem vypočtených frekvenčních rozsahů – viz vztahy (3.4) a (3.6). V těchto vztazích však nepoužívám uvedené hodnoty pro spodní a horní hranici (20Hz až 20kHz), protože, ačkoliv tento rozsah můžeme slyšet, není sám o sobě tolik směrodatný, pokud na problematiku nahlédneme z pohledu hudby. Volím proto rozdělení frekvenčního spektra



Obrázek 6.2: Neformální diagram rozdělení hlavních tříd dle jejich funkcí

mezi 50Hz do 5kHz. Pomocí takto sestavených filtrů získám sedm nových, dílčích signálů, nad kterými také provádím výpočet RMS, abych získal poměr zastoupení různých frekvenčních pásem. Všechny tyto operace probíhají zvlášť pro levý a zvlášť pro pravý kanál.

Pro detekci přízvukných dob, respektive odhad rytmu v hudbě, slouží třída *BeatDetector*. Té předávám vzorky všech signálů získaných band-pass filtrací. Tyto vzorky si třída postupně ukládá stranou, dokud jich nemá tolik, na kolik je v ní určená velikost analyzovaného rámce, standardně 1024. Poté vypočítá intenzity jednotlivých signálů, zjistí jejich celkovou váženou intenzitu podle vztahu (3.11) a zaznamená do historie průběhu zvuku. Tuto hodnotu pak porovnává s předchozími a podle charakteristik rozebíraných v kapitole 3.3 určujeme, zda byl v posledním analyzovaném rámci detekován náběh pravděpodobné hlavní doby. Pokud ano, rozhodneme společně s tím o tom, na kolik je aktuálně detekovaný rytmus pravidelný. To probíhá na základě porovnávání intervalů několika posledních detekovaných rázů, respektive poměry mezi nimi.

6.2.2 Kalibrace kamer a sledování pozorovatele

Po spuštění se program zastaví v kalibrační části. Zobrazí se nabídka s různými kalibračními možnostmi. Při výběru některé z nich se spustí odpovídající procedura implementovaná v třídě pro to uzpůsobené – *StereoSetupCalibration*. Ta obstarává čtení a zápis do konfiguračních souborů spojených s nastavením kamer. Například při kalibraci vnějších parametrů systému kamer v jejich obrazech vyhledává kalibrační vzor – šachovnici o rozměrech specifikovaných v souboru *config_CameraCalibration*. Pokud jej najde v obrazech obou kamer ve stejném okamžiku (obr. 5.7), dopočte a zaznamená pozice klíčových bodů šachovnice. Na následující chvíli (přibližně vteřinu) zaznamenávání deaktivuje, aby mohl uživatel šachovnici přemístit do nové pozice. Po ukončení této procedury jsou zaznamenaná data předána k výpočtu hledaných parametrů pomocí funkcí knihovny OpenCV, ty jsou následně uloženy do speciálních souborů (*Cam1Calib.yml*, *Cam2Calib.yml*, *StereoCalib.yml*) pro budoucí použití. V případě procedury hledající vhodnou transformaci pro převod ze souřadného systému

první kamery do systému scény se využívá také funkcí implementovaných ve třídě *ObserverTracking*.

Primární funkcí třídy *ObserverTracking* je vyhledávání pozice markeru v obraze. Obsahuje vlastní proceduru, pomocí které lze upravit a uložit hodnoty využívané pro prahování obrazů kamer. Pokud je využívána se známými vnitřními parametry kamer, využije těchto informací a před samotným vyhledáváním ještě celý obraz upraví tak, aby se vykompenzovalo radiální a tangenciální zkreslení čoček (obr. 5.6). Pokud pak známe projekční matice P_1 a P_2 našeho systému kamer, umožní i rekonstrukci prostorové informace o poloze markeru ze získaných obrazů.

Získávání informací o pozici pozorovatele z obou kamer v průběhu vizualizace probíhá paralelně. Vyhledávání v obraze může při špatně zvolených hodnotách thresholdingu vlivem přítomnosti mnoha potenciálních kontur trvat relativně dlouhou dobu. Je proto lepší, když běh hlavní části, tedy zobrazování, není ovlivněn těmito prodlevami a animace tak mohou být plynulé. Každá kamera má proto přiřazené jedno výpočetní vlákno a z něho aktualizuje svou nejlepší rychlostí údaj o poloze markeru v jejím obraze.

6.2.3 Napojení na výslednou prezentaci

Při samotné vizualizaci je aktualizace scén řízena periodicky v přesně daných intervalech. V těchto intervalech se vždy dotážeme na aktuální údaje získané ze zvukového systému – na zastoupení různých frekvenčních pásem a na případnou přítomnost hlavní doby v daném okamžiku. Vzhledem k paralelnímu vyhodnocování detekce rázů v hudbě je pak každá taková doba hlášena v nejbližším následujícím snímku. Dokud nebyl proveden odpovídající dotaz pro vizualizaci, údaj o její přítomnosti není nulován.

Protokol definovaný pro scény, které mají být schopny reagovat na hudbu, je určený abstraktní třídou *RTAVScene* (*RealTimeAudioVisualizationScene*). Je v ní předepsána podoba funkcí volaných za účelem aktualizace scény, zpracování uživatelského vstupu, informování scény o přítomnosti či nepřítomnosti hlavní doby v hudbě a jejích intenzitách v jednotlivých pásmech. Potomkem této třídy je další abstraktní třída *CubeSpaceMappingScene*. Ta poskytuje základní funkcionalitu spojenou s prezentací v podobě projection mappingu realizovaného na prostoru podobném hale IIM, tedy navzájem kolmé stěny okolo pozorovatele. Zároveň poskytuje rozhraní pro aktualizace údajů o poloze diváka ve scéně. Při inicializaci instance některého z potomků načte údaje o fyzických rozměrech prostoru ze souboru *config_mappingDimensions*. Na základě těchto údajů a informací o poloze pozorovatele pak před vykreslením obrazu pro každý jednotlivý projektor upraví použité perspektivní zkreslení.

