

Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra informatiky a výpočetní techniky

Bakalářská práce

System pro vizuální záznam pomalých dějů

Prohlášení

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně a výhradně s použitím citovaných pramenů.

V Plzni dne 19. července 2012

Pavel Kocábek

Poděkování

Tímto bych rád poděkoval Ing. Kamilu Ekšteinovi, Ph.D. za vedení této práce a za zapůjčení zařízení. Za zapůjčení zařízení děkuji také BcA. Vojtěchu Aubrechtovi a Janu Džuganovi.

Abstract

The goal of this work is to acquaint the reader with methods of capturing time lapse frames. Second part of this work is to implement a custom software which would make an alternative solution to the existing software bundles. This part should inform the reader about the methods and algorithms used in the designed software solution and explain the usage of the created application.

Obsah

1	Úvod	1
2	O časoběrném snímání	2
3	Existující software	6
3.1	AnimatorHD	6
3.2	HandyAvi 4.3	6
3.3	CandyLabs VideoVelocity	7
3.4	GBTimelapse	7
3.5	DIYPhotobits.com Camera Control	8
3.6	Tilaphos	8
3.7	Perios	8
3.8	Srovnání	9
4	Techniky úpravy obrazu	10
4.1	Základní pojmy	10
4.1.1	Barevný model RGB	10
4.1.2	Histogram	10
4.2	Stereoskopie	11
4.2.1	Paralaxa	12
4.3	Úpravy obrazu v programu FlyCatcher	12
4.3.1	Ořez	12
4.3.2	Korekce barev a expozice	12
4.4	Detekce změn	15
4.4.1	Gaussovo rozostření a konvoluce	15
5	Programová dokumentace	17
5.1	Aplikační logika	17
5.1.1	CivilGrabber	17
5.1.2	WiaGrabber	18
5.1.3	NetworkGrabber	21

5.1.4	Řízení chodu programu	21
5.1.5	Manipulace s obrazem	22
5.1.6	Sít'ová komunikace	23
5.2	Grafické uživatelské rozhraní	24
5.2.1	Window	24
5.2.2	Průvodce	25
5.2.3	Panely oddělené od AbstractPanel	26
5.2.4	Plátno VideoCanvas	27
5.2.5	Ostatní okna a dialogy	27
5.2.6	Lokalizace	28
6	Závěr	29
A	Uživatelská dokumentace	34
A.1	Požadavky	34
A.2	Překlad a spuštění programu	34
A.2.1	Překlad programu	34
A.2.2	Spuštění programu	34
A.3	Obsluha programu	35
A.3.1	Hlavní okno	35
A.3.2	Snímání	35
A.3.3	Náhled videa	37
A.3.4	Časování	38
A.3.5	Možnosti ukládání	39

1 Úvod

První částí této práce je seznámit čtenáře s metodami časosběrného snímání a existujícím softwarem pro tvorbu časosběrných fotografií a videa.

Cílem druhé části práce je navrhnout a vytvořit systém pro časosběrné vizuální snímání pomalých dějů pomocí digitálních videokamer či fotoaparátů, který bude alternativou dostupných programů, zabývajících se touto tematikou.

Výsledný program by měl umět zpracovávat snímky z různých zařízení připojených k počítači, jako jsou např. webové kamery či digitální zrcadlovky. Část práce je také věnována stereoskopickému snímání ze dvou zařízení najednou.

2 O časoběrném snímání

Obsahem této kapitoly je seznámit čtenáře s principy časoběrného snímání a poskytnout mu tak základní přehled o metodách získávání takového záznamu.

Časoběrné snímání je hojně používaná fotografická a filmařská technika, spočívající ve snímání scény nižší rychlostí, než je tomu u klasického videa. Zaznamenané video se zrychlí na běžnou snímkovou frekvenci videa, typicky 25 nebo 30 snímků za sekundu. Snímkovým rychlostem se více věnuje zdroj[5].

Na první pohled se jako nejjednodušší metoda jeví záznam videa z videokamery v reálném čase či ukládání všech jednotlivých snímků ze zařízení. Výhodou této metody je, že se uživatel nemusí starat o snímací frekvenci a rychlost výsledné videosekvence určí až ve video editoru. Nevýhodou je ovšem obrovské množství souborů, které se na disk musí v průběhu snímání uložit, a jejich velikost. Druhá, smysluplnější metoda, je ukládat pouze ty snímky, které uživatel ve svém videu využije. Jednotlivé snímky se tedy budou ukládat v předem určených intervalech. Tímto způsobem se uloží minimální počet snímků a sníží se zatížení procesoru a disku. Při použití této metody je rovněž možno ukládat pouze takové snímky, které se liší od předchozích.

Jaké intervaly je tedy vhodné použít? Záleží na situaci, kterou chceme snímat. Existují dva různé pohledy, podle kterých snímací interval určit, za předpokladu, že předem známe snímkovou frekvenci budoucího videa (FPS):

1. *Chceme video zrychlit oproti skutečnosti v určitém poměru (např. 10:1):* interval I budeme volit jako poměr snímkové frekvence videa ku požadovanému zrychlení, čili $I = \frac{FPS}{zrychlení} = \frac{25}{10} = 2,5$ s.
2. *Chceme video požadované délky (např. 24 hodin chceme napasovat do 5ti-minutového videa):* interval I vypočteme podle následujícího vzorce:
$$I = \frac{\text{reálná doba}}{\text{požadovaná délka} \cdot FPS} = \frac{86400}{300 \cdot 25} = 11,52$$
 s

Způsoby snímání

Některá zařízení mají vestavěnou podporu pro časoběrné snímání již ve svém firmware. K drtivé většině digitálních zrcadlovek lze dokoupit externí zařízení

– tzv. intervalometr. To je zařízení, které v pravidelných intervalech dává pokyn zařízení pro exponování snímku. Třetí možností je použít pro časosběrné snímání externí program. Externí programy často nabízejí oproti předchozím dvěma možnostem různé speciální funkce a široké možnosti nastavení.

Umístění zařízení

Při snímání musí být zařízení nehybné. Jakýkoliv sebemenší pohyb znamená znehodnocení výsledku. Ideální je zařízení umístit na těžký a robustní stativ. Středový sloupek zásadně nevysouváme, jednotlivé nohy stativu necháváme pokud možno také nevysunutý. Pro získání maximální stability je vhodné stativ zatížit zavěšením těžkého břemene za středový háček.