Samotnou podobu vizualizace, jejího průběhu a vykreslování zaopatřují jednotlivé *Scény*. Podoba implementovaných scén byla rozebírána v sekci 4.4.

6.3 Testování a hodnocení výstupu

6.3.1 Nasazení v prostorách Institutu Intermédií

Pro projekci v hale IIM využíváme třech ploch. Ty jsou na sebe kolmé a nachází se při předpokládané pozici pozorovatele po jeho stranách a před ním. Projektory jsou upevněny na

lešení nad využívaným prostorem. Pro zvukovou stránku prezentace se využívá velkých reproduktorů umístěných v rozích tohoto prostoru. Jako řídicí jednotku využíváme notebooku, do kterého je veškerá technika zapojená. Kvůli omezenému počtu portů pro grafický výstup na přenosných počítačích je třeba využít dodatečného zařízení, které umožní distribuce z jednoho výstupu na vstup všech třech projektorů. Tím je v našem případě TripleHead2Go společnosti Matrox [23].

Pro získávání údajů o poloze pozorovatele využíváme dvojice kamer Sony PlayStation Eye [34]. Volba jejich umístění byla asi nejproblematictější částí v průběhu instalace. Snažíme se najít takovou polohu, ze které bude mít pohled obou kamer co největší průnik, aby se uživatel mohl pohybovat po co největší ploše. Aby nebyla projekce rušena jejich přítomností, bylo by lepší umístit je za pozorovatele. Pak ovšem narazíme na problémy spojené s vyhledáváním markeru v obrazu, protože budeme neustále zabírat jasně osvětlené plochy, na kterých projekce probíhá. Ta v průběhu mění své barvy, jejich odstíny, saturaci i intenzitu. Abychom tomuto zabránili, bylo by nejlepší odstínit spektrum viditelného světla dodatečným filtrem u čočky kamer, a využívat markeru používajícího infračervené záření. Odstraněním IR filtru u používaných kamer však často vede k jejich špatnému ostření, což by velice znepráhnilo jejich kalibraci.

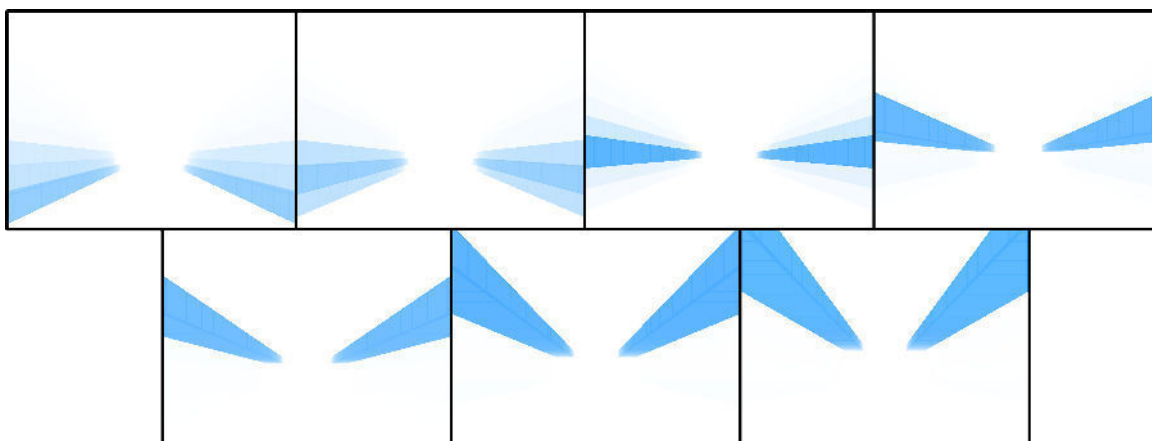
Kamery jsem proto umístil v přední části prostoru a natočil tak, aby přímo nezabíraly projekční plátna. Tím ovšem nastal obdobný problém. Vzhledem k tomu, že je obraz na plátno promítaný přední projekcí, je v obrazech kamer vidět přímo světlo vycházející z projektoru. Bylo proto třeba ještě kamery umístit do dostatečné výšky tak, aby je bylo možné natočit směrem dolů a světlo projektoru nenarušovalo vyhledávání markeru. Nejvhodnější by bylo kamery instalovat napevno nad prostor haly, v rámci práce jsme se však spokojili jen s přizvednutím kamer připevněním na polohovatelné stojánky (obr. 6.3).



Obrázek 6.3: Kamery používané pro rekonstrukci polohy pozorovatele

6.3.2 Obrazová reprezentace rysů zvukového signálu

Abych ověřil kvalitu zpracování zvukového signálu po stránce jeho rozkladu do sedmi frekvenčních pásem, generoval jsem na zvukový vstup jednoduchý signál. Ten měl vždy tvar sinu o jedné konkrétní frekvenci se stejnou amplitudou. Ve třetí vizualizační scéně jsem pak pozoroval, jak byla frekvence signálu vyhodnocena. Rozdělení přitom probíhalo pomocí band-pass filtrů, s ohledem na logaritmické vnímání výšky tónů, přibližně od 50Hz do 5kHz. Oproti plnému slyšitelnému spektru je to analyzované o něco zúžené, aby lépe zachycovalo zastoupení frekvencí přítomných v běžné hudbě.