Expozice a filtry

Je známo, že expozice závisí na třech faktorech: na clonovém čísle, čase závěrky a citlivosti ISO. S vyšším clonovým číslem roste hloubka ostrosti snímku, s vyšší citlivostí šum a s delším časem pohybová neostrost. Právě pohybová neostrost tvoří časosběrné video plynulejším. Je tedy vhodné přicloubit a zvýšit dobu expozice i na několik sekund (samozřejmě v závislosti na snímané scéně). K dosažení tohoto efektu nám může pomoci neutrální šedý filtr, který omezí světlo procházející objektivem a zachová barevné podání snímku. Je nutno počítat s tím, že součet doby expozice a času, potřebného ke zpracování a uložení snímku, nesmí přesáhnout délku našeho snímacího intervalu.

Při snímání krajiny je vhodné použít polarizační filtr, který polarizuje jím procházející světlo do kolmé roviny vůči objektivu. Pomocí polarizačního filtru lze omezit dopadající polarizované světlo z oblohy, čímž dojde k jejímu ztmavení. Mějme však na paměti, že míra tohoto efektu závisí na úhlu slunečního svitu vůči objektivu, a proto bychom se polarizačnímu filtru měli vyhnout v případě, že plánujeme scénu snímat delší dobu, při které se slunce po obloze posune o velkou vzdálenost. V takovém případě je lepší oblohu ztmavit pomocí filtru přechodového.

Blikání

Při časosběrném snímání je silně doporučeno nastavit zařízení na plně manuální režim. Toto doporučení neplatí pro snímání scény, v níž chceme zaznamenat proces, při němž dochází k rapidní změně světla ve scéně. Například kdybychom při snímání soumraku použili manuální režim, světlo ve scéně by postupem času pokleslo natolik, že by výsledný snímek byl úplně černý. Pokud to není záměr, je třeba tento úbytek světla kompenzovat např. přepnutím zařízení do poloautomatického či plně automatického módu. Pokud nastavíme zařízení do režimu priority času, expoziční automatika bude dopočítávat vhodnou clonu tak, aby snímek byl korektně exponován. Zde ovšem nastává problém. Pokud zařízení vyhodnotí, že by scéna byla díky úbytku světla podexponována, změní clonové číslo na nižší (typicky o jeden krok). Po této změně projde za stejnou dobu objektivem dvojnásobné množství světla, čímž se výsledný snímek bude jevit jako světlejší. Protože clonové číslo nelze nastavit spojitě, přičemž by se clona nastavila adekvátně pro každý snímek, dochází k jeho nastavení nepravidelně skokově, a proto se nám sekvence snímků jeví jako blikající. Analogicky dochází ke skokové změně expozice při použití priority clony či plně automatického režimu.

K blikání může rovněž dojít, pokud použijeme automatické vyvážení bílé. Proto je nutno vyvážení bílé nastavit manuálně.

Nicméně, k mírnému blikání může docházet i při použití manuálního nastavení digitálních zrcadlovek, tvrdí [6]. Clona v objektivěch, určených pro digitální zrcadlovky, se po exponování snímku zcela otevře, než se opět nastaví na požadovanou hodnotu. Tímto způsobem mohou vznikat malé mechanické nepřesnosti, které rovněž způsobí blikání obrazu. Pokud budeme clonit málo, tyto nepřesnosti budou zanedbatelné, ale budeme muset použít kratší expoziční časy, které mohou být také nepřesné. Jiné řešení je objektiv pootočit tak, abychom odpojili elektrické kontakty u objektivu od těla zrcadlovky a clona tak zůstala konstantní.

V případě snímání v (polo)automatickém režimu je třeba k potlačení blikání použít post-processing¹.

¹Úprava snímků mimo zařízení, kterým bylo pořízeno.

Využití

Časosběrné video lze například uplatnit v následujících oblastech:

- snímání růstu rostlin,
- snímání oblačnosti,
- záznam jízdy dopravními prostředky,
- monitorování dopravy,
- záznam průběhu stavby či demolice budov,
- zamrzání ploch vodních toků.

3 Existující software

Cílem této kapitoly je seznámit čtenáře s existujícím softwarem používaným pro časoběrné snímání pomalých dějů, ukázat jejich funkce a provést srovnání s navrhovaným programem.

Existujících programů, určených k časoběrnému snímání, je celá řada. Uvedeny budou pouze některé z nich.

3.1 AnimatorHD

AnimatorHD je profesionální nástroj, sloužící k vytváření Stop Motion¹ animací a k časoběrnému snímání.

Program umí zpracovávat snímky z webových kamer a digitálních zrcadlovek od firmy Canon a Nikon. Umí zpracovávat fotografie i z fotoaparátů ostatních značek, ale pouze s použitím softwaru třetích stran. Dokáže ukládat do různých formátů, umí vytvořit i HDR² záznam a umí zobrazit živý náhled. Dovede snímat z více zařízení současně, komunikovat s pojezdovými zařízeními přes sériový port, náhled a export snímků ve anaglyfovém formátu, ale pouze od verze Professional. Z uložených snímků je schopen zpracovat videosekvence.

Pro vyzkoušení je zdarma dostupná demoverze, umožňující uložit ze zařízení 50 snímků s logem softwaru.

Více informací je dostupno na <http://animatordv.com>.

3.2 HandyAvi 4.3

HandyAvi je software, sloužící ke snímání a editaci časoběrných videí z webových kamer a televizních tunerů. Podporuje detekci pohybu, blesků a mete-

¹Technika spočívající ve fotografování nehybné scény, ve které jsou jednotlivé objekty manuálně upravovány a posouvány o malé úseky tak, aby se po spojení snímků do videosekvence navodilo dojem pohybu.

²High Dynamic Range

oritů. Program dovede ovládat dalekohledy firmy Meade Instruments.

Pro vyzkoušení je k dispozici 10denní verze zdarma.

Více informací je dostupno z <http://www.azcendant.com>.

3.3 CandyLabs VideoVelocity

VideoVelocity je jednoduchý program, zpracovávající snímky z webových kamer.