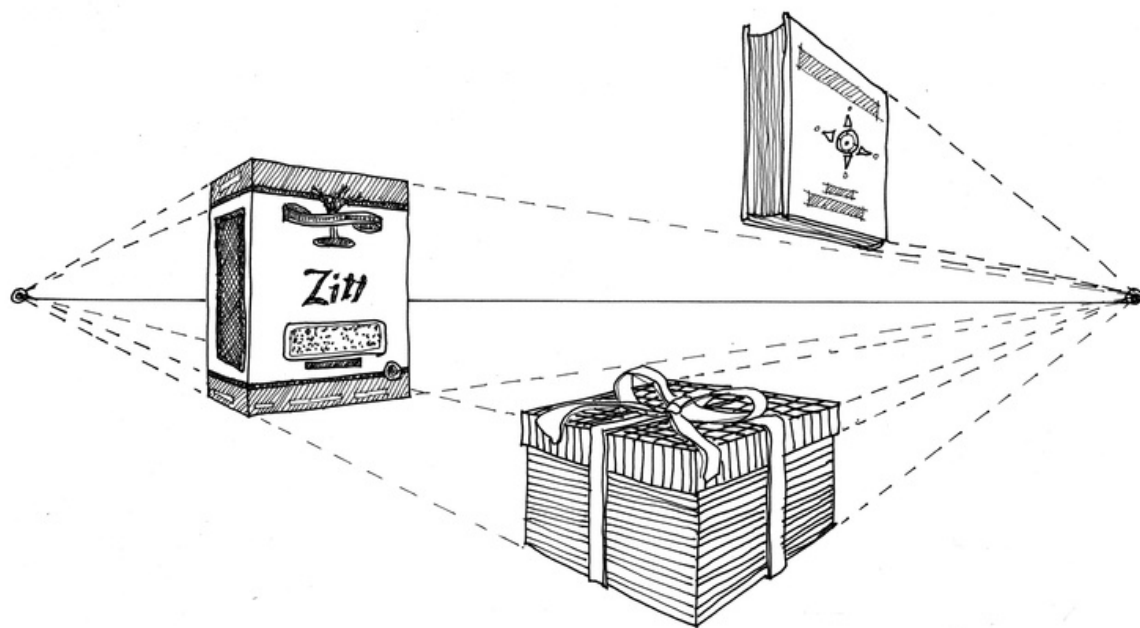
Obrázek 6.4: Znázornění rozkladu do frekvenčních pásem v průběhu přelaďování sinusového signálu. Vybrané frekvence: 75Hz, 150Hz, 300Hz, 600Hz, 1200Hz, 2400Hz a 4800Hz

Jak je vidět na obr. 6.4, výsledek odpovídá očekáváním, každá z testovaných frekvencí byla zaznamenána v jednom hlavním pásmu. Jen u nízkých frekvencí můžeme pozorovat ztrátu přednosti vyhodnocování. Příchozí signál se v těchto místech projevil i na sousedících pásmech a společně s tím došlo k úbytku intenzity vyhodnocené v „cílovém“ pásmu.

Testování kvalit v návaznosti na nalezení rytmu ve vizualizaci je problematičtější. Ne všichni lidé cítí hudbu stejně. A právě to, jak je hudba vnímána člověkem, je pro účely vizualizace důležitější než formální přesnost detekovaného metra. Zajímavým a informativním by proto mohl být experiment, který by vizuální a hudební složku postavil samostatně. Participantovi navrhovaného experimentu by byla nejdříve samostatně puštěna hudba. V jejím průběhu by bylo jeho úkolem mačkat nějaké tlačítko buďto do rytmu, nebo jednoduše jak sám průběh hudby cítí. Okamžiky stisku tlačítka by se zaznamenaly pro dodatečné vyhodnocení. V druhé části by naopak participant viděl jen obrazový projev vizualizace bez zvukového doprovodu. Jeho úkol by byl podobný. Zaznamenávat, kdy na něho vizualizace zapůsobila nějakým „zajímavým“ dojmem (například rozsvícení objektu v detekovaném rázu). Kvalita algoritmu, který je pro vizualizaci používán, by se pak mohla vyhodnocovat na základě korelace mezi okamžiky, které participant vyhodnotil jako nějakým způsobem výrazné. Zde počítám s tím, že je pro naše účely důležitější dojem, kterým výstup působí, než čistě přesnost detekovaného rytmu.