Program umí ukládat jednotlivé snímky či rovnou generovat video, může být ovládán z příkazové řádky a tím být plně automatizován. Umožňuje také snímat v HD³ a podporuje Motion triggered snímání, kdy se snímek uloží pouze za předpokladu, že se liší od předchozího. Za tyto funkce si ale již musíme připlatit.

Web produktu je na <http://www.candylabs.com/videovelocity>.

3.4 GBTimelapse

GBTimelapse je software, umožňující časosběrné snímání ze zrcadlovek firmy Canon.

Program dovede snímat v různých formátech, dokáže snímat více snímků s různou expoziční hodnotou a sestavit HDR snímky. Disponuje živým náhledem snímků a interaktivním náhledem videa. Má rozsáhlé nastavení časování – dokáže např. spustit nahrávání v určitý čas před západem slunce, apod. Ve verzi Pro umí synchronně snímat z více zařízení najednou.

Výrobce vyvíjí také nástroj GBDeflicker, sloužící k odstranění blikání obrazu, dostupný samostatně nebo jako plugin do produktů firmy Adobe.

Na internetu jsou dostupné zkušební verze všech výše uvedených produktů.

³Video s rozlišením vyšším než 800 × 600 bodů.

Více informací je k dispozici na
<http://www.granitebaysoftware.com/Products/ProductGBTEOS.aspx>.

3.5 DIYPhotobits.com Camera Control

Jedná se o skript, umožňující snímání fotografií ze zrcadlovek firmy Nikon. Umožňuje nastavit parametry fotoaparátu jako je např. clona, čas závěrky, ISO, atd. a má vestavěnou podporu pro časosběrné snímání.

Skript je lze stáhnout na stránce
<http://www.diyphotobits.com/download-diyphotobitscom-camera-control>.

3.6 Tilaphos

Tilaphos je jednoduchý volně dostupný nástroj pro zpracování snímků z webových kamer. Ukládá jednotlivé snímky do pevně definované složky. Z těchto snímků dokáže vytvořit video.

Program je možno stáhnout z <http://tilaphos.sourceforge.net>.

3.7 Perios

Perios je volně dostupný program, zpracovávající snímky z webových kamer. Rovněž dovede pořizovat snímky z kamer přes protokol HTTP. Z uložených snímků umožňuje vytvořit videosekvenci. Program obsahuje pouze základní nastavení a není už vyvíjen.

Program je k dispozici na <http://perios.ic.cz>.

3.8 Srovnání

Jak je vidět v předchozích odstavcích, většina kvalitnějších programů je placená. Volně dostupné programy a volně dostupné verze placených programů disponují pouze jednoduchými funkcemi. Navrhovaný program Fly-Catcher nabízí k těmto programům alternativu. Srovnání jednotlivých programů podle podpory zařízení je provedeno v tabulce 3.1 a jejich vybrané vlastnosti v tabulce 3.2.

Program	Webkamery	Canon	Nikon	Ostatní foto	Jiná zař.
AnimatorHD	Ano	Ano	Ano	Ano*	Ano
HandyAvi	Ano	Ne	Ne	Ne	Ano
GBTimelapse	Ne	Ano	Ne	Ne	Ne
DIYPhotobits	Ne	Ne	Ano	Ne	Ne
VideoVelocity	Ano	Ne	Ne	Ne	Ne
Tilaphos	Ano	Ne	Ne	Ne	Ne
Perios	Ano	Ne	Ne	Ne	Ano
<i>FlyCatcher</i>	<i>Ano</i>	<i>Ne</i>	<i>Ano</i>	<i>Ne</i>	<i>Ano</i>

Tabulka 3.1: Možnosti připojení zařízení: Tabulka ukazuje, která zařízení jsou podporována jednotlivými programy.

* Pouze snímá obsah určitého adresáře

Program	Detekce pohybu	Stereoskopie	Korekce barev
AnimatorHD	Ne	Ano	Částečně
HandyAvi	Ano	Ne	Ne
GBTimelapse	Ne	Ne	Ano*
DIYPhotobits	Ne	Ne	Ne
VideoVelocity	Ano	Ne	Ne
Tilaphos	Ne	Ne	Ne
Perios	Ne	Ne	Ne
<i>FlyCatcher</i>	<i>Ano**</i>	<i>Ano</i>	<i>Ano**</i>

Tabulka 3.2: Speciální funkce programů: ukázka vybraných nadstandardních funkcí jednotlivých programů.

* s programem GBDeflicker

** neplatí pro zrcadlovky

4 Techniky úpravy obrazu

4.1 Základní pojmy

4.1.1 Barevný model RGB

Barevný model RGB¹ patří v počítačové grafice mezi nejpoužívanější způsoby uložení barevné informace. Jeho jednotlivé barevné složky R, G a B jsou udávány procentuálně v rozsahu $\langle 0, 1 \rangle$ nebo pomocí tzv. bitové hloubky. Bitová hloubka obsahuje typicky 8 bitů na složku, kde bývá reprezentována hodnotami $[0 - 255]$ a model tudíž dokáže zobrazit 256^3 různých barevných kombinací. Model RGB je aditivní, tzn. čím vyšší je hodnota jednotlivých složek, tím světlejší je výsledná barva.

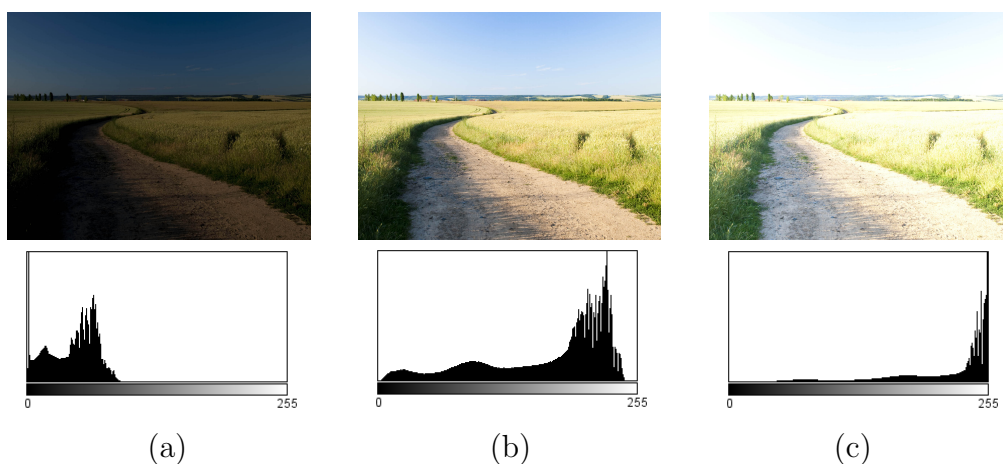
4.1.2 Histogram

Histogram je grafické znázornění distribuce jasu v obrázku. Nejčastěji bývá reprezentován grafem, v jehož x-ové ose je vynesena jas a v jeho y-ové ose relativní počet pixelů odpovídajících příslušnému jas. Dokážeme z něj určit, zda byl snímek správně exponován a jeho vlastnosti, mezi které patří např. jas a kontrast. Ukázky závislosti histogramů na nastavení expozice nalezneme na obr. 4.1.

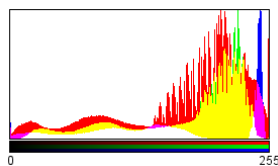
RGB histogram

RGB histogram je histogram, ve kterém je každá barevná složka obrázku vynesena zvlášť, viz obr. 4.2.

¹barevné složky Red (Červená), Green (Zelená) a Blue (Modrá)



Obrázek 4.1: Ukázky histogramů: (a) podexponovaný snímek; (b) správně exponovaný snímek; (c) přexponovaný snímek



Obrázek 4.2: RGB histogram snímku (b) z obr. 4.1

4.2 Stereoskopie

„Slovo stereo pochází z řečtiny a má význam prostorový. Stereoskopie se pak zabývá prostorovým zobrazováním. Tedy tvorbě a prohlížení obrazů, které obsahují informaci pro obě oči, díky níž si vytvoříme vjem hloubky prostoru.“
[7, s. 5]

Protože navrhovaná aplikace umožňuje stereoskopické snímání ze dvou zařízení, uveďme si nyní pojem paralaxy, který ke stereoskopickému snímání neodmyslitelně patří a na nějž bude později odkazováno.

4.2.1 Paralaxa

Paralaxou se rozumí úhel, který svírají přímky, vedoucí ze středu objektivů snímacích zařízení do jednoho bodu v prostoru, říká [12]. Tento bod bude na snímcích posunut o vektor úměrný paralaxe.

Paralaxu rozdělujeme na horizontální a vertikální. Horizontální paralaxa navozuje prostorový dojem, kdežto vertikální jej může rozbít. Pro odstranění vertikální paralaxy je nutno zařízení umístit do stejné roviny tak, aby přímky kolmé k plochám senzorů zařízení byly rovnoběžné. Více se této problematice věnuje zdroj [7].

4.3 Úpravy obrazu v programu FlyCatcher

4.3.1 Ořez

„Oříznutí odstraňuje okrajové části obrazu, z kvalitňuje rámování a kompozici, zvýrazňuje předměty nebo mění poměr stran.“ [13]

Kromě výše uvedených úkonů můžeme ořez provést při stereoskopickém snímání k částečnému odstranění vertikální paralaxy.

4.3.2 Korekce barev a expozice

Při snímání scény, v níž se mění světelné podmínky (např. východ a západ slunce, jízda autem), zařízením s automatickým nastavením expozice a vyvážením bílé (typicky webové kamery) může nastat jev, při kterém se skokově změní světelnost nebo barevný tón dvou po sobě jdoucích snímků. Pro odstranění skokové změny expozice slouží algoritmus přizpůsobení histogramu, který popisuje [4].

Mějme dva obrázky, referenční a upravovaný. Z histogramů obou obrázků vypočteme distribuční funkce F_R a F_U . Pro každou jasovou hodnotu $f_1 \in [0 - 255]$ nalezneme odpovídající jasovou hodnotu f_2 takovou, že $F_R(f_1) = F_U(f_2)$. Získanou mapovací funkci $f_2 \rightarrow f_1$ aplikujeme na každý pixel upravovaného snímku. [9]

Algoritmus v této podobě umí pouze ovlivňovat jas upravovaného obrázku. Aby dovedl opravit i barevný posun, musí se aplikovat mapovací funkce $f_2 \rightarrow f_1$ pro každou barevnou složku zvlášť. To vyžaduje výpočet histogramu a jeho distribuční funkce zvlášť pro červenou, zelenou a modrou složku. Výsledky algoritmu lze vidět na obr. 4.3.



(a)



(b)



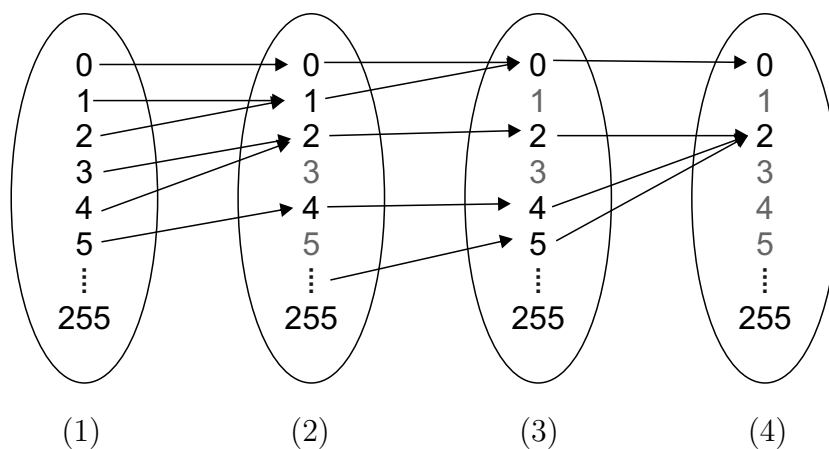
(c)

Obrázek 4.3: Korekce barev a expozice: (a) referenční snímek; (b) následující snímek; (c) opravený následující snímek

Sekvence snímků

Výše uvedený algoritmus se zabývá korekcí jednoho obrázku podle referenčního. Při snímání více snímků se ovšem musíme zamyslet nad tím, podle kterého snímku se budou další snímky opravovat.