6.3.3 Věrohodnost projekce

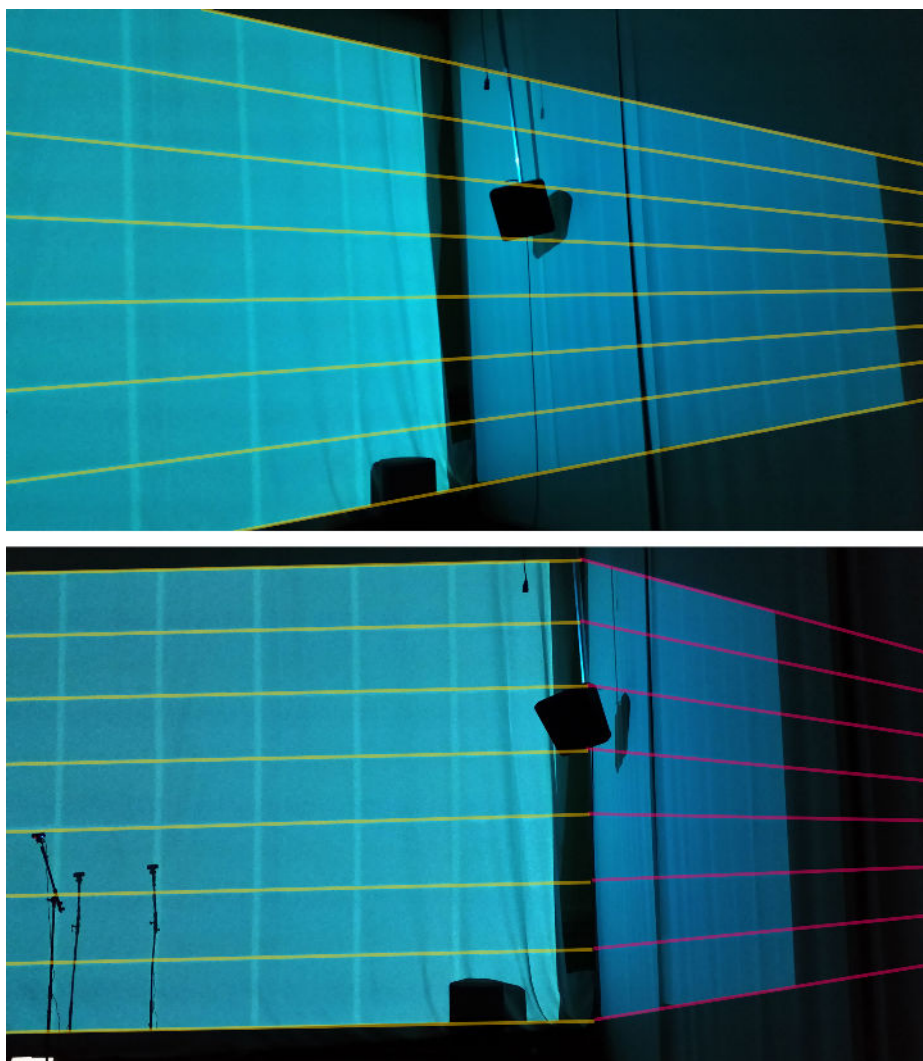
Kdybychom chtěli nějakým způsobem určit, do jaké míry je vytvářená iluze realistická, mohli bychom využít vlastností perspektivního vidění. Například máme-li ve scéně přímky, které jsou v jejím virtuálním prostoru rovnoběžné, musí se jejich obrazy, zachycené například fotoaparátem snímajícím celou projekci, sbíhat v jednom bodě, anglicky označovaném jako *vanishing point* (obr. 6.5).



Obrázek 6.5: Perspektivní kresba s protaženými rovnoběžnými hranami do společných úběžných bodů [12]

Druhý rys, ukazující zejména kvalitu kalibrace kamer a návaznosti promítaných obrazů, můžeme pozorovat na rozhraní mezi projekčními plochami. Předpokládejme, že se nacházíme v bodě, na který je projekce uzpůsobena. Pak musí všechny linie, které jsou v prostoru scény rovné a na sebe navazující, tvořit rovné navazující úsečky i v dvourozměrném prostoru kamerou zachyceného obrazu. Srovnání v případě správně nastavené perspektivy a pohledu mimo její střed ukazuje obr. 6.6.

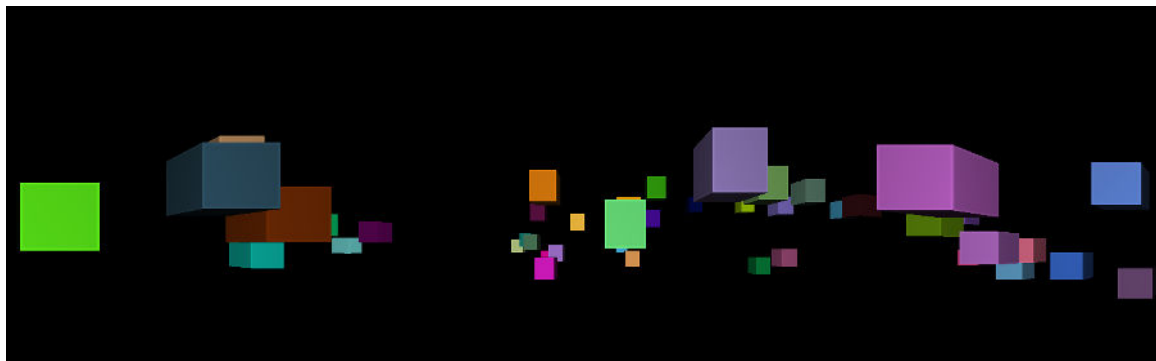
Tyto ukazatele jsou nezbytné pro rozhodování o kvalitě výstupu. Jelikož se ale jedná o instalaci zaměřenou na výsledný prostorový dojem vyvolaný v divákovi, je důležité zabývat se i působením na smysly reálných uživatelů. Bohužel, v daném prostoru není vlivem omezení rozebíraných ve čtvrté kapitole (velikost prostoru s ohledem na binokulární vidění, nedoléhání projekčních ploch, přílišná difuze světla do prostoru a podobně) výsledný efekt zcela přesvědčivý a podobná instalace by vyžadovala alespoň třikrát větší plochu. Poté by bylo možné zkoumat působení na základě testování s přizvanými participanty. V první řadě nejjednodušší dotazníkovou metodou hodnotící jednotlivé aspekty prezentace, v druhé pak speciálně sestavenými testy.



Obrázek 6.6: Sbíhání rovnoběžných linií a jejich návaznost na rozhraní kolmých projekčních ploch v případě rozdílného středu projekce

Nejprve by bylo vhodné realizovat s každým participantem test schopnosti odhadu vzdálenosti v reálném světě, například určit, v jaké vzdálenosti se nachází konkrétní objekt. Takto bychom získali orientační představu o schopnosti odhadu daného člověka. Následně bychom s ním přešli do prostor s instalovanou technikou pro rozebíraný projection mapping. Byla by mu představena virtuální scéna a v ní nějaký virtuální objekt. Uživatel by se mohl pohybovat po tomto prostoru, pak by měl rozhodnout, v jaké vzdálenosti je podle jeho odhadu daný objekt umístěný. Alternativou by mohla být scéna s více objekty různých velikostí, relativně náhodně rozmístěnými ve virtuálním prostoru kolem pozorovatele. Opět by měl příležitost pohybovat se po místnosti, tedy měnit perspektivu celé projekce. Poté by rozhodl o relativních pozicích mezi objekty – který je kolikrát dále či blíže než ostatní a podobně. Při ideálních podmínkách by měl dosahovat podobné přesnosti odhadu, kterou vykazoval při určování vzdálenosti v reálném světě. Ačkoliv přímo takové testování neproběhlo, protože by

v daných prostorách nemělo žádný přínos, do aplikace jsem scény vhodné pro navrhované testování v jejich základní podobě přesto implementoval. Jednu z nich lze vidět na obr. 6.7.



Obrázek 6.7: Scéna se statickými objekty, u kterých může participant případného testování určovat jejich relativní pozice na základě pohybu a rozhlížení se v prostoru

6.4 Diskuse k výstupu práce

Projection mapping je technika, která má speciální požadavky na prostor, ve kterém je realizována. I když překonáme její hlavní omezení v podobě potřeby sledování scény z jednoho konkrétního bodu v prostoru, stále v prostorách IIM nemůžeme zanedbat nedostatečnou vzdálenost od stěn místnosti, a tudíž lidskou schopnost stereoskopického vidění. Nabízí se proto možnost zapojit ještě další techniku, a sice stereoskopickou projekci [22]. Na každé plátno bychom pak museli promítat dva rozdílné obrazy – jeden pro levé a druhý pro pravé oko. Aby pak byla projekce přesvědčivá, museli bychom být schopni určit nejen pozici pozorovatele, ale přímo pozici jeho levého a pravého oka. Stačilo by alespoň určit pozici hlavy a její natočení v prostoru, podle čehož bychom už dokázali upravit odpovídající kamery ve virtuální scéně.

Vytvořený program poskytuje z pohledu projekce ucelený podklad pro libovolnou aplikaci. Ačkoliv v rámci realizovaného projektu byla vytvořena aplikace využívající techniky 3D mapování pro vizualizaci hudby, lze na stejném základu vytvořit libovolné scény a programy pracující s efektem hloubky. Zajímavým produktem by mohl být například program pro architekty, ve kterém by mohli vytvářet modely budov a sídlišť, následně by se pak mohli v části jimi vytvořeném prostoru procházet. Mohli by tak prohlížet jimi navržené budovy, ulice a zahrady ze stejné perspektivy, z jaké je pak uvidí za několik let po realizaci daného projektu. Tato technika jednoduše poskytne možnost vidět vše na vlastní oči včetně dojmu „majestátnosti“ celého navrhovaného prostranství.

V neposlední řadě se pak nabízí využít tohoto řešení pro počítačové hry. Vzhledem k volnému pohybu v prostoru by mohly být zajímavé hry pro více hráčů. V takovém případě by však bylo potřeba pro každého z nich promítat obraz s projekcí uzpůsobenou jejich umístění v prostorách místnosti. Toho by bylo možné docílit pomocí technik využívaných pro stereoskopickou projekci, tedy buďto s využitím polarizace promítaného obrazu a polarizovaných brýlí, nebo pomocí aktivních brýlí (*shutter glasses*) a synchronizace promítaného obrazu.

Již nyní je v aplikaci kompletně implementované řešení pro určování polohy markeru v rámci scény. Drobnou modifikací bychom mohli přidat schopnost rozlišování většího počtu markerů a následně sledovat například ruce uživatele. To by představilo celou řadu možností další interakce, jako třeba ovládání aplikací pomocí gest.

Pro alternativní aplikace však nemusíme využívat jen řešení realizovaného za cílem projekce. I nad částí implementovanou pro zpracování zvuku by bylo možné navrhnout multi-mediální výukové aplikace či hry, ve kterých se využívá rytmu v přehrávané hudbě, na který musí uživatel včas reagovat a podobně.

Kapitola 7

Závěr

Vizualizace hudby není na poli dnešních počítačů nijak vyjímečnou záležitostí. Zpravidla je ale napojena na přehrávání hudebních souborů a jen zřídka narazíme na plně automatický systém, který by byl schopný reagovat na hudební vystoupení v reálném čase. Mým cílem bylo právě vytvoření uživatelsky zajímavé instalace, která by jednak byla schopna reagovat na aktuální zvukový projev, ale také by obohacovala zážitek z hudebního poslechu o iluzi přítomnosti v zajímavém, na hudbu reagujícím světě. Zaměřil jsem se zejména na dojem vyvolávaný výslednou vizualizací a na jeho prostorovou uvěřitelnost.

Hudba je v aplikaci využívána jako hlavní faktor řídící průběh dění v několika virtuálních scénách. Ty byly koncipovány tak, aby demonstrovaly možnosti zvolené vizualizační techniky, tedy vzbuzovaly dojem, že prostor není ohraničen stěnami, popřípadě že se některé stěny táhnou za hranice fyzických rozměrů místnosti a podobně. Využívá se v nich jak výškových rysů hudby, tak její rytmické složky. Identifikace rytmických dob je v reálném čase doprovázena řadou komplikací a nelze tak zaručit stejnou přesnost, jako kdybychom analyzovali předem známé hudební dílo. Pro účely vizualizace za pomoci navrženého algoritmu však dokážeme dosáhnout přinejmenším zajímavých výsledků.

Z technik, které lze využít k docílení prostorového efektu, jsem s ohledem na možnosti dostupných prostor zvolil techniku *projection mapping*. Abych však eliminoval některá její omezení, doplnil jsem ji navíc o dodatečnou interakci v podobě úpravy perspektivního zobrazení na základě aktuální polohy pozorovatele. Tím je výsledný efekt uvěřitelnější, protože uživatel není nijak omezován v pohybu. Samotné určování polohy probíhá s využitím optického systému za použití dvojice kamer. Při správné kalibraci mohou být tyto kamery umístěny na libovolném místě v rámci sálu.