Nabízí se dvě možnosti, každá ovšem podává jiné výsledky:



Obrázek 4.4: Zřetěžené mapování: (1) původní referenční snímek; (2) opravený snímek podle (1) a referenční snímek pro (3); apod.

1. *První snímek jako referenční:* Tento způsob podává uspokojivé výsledky za předpokladu, že se barevnost a světelnost scény nemění drasticky. Pokud se scéna významně mění, může docházet k redukci dynamického kontrastu upravovaných obrázků.
2. *Předchozí snímek jako referenční:* Pokud každý předchozí opravený snímek použijeme jako referenční, každý následující snímek bude disponovat menším nebo rovným dynamickým rozsahem, jako snímek referenční. To vyplývá z Dirichletova principu, který popisuje zdroj [10]. Každá nenulová hodnota upravovaného histogramu je namapována na jedinou hodnotu. Pokud tato jediná hodnota bude stejná pro více nenulových hodnot upravovaného histogramu, dojde v obrázku ke snížení počtu barev a tím pádem dynamického kontrastu, viz obr. 4.4.

Protože u způsobu č. 1 dochází k menší a netrvalé ztrátě informace, na rozdíl od druhého způsobu, je tedy lepší použít první snímek jako referenční. Abychom eliminovali i tuto ztrátu, je třeba informace do histogramu postupně doplňovat. Velmi dobré výsledky podává metoda, při níž se do referenčního histogramu přidá část histogramu nejnovějšího neupraveného obrázku. Nový referenční histogram bude tedy váženým průměrem původního histogramu a histogramu neupraveného nejnovějšího obrázku. Velmi dobrých výsledků dosáhneme, pokud váha původního referenčního histogramu bude činit 95 %. Při nižší váze může již být patrná skoková změna expozice, při vyšší zase dočasná ztráta dynamického rozsahu snímků.

4.4 Detekce změn

Pro vyhodnocení míry shody dvou obrázků používám následující algoritmus: Na obrázek se nejprve aplikuje Gaussovo rozostření, které odstraní šum z obrázku a poté se obrázek pixelizuje a změní na velikost 16 x 16 pixelů. Jednotlivé barevné složky každého pixelu se od sebe odečtou a jejich absolutní hodnoty $A_R(x, y)$, $A_G(x, y)$ a $A_B(x, y)$ se zpracují podle následujícího vzorce:

$$S = \frac{\sum_{x=0}^{15} \sum_{y=0}^{15} A_R(x, y) + A_G(x, y) + A_B(x, y)}{256 \cdot 3 \cdot 2.55}$$

Číslo S značí míru shody a pohybuje se v rozmezí od 0 do 100. Nyní se podíváme podrobněji na Gaussovo rozostření.

4.4.1 Gaussovo rozostření a konvoluce

Gaussovo rozostření se počítá pomocí tzv. diskrétní konvoluce. Konvoluce je binární matematický operátor, značený symbolem $*$ a definovaný následujícím vzorcem:

$$g(i, j) = \sum_{k, l} f(i + k, j + l)h(k, l)$$

Funkce f představuje původní obrázek, g je nově vypočtená světelnost pixelu na souřadnicích (i, j) a h je tzv. konvoluční jádro. Předchozí vztah lze zapsat pomocí zkrácené notace $g = f * h$, uvádí [8].

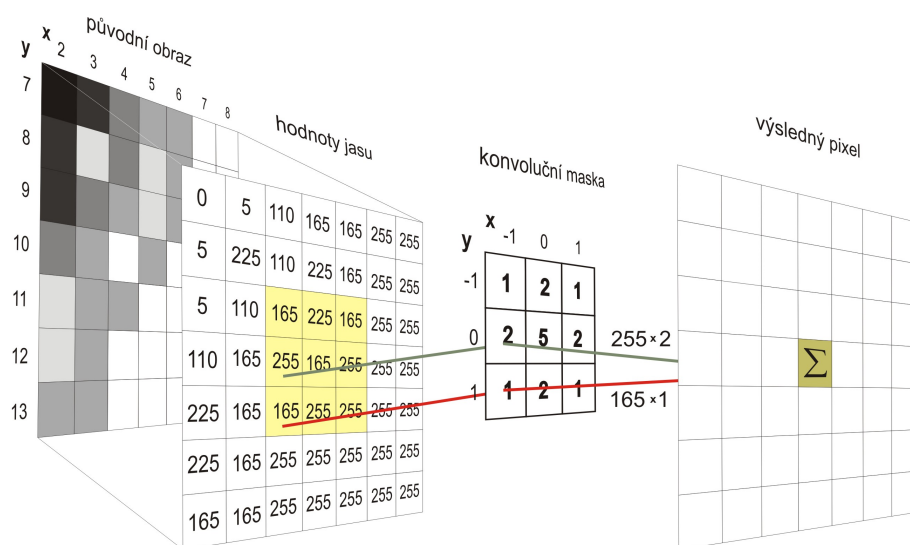
Gaussův filtr lze implementovat dvěma způsoby:

1. Dvoupřúchodový jednodimenzionální filtr:

Nejprve se vypočtou data jednodimenzionálního konvolučního jádra o velikosti $1 + \text{poloměr} \cdot 2$ prvků. Jednotlivé body se spočítají podle následující funkce:

$$G(r) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{r^2}{2\sigma^2}}$$

kde r je vzdálenost bodu od středu jádra a rozptyl σ se volí jako 1/3 poloměru. Poté provedeme konvoluci obrazu s tímto jádrem orientovaným na šířku a následně na výšku.



Obrázek 4.5: Princip diskrétní dvojrozměrné konvoluce; převzato z [11]

2. Dvojdímenzionální filtr:

Spočítáme parametry dvojdímenzionálního konvolučního jádra o velikosti $[1 + \text{poloměr} * 2, 1 + \text{poloměr} * 2]$ podle následující funkce:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Parametry x a y jsou vzdálenosti ve směru x a y od středu jádra. S tímto jádrem provedeme konvoluci obrazu.

Asymptotická složitost výpočtu dvouprůchodového jednodímenzionálního filtru bude tedy $O(2WHn)$, složitost jednodímenzionálního filtru $O(WHn^2)$, kde n je velikost strany jádra, W šířka obrazu a H jeho výška v pixelech. Výhodnější je tedy použít jednodímenzionální filtr se dvěma průchody.