Vzhledem k omezením v prostorách, kde byla instalace realizována, nelze docílit zamýšlené iluze v plné míře. Na to bychom potřebovali prostor násobně větší. I tak se však nabízí možnost zkvalitnění výsledného efektu, která by mohla být implementována dodatečně. Největším nepřítelem *projection mappingu* na malém prostoru je binokulární vidění. Jako námět pro vylepšení v rámci prostor IIM se proto nabízí obohatit zmiňovanou techniku o stereoskopickou projekci. Nad rámec audiovizualizace pak přináší práce široké možnosti využití v alternativních multimediálních aplikacích. I při využití jen některých z použitých principů lze vytvořit jak užitečné programy, tak kupříkladu interaktivní prezentace či projekty orientované na zábavní průmysl.

Literatura

- [1] *Institut intermédií / Oficiální web IIM*. [online]. March 2015. Dostupné z: <<http://www.iim.cz/>>.
- [2] 4AV projectionmapping for Manpower Experis. <<http://www.projectionfreak.com/4av-projectionmapping-for-manpower-experis>>, 2014.
- [3] Epipolar geometry - Wikipedia, the free encyclopedia. <http://en.wikipedia.org/wiki/Epipolar_geometry>, .
- [4] 4AV projectionmapping for Manpower Experis. <<http://www.projectionfreak.com/4av-projectionmapping-for-manpower-experis>>, . (Visited on 04/27/2015).
- [5] Onset detection and Stroke Recognition for Indian Percussion instruments | Music Research, IIT Madras & CompMusic. <<http://www.iitm.ac.in/donlab/music/mridangam>>, .
- [6] "Viagem"Urbana| Artistas criam incríveis ilusões de ótica nas grandes cidades | Tudo Interessante | Curiosidades, Imagens e Vídeos interessantes. <<http://www.tudointeressante.com.br/2015/04/viagem-urbana-artistas-criam-incriveis-ilusoes-de-otica-nas-grandes-cidades.html>>, 4 2015.
- [7] GoPro | Bovine Aerospace. <<https://bovineaerospace.wordpress.com/tag/gopro/>>, . (Visited on 05/04/2015).
- [8] ANDERSON, B. L. Audio Filters. 2006. Dostupné z: <<http://www2.ece.ohio-state.edu/~anderson/Outreachfiles/AudioEqualizerPresentation.pdf>>.
- [9] BARGH, M. S. – GROOTE, R. Indoor localization based on response rate of bluetooth inquiries. In *Proceedings of the first ACM international workshop on Mobile entity localization and tracking in GPS-less environments*, s. 49–54. ACM, 2008.
- [10] Bruno Ribeiro. *Cyclique* [online]. October 2013. Dostupné z: <<http://www.nohista.org/?portfolio=cyclique>>.
- [11] BUDAK, S. Physics Extended Essay - A research about the effect of sound waves on standing waves by using Ruben's Tube. *TED Ankara College Foundation Private High School*. 2009. Dostupné z: <<http://tedprints.tedankara.k12.tr/41/1/2009-SeleneBudak.pdf>>.

- [12] COOK, J. Lecture – Painting plein air and Landcape basics | Landscape tips. <<https://janetcook.wordpress.com/2009/05/01/landscape-painting-in-florence/>>, may 2009. (Visited on 05/3/2015).
- [13] FLUSSER, J. Moment invariants in image analysis. 2006, 11, 2, s. 196–201.
- [14] FREEGLUT. *The Free OpenGL Utility Toolkit* [online]. April 2013. Dostupné z: <<http://freeglut.sourceforge.net/>>.
- [15] GAINZA, M. – COYLE, E. – LAWLOR, B. Onset detection using comb filters. In *Applications of Signal Processing to Audio and Acoustics, 2005. IEEE Workshop on*, s. 263–266. IEEE, 2005.
- [16] HANSEN, C. H. Fundamentals of acoustics. *Goelzer, B., H. Hansen, CH, Se-hrndt, GA (eds.) Occupational Exposure to Noise: Evaluation, Prevention and Control. World Health Organization, Geneva.* 2001. Dostupné z: <http://cdrwww.who.int/occupational_health/publications/noise1.pdf>.
- [17] HARTLEY, R. – ZISSERMAN, A. Multiple view geometry in computer vision. 2003.
- [18] HLAVÁČ, V. – PRAZE, Č. Fourierova transformace v 1D a 2D. *Přednáška, [cit. 10.11. 2010]. Dostupné z: <http://cmp.felk.cvut.cz/~hlavac/TeachPresCz/11DigZprObr/12FourierTxCz.pdf>.* 2012. Dostupné z: <http://cw.felk.cvut.cz/wiki/_media/courses/a4m33dzo/12fouriertxcz.pdf>.
- [19] JUSLIN, P. N. – VÄSTFJÄLL, D. Emotional responses to music: The need to consider underlying mechanisms. *Behavioral and brain sciences*. 2008, 31, 05, s. 559–575.
- [20] KOSUGI, N. Music Visualization based on Acoustic Data. 2010. Dostupné z: <<http://www.brl.ntt.co.jp/people/nao/doc/p607-Kosugi.pdf>>.
- [21] LABORATORIES, C. *CL PS3 Eye, month = jul, year = 2014, url = https://codelaboratories.com/home/, urldate = 2015-04-19* [online].
- [22] LAUBR, D. Stereoskopická projekce. *Praha: České vysoké učení v Praze, Fakulta elektrotechnická.* 2006.
- [23] Matrox. *Matrox Graphics - Solutions* [online]. December 2015. Dostupné z: <<http://www.matrox.com/>>.
- [24] Oculus VR. *Immersive virtual reality technology* [online]. December 2014. Dostupné z: <<https://www.oculus.com/>>.
- [25] OPENCV. *OpenCV (Open Source Computer Vision)* [online]. May 2015. Dostupné z: <<http://www.opencv.org/>>.
- [26] OPENGL. *OpenGL - The Industry Standard for High Performance Graphics* [online]. April 2015. Dostupné z: <<https://www.opengl.org/>>.
- [27] PAJDLA, T. Elements of Geometry for Computer Vision. April 2013, s. 95–105.

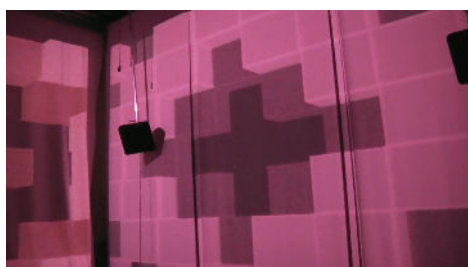
- [28] PATIN, F. Beat Detection Algorithms. 2003. Dostupné z: <<http://www.flipcode.com/misc/BeatDetectionAlgorithms.pdf>>.
- [29] Pur3 Ltd. Morphyre, 2011. Dostupné z: <<http://www.morphyre.com/>>. [počítačový program]. Verze 1.57. [cit. 2015-01-05].
- [30] PUZÓN, B. – KOSUGI, N. Extraction and visualization of the repetitive structure of music in acoustic data: misual project. In *Proceedings of the 13th International Conference on Information Integration and Web-based Applications and Services*, s. 152–159. ACM, 2011. Dostupné z: <http://www.brl.ntt.co.jp/people/nao/doc/iiWAS2011_107_Kosugi.pdf>.
- [31] Ross Bencina, Phil Burk. *Portaudio - Portable Cross-platform Audio I/O* [online]. January 2014. Dostupné z: <<http://www.portaudio.com/>>.
- [32] SCHMIDT-JONES, C. The basic elements of music. 2012.
- [33] SETTLES, G. S. et al. Full-scale schlieren flow visualization. *Flow Visualization VII. New York, NY: Begell House, Inc.* 1995. Dostupné z: <<http://www.mne.psu.edu/PSGDL/Pubs/1995-Settles-FV.pdf>>.
- [34] SONY. *Sony PlayStation Eye* [online]. 2012. Dostupné z: <<http://us.playstation.com/ps3/accessories/playstation-eye-camera-ps3.html>>.
- [35] Surge17. Frequency Domain, 2014. Dostupné z: <<http://steamcommunity.com/sharedfiles/filedetails/?id=142393387>>. [počítačový program]. Verze 0.5. [cit. 2015-01-05].
- [36] WATTENBERG, M. Arc diagrams: Visualizing structure in strings. In *Information Visualization, 2002. INFOVIS 2002. IEEE Symposium on*, s. 110–116. IEEE, 2002. Dostupné z: <<http://hint.fm/papers/arc-diagrams.pdf>>.
- [37] ZHANG, D. et al. Localization technologies for indoor human tracking. In *Future Information Technology (FutureTech), 2010 5th International Conference on*, s. 1–6. IEEE, 2010.

Příloha A

Popis ukázek projekce z přiloženého disku

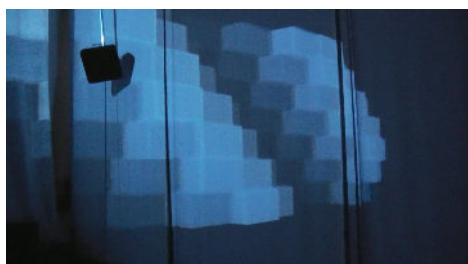
Zde je náhled a stručný popis videí z přiloženého DVD. Z důvodu pro kameru nepříznivých světelných podmínek, za kterých projekce probíhá, obsahují některá videa velké množství šumu. Zároveň nedostatek světla v kombinaci s dramatickými změnami jasu promítaných obrazů zhoršoval schopnosti ostření použité kamery. Vestavěný mikrofón kamery pak neumožňoval věrné zachycení zvukové stopy. Přiložená videa zde ale slouží primárně pro demonstraci prostorového efektu zvolené techniky projekce, který je patrný i přes omezenou kvalitu nahrávek.

am01.mp4

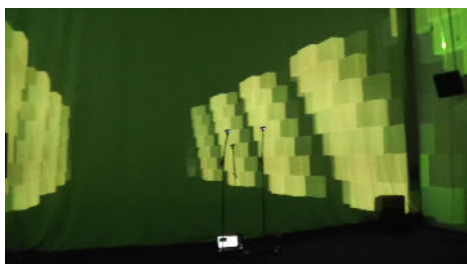


První scéna vizualizační aplikace v jejím demo režimu, demonstrující efekt kostek vystupujících z přední i postranních stěn.

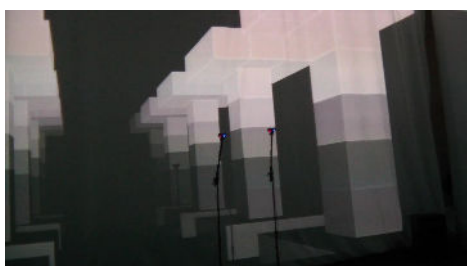
am02.mp4



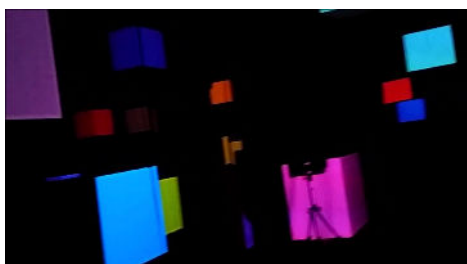
Druhá scéna aplikace v jejím demo režimu, evokující iluzi protažení přední stěny za hranice fyzicky dostupného prostoru.