5 Programová dokumentace

5.1 Aplikační logika

Aplikační logika je klíčová část aplikace. Její UML¹ diagram tříd je zobrazen na obr. 5.1. Všechny třídy, spadající do aplikační logiky, jsou umístěny v balíčku *grabber*.

Základem aplikace je rozhraní *Grabbable*. Toto rozhraní deklaruje veškeré potřebné metody pro komunikaci s nahrávacími zařízeními. Toto rozhraní implementují čtyři třídy. Tři z těchto tříd přistupují přímo k nahrávacím zařízením – budeme jim říkat *grabbery*. Konkrétně to jsou následující třídy:

- *CivilGrabber*: Stará se snímání z webových kamer a kamer připojených přes FireWire.
- *WiaGrabber*: Stará se o snímání z digitálních zrcadlovek a fotoaparátů.
- *NetworkGrabber*: Stahuje snímky z kamer dostupných přes protokol *http*.

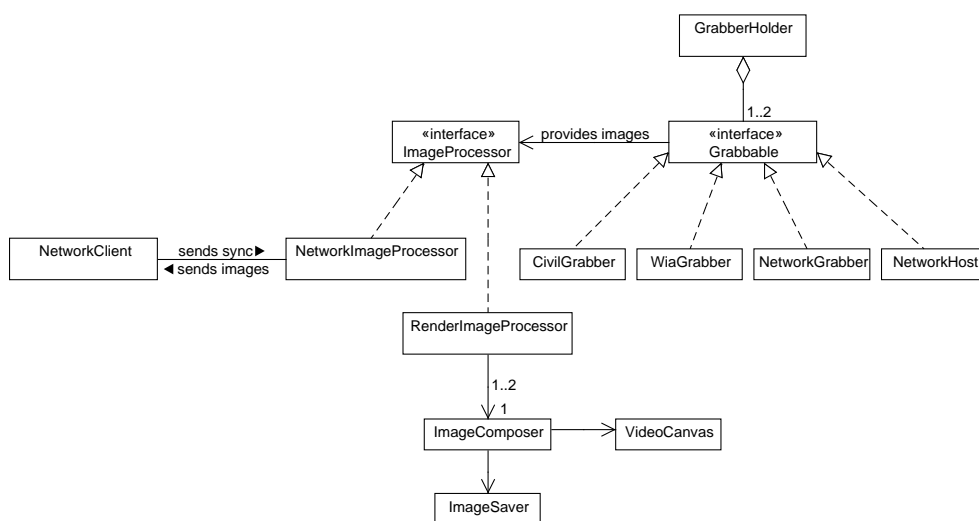
Poslední třída, implementující rozhraní *Grabbable*, je *NetworkHost* a její účel je lehce odlišný od *grabberů*. O té si povíme později. Nyní se podíváme na jednotlivé *grabbery*.

5.1.1 *CivilGrabber*

CivilGrabber je třída, sloužící pro připojení k zařízením, které poskytují kontinuální datový tok obrazu. Jednotlivé obrázky dokáže poskytovat instanci třídy implementující rozhraní *ImageProcessor* pro další využití.

CivilGrabber využívá služeb knihovny LTI-CIVIL[1]. LTI-CIVIL je volně dostupná knihovna, umožňující snímání z výše uvedených zařízení. Postup pro vybírání a konfiguraci zařízení je jednoduchý:

¹Unified Modeling Language



Obrázek 5.1: UML diagram aplikační logiky programu

1. Vytvoří se snímací systém.
2. Ze systému se získá seznam zařízení a k jednomu se připojí.
3. Z připojeného zařízení se získá seznam formátů a jeden se zvolí.
4. Spustí se snímání.

Získané snímky se pomocí zabudované funkce převedou na instanci `BufferedImage`, které se dále zpracovávají.

5.1.2 WiaGrabber

`WiaGrabber` je třída, umožňující připojení k digitálním zrcadlovkám za pomoci WIA² Automation Library[3].

²Windows Image Acquisition

WIA

Windows Image Acquisition je platforma firmy Microsoft, zprostředkovávající komunikaci mezi snímacími zařízeními (především skenery a fotoaparáty) a aplikacemi na platformě Microsoft Windows.

WIA API³ má podle zdroje [3] zabudovanou podporu pro následující činnosti:

- zobrazení seznamu dostupných zařízení,
- možnost připojení se na více zařízení současně,
- čtení a zápis vlastností zařízení,
- přenos dat ze zařízení se zachováním všech jejich vlastností,
- notifikace o stavu zařízení a řízení událostí.

Propojení jazyka Java s WIA

Pro komunikaci s WIA je použita knihovna Com4J, zprostředkovávající rozhraní pro spolupráci Javy a Microsoft COM⁴, uvádí [2]. Pro přístup k zařízením slouží knihovna WIA Automation Library, umístěná v souboru `wia-aut.dll`.

Aby bylo možno používat metody knihovny WIA Automation Library, je nutné vytvořit hlavičkové soubory s deklaracemi jejich metod, umožňující napojení Javy na COM. Tuto práci za nás udělá Com4J. Spuštěním programu ze souboru `tlbimp.jar`, dostupného v balíku Com4J, pomocí příkazu `java -jar tlbimp.jar -o wsh -p <balíček> <cesta ke knihovně>` vytvoříme rozhraní pro Javu. Každé rozhraní je identifikováno jednoznačným řetězcem zvaným GUID⁵, podle něhož dokáže Com4J vyhledat definici metody v systému.

³Application Programming Interface

⁴Component Object Model – Komponentová technologie firmy Microsoft

⁵Globally Unique Identifier

Implementace grabberu

Postup pro připojení se k zařízení je podobný postupu pro připojení kamery přes LTI-CIVIL:

1. Vytvoří se manažer, sloužící pro zprostředkování informací o dostupných zařízeních.
2. Získá se seznam zařízení a k jednomu se připojí.
3. Registruje se posluchač, který v případě vytvoření nového snímku provede jeho zpracování.
4. Může se začít snímat.

Každé připojené zařízení obsahuje následující položky:

- *Seznam vlastností*: Obsahuje veškeré vlastnosti zařízení. Některé vlastnosti jsou pouze informativního charakteru (název zařízení, stav baterie, verze firmwaru...), jiné slouží pro čtení i zápis (clonové číslo, rychlost závěrky, ISO, vyvážení bílé...).
- *Příkazy*: Seznam příkazů, které je možno posílat do zařízení z aplikace. Aby bylo možné pořídit obrázek, musí dané zařízení podporovat příkaz `Take Picture`.
- *Položky*: Obsahuje položky uložené v zařízení (typicky fotografie na paměťové kartě).
- *Události*: Seznam událostí, které mohou v zařízení nastat. Těmto událostem lze přiřadit posluchače, kteří při vyvolání dané události vyvolají metodu `void onEvent()` z rozhraní `_IDeviceManagerEvents`. `WiaGrabber` využívá této metody k přenosu fotografie do počítače a její uložení na pevný disk.

Protože je knihovna `Com4J` stále ve vývoji, nelze využít všech možností, které by šly využít např. v jazyce `C#`. Z tohoto důvodu nelze přistupovat k jednotlivým pixelům obrázku a převést jej na `BufferedImage`. Proto se pořízené fotografie přímo ukládají na pevný disk.

5.1.3 NetworkGrabber

`NetworkGrabber` slouží ke stahování obrázků z kamer dostupných přes protokol HTTP⁶. Snímky jsou stahovány periodicky a převáděny do `BufferedImage`, které se posílají k dalšímu zpracování instancí třídy implementující rozhraní `ImageProcessor`.

5.1.4 Řízení chodu programu

Systém podporuje jeden až dva současně běžící grabbery. K uchování informací, o jaké grabbery se jedná, slouží třída `GrabberHolder`. Tato třída má za úkol udržovat referenční proměnné na právě používané grabbery. Dále disponuje metodami pro řízení chodu programu: Umí uchovávané grabbery jednotně spouštět či zastavovat a nastavovat časování.

Časování

Časování spočívá ve vytvoření vlákna, které po každém uplynutí časového intervalu nechá uložit snímky. `GrabberHolder` umí nastavit následující vlastnosti časování:

- *Začátek časování:* ihned, v zadaný čas nebo po uplynutí určené doby.
- *Zastavení časování:* ručně, v zadaný čas, po uplynutí určité doby či po uložení stanoveného počtu snímků.
- *Interval časování, složku pro ukládání dat a název souboru.*

Třídy implementující `ImageProcessor`

`GrabberHolder` je zodpovědný i za přiřazování tzv. procesorů grabberům. Procesorem se rozumí třída implementující rozhraní `ImageProcessor`. `ImageProcessor` disponuje jednou metodou, a to `void processImage(BufferedImage img)`, která předá dodaný snímek dedikované třídě. Existují dvě třídy implementující `ImageProcessor`:

⁶Hypertext Transfer Protocol

1. *RenderImageProcessor*: slouží k předávání obrázků instanci třídy *ImageComposer*, která transformuje předávaný obrázek pro zobrazení na obrazovku programu nebo pro uložení.
2. *NetworkImageProcessor*: slouží k předání obrázků síťovému klientovi, který jej odešle po síti hostující instanci *FlyCatcheru*. Síťová komunikace bude podrobně popsána později.

5.1.5 Manipulace s obrazem

Pro manipulaci s obrazem slouží třída *ImageComposer*. Této třídě jsou periodicky dodávány obrázky ze zařízení určené k zobrazení na vykreslovací plátno nebo k uložení. Každý dodávaný snímek projde zpracovávacím cyklem, ve kterém je patřičně upraven. Při dodávání snímků v režimu stereoskopie se musí tyto snímky synchronizovat za pomoci semaforů.

Nyní si uveďme přehled metod, pomocí nichž může *ImageComposer* dodávané obrázky upravovat:

- *cropImage()*: Ořízne obrázek na podle nastavených parametrů.
- *createAnaglyphImage()*: Zadané obrázky převede do formátu červeno-azurového anaglyfu.
- *createAdjacentImage()*: Ze zadaných obrázků vytvoří nový obrázek dvojnásobné šířky, který bude obsahovat data z obou obrázků vedle sebe.
- *createBlendedImage()*: Prolne obrázky s 50% průhledností přes sebe.
- *convertToGrayscale()*: Převede obrázek do černobílé škály pomocí vestavěné funkce.
- *mosaic()*: Pixelizuje obrázek.
- *gaussianBlur()*: Aplikuje na obrázek Gaussovo rozostření.
- *expandEdges()*: Zvětší plochu obrázku na okrajích o zadaný počet pixelů tak, aby bylo možno provést konvoluci v každém bodě obrázku.
- *applyTables()*: Aplikuje tabulky s barevnou korekcí.

- `transform()`: Provede transformaci obrázku – pokud je nastaven ořez, obrázek ořízne, a poté změní velikost obrázku na nastavenou šířku a výšku.

5.1.6 Sít'ová komunikace

Program je možno provozovat v sít'ovém režimu. Tento režim nalezne využití v případě, že z technických důvodů nelze snímat ve stereoskopickém režimu ze dvou zařízení najednou.

Sít'ový režim spočívá ve spuštění dvou instancí programu: v instanci programu, ke které je připojena levá kamera, se spustí server. Tím se vytvoří instance třídy `NetworkHost`, která otevře zadaný port a začne na něm naslouchat. Ve druhé instanci se nastaví kamera pro pravé oko a program se připojí k námi předem vytvořenému serveru. Při této akci se vytvoří instance třídy `NetworkClient`, která se automaticky připojí k našemu serveru. Po připojení se na serverové straně ve třídě `GrabberHolder` nastaví sekundární grabber na `NetworkHost` a spustí se snímání. Tímto se odešle příkaz klientské straně, která jako reakci začne periodicky posílat serveru obrázky. Ty se pak zpracovávají běžným způsobem.

Je nutno podotknout, že jako klient se smí připojit pouze program, který využívá grabber s automatickým exportem obrázků v `BufferedImage`. `WiaGrabber` tedy připojit nepůjde.

Sít'ový protokol

Klient se serverem komunikují hybridním protokolem. Řídící povely se posílají v textové formě a obrázky ve formě binární. Podívejme se ještě na formát řídicích sekvencí:

- *Klient* → *Server*
 - `r <šířka> <výška>`: Klient je připraven posílat obrázky o zadané velikosti.
 - `i<typ> <délka>`: Klient bude posílat obrázek zadaného typu o délce v bytech. Typ je zatím neměnný – v budoucích verzích bude

určovat způsob, zda bude obrázek určen pro vykreslení nebo uložení.