am03.mp4

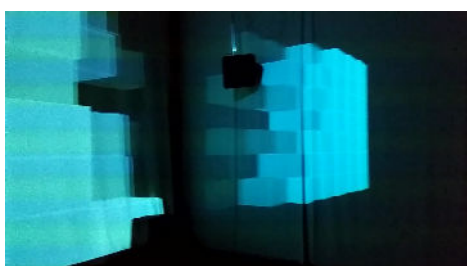
Třetí scéna v demo režimu, která má za cíl vyvolání dojmu, že postranní stěny pokračují do dále před pozorovatele.

am04.mp4

Demo ukázka čtvrté scény, která je tvořena vždy dvěma řadami sloupů, které pokračují za hranicemi všech projekčních ploch do dále.

am05.mp4

Statická scéna z kostkami rozmístěnými v prostoru, vhodná k navrhovaným experimentům pro ohodnocení úrovně dojmu hloubky.

am06.mp4

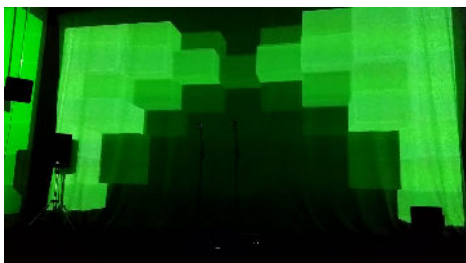
Audioreaktivní verze druhé scény v základní podobě.
--*Gotye, Somebody That I Used to Know*

am07.mp4



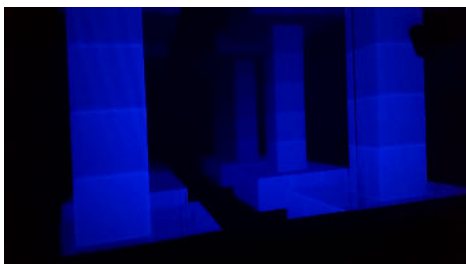
Třetí scéna reagující na hudbu, ve které jsou razantní změny dynamiky v krátkém čase
 --Queen, *We Will Rock You*

am08.mp4



Sekundární (invertovaná) varianta druhé scény s pozměněným barevným schématem. Na videu je patrná rozdílná míra saturace barev jednotlivých projektorů.
 --AC/DC, *T.N.T.*

am09.mp4

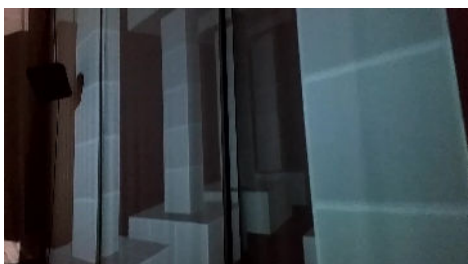


Scéna se sloupy reagujícími na rytmus hudby zpracovávané v reálném čase. Velice potemněné barevné ladění eliminuje *color bleeding* mezi plátny.
 --Amanda Lear, *Enigma*

am10.mp4

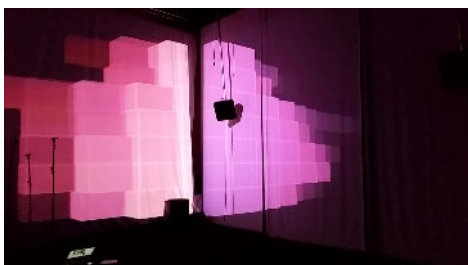
Čtvrtá scéna reagující na rytmus hudby. Je zde velice výrazná rozdílnost barevného projevu jednotlivých projektorů. Zajímavý je také začátek videa, kdy se aplikace pokouší najít správný rytmus

--*Fool's Garden, Lemon Tree*

am11.mp4

Obdoba dvou předchozích ukázek s tím rozdílem, že je v aplikaci deaktivované vyhledávání pozorovatele v rámci scény. Perspektivní zkruslení tak není upravováno a je zde vidět, jak se dojem hloubky s rostoucí vzdáleností od středu místnosti vytrácí.

-- *Vladimír Šimůnek, Mafia*

am12.mp4

Invertovaná varianta druhé scény, ve které je navíc aktivována detekce rytmu v hudbě. Hlavní doby jsou pak znázorňovány dočasným zvýšením saturace barev kvádrů tvořících virtuální stěnu.

--*Journey, Separate Ways*

am13.mp4



Stejné nastavení jako v předchozím případě, jen s jiným barevným tématem. Zajímavý je zejména pohled na projekci v blízkosti přední stěny.

--Mark Mueller, *Duck Tales*

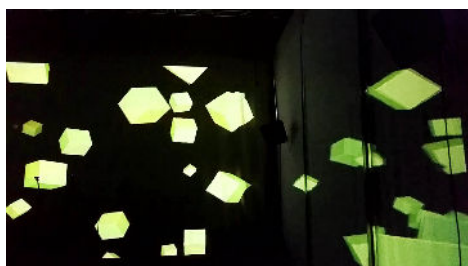
am14.mp4



Pátá scéna vizualizace, ve které reagují krychle ve virtuálním prostoru okolo diváka na rytmus hudby změnou rychlosti přehrávání jejich animace, společně se zvýrazněním jejich barvy.

--Journey, *Separate Ways*

am15.mp4



Stejná konfigurace scény jako v předchozím videu. Cílem byla demonstrace algoritmu detekce rytmu na klasické hudbě. Ta bývá pro tuto úlohu problematická, na některých dílech však může detekce přesto fungovat dobře.

--Wolfgang Amadeus Mozart, *Piano Concerto No. 21, Andante*

Příloha B

Obsah přiloženého DVD

proj	
├ bin.....	samotná aplikace a potřebné pomocné soubory
├ doc	stručná dokumentace hlavních struktur kódu
├ src.....	složka projektu se zdrojovými kódy
├ CLEyeDriver.exe	ovladač pro komunikaci aplikace s kamerami
└ readme.txt	rozsáhlá příručka k instalaci a užívání aplikace
text	
└ dp_tyc_ladislav_2015.pdf	text práce ve formátu PDF
vids	video z projekce realizované v IIM