– **x**: Klient se odpojuje.

- *Server* → *Klient*

– **r**: Server je připraven.

– **i**: Server čeká na obrázek.

– **f<stav>**: Změna stavu posílání obrázků. Stav 1: Klient má začít posílat obrázky; stav 0: Klient má přestat posílat obrázky.

– **x**: Server se ukončuje.

5.2 Grafické uživatelské rozhraní

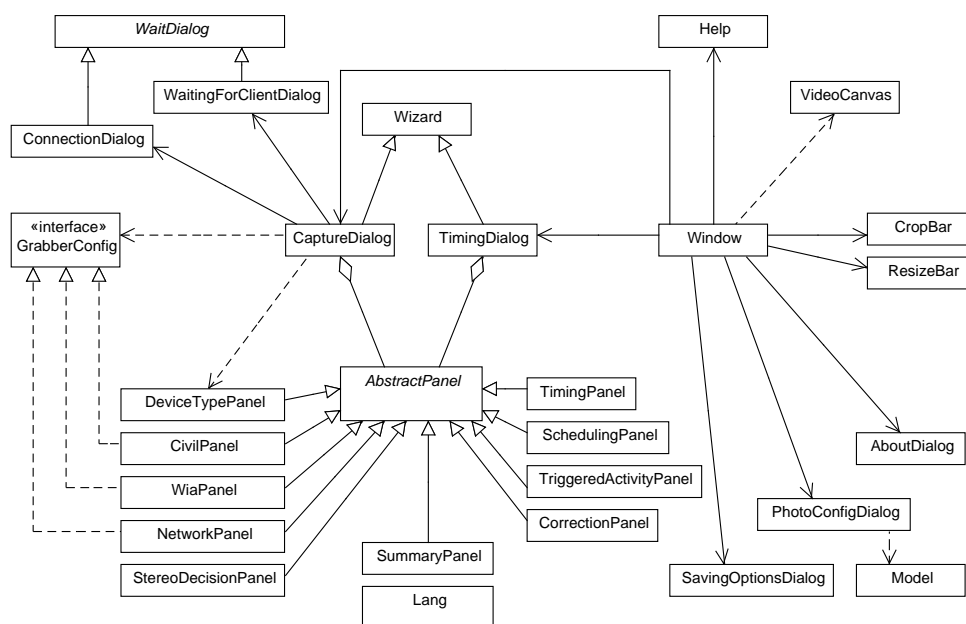
Grafické uživatelské rozhraní programu FlyCatcher se skládá z 28 veřejných tříd. Všechny tyto třídy jsou zobrazeny v UML diagramu na obrázku 5.2. Veškeré třídy, spadající do GUI jsou umístěny v balíčku `grabber.gui`.

5.2.1 Window

Hlavním stavebním kamenem GUI je třída `Window`. Jedná se o třídu oddělenou od `JFrame`, starající se o vykreslování hlavního okna. Tato třída slouží jako vstupní bod programu, tj. obsahuje metodu `main(String[] args)`.

Tato třída se stará o vytvoření celého uživatelského rozhraní. Protože je GUI celkem obsáhlé a trvalo by velmi dlouho, kdyby se měly instance všech tříd vytvořit při startu programu, je zde z velké části použita odložená inicializace. To znamená, že instance zůstávají neinicializovány a k jejich inicializaci dojde až ve chvíli, kdy mají být použity. Typickým příkladem je třída s nápovědou. Většina uživatelů nápovědu využívat nebude, a proto není zapotřebí ji od počátku udržovat v paměti.

Nyní se vrátíme zpět k hlavnímu oknu. To obsahuje menu a nástrojovou lištu. Nástrojová lišta obsahuje vybrané položky z menu. Abych nemusel programovat funkčnost položek v menu a tlačítek na liště dvakrát, využil jsem třídu `AbstractAction`, od které jsem oddědil potomky, které zastupují právě



Obrázek 5.2: UML diagram uživatelského rozhraní aplikace

tyto společné položky. Všechny tyto „akce“ jsou ve třídě `Window` uloženy jako vnitřní třídy.

5.2.2 Průvodce

Další důležitou částí uživatelského rozhraní jsou dialogy typu průvodce. Všechny třídy, spadající do této skupiny jsou uloženy v balíčku `wizard`.

Společnou třídou těchto průvodců je třída `Wizard`, oddělená od `JPanel`. Disponuje obrázkem ve své levé části a třemi různými tlačítky vespod pro pohyb v průvodci. Z těchto tlačítek jsou viditelná vždy jen dvě. Tlačítko pro posun dále může být nahrazeno tlačítkem pro uzavření dialogu. Na hlavní ploše těchto dialogů je panel s rozvržením `CardLayout`, na kterém se zobrazují příslušné panely s obsahem. Tyto panely sdílí vlastnosti abstraktní třídy `AbstractPanel`. Tyto panely se dělí do dvou skupin podle typu průvodce.

V programu existují dva průvodce: průvodce snímáním (`CaptureDialog`) a průvodce časováním a ukládáním (`TimingDialog`).

Nyní se podívejme podrobněji na obsahové panely a jejich obsah.

5.2.3 Panely oddělené od `AbstractPanel`

Každý panel, oddělený od třídy `AbstractPanel` má svého „rodičovského průvodce“, do kterého je přiřazen. Musí rovněž implementovat abstraktní metody pro obnovu při zpětném posunu v průvodci a kompletní obnovu. Dále tyto panely obsahují metodu pro změnu jazyka, ke které se dostaneme později.

Panely patřící do dialogu pro snímání

- `DeviceTypePanel`: Jedná se o panel, ve kterém uživatel vybírá typ zařízení. Použit je `GridLayout`.
- `CivilPanel`: Panel, sloužící k výběru a nastavení vlastností webových kamer přes knihovnu LTI-CIVIL.
- `WiaPanel`: Jedná se o panel, sloužící k výběru digitálních zrcadlovek pomocí Windows Image Acquisition.
- `NetworkPanel`: Jedná se o panel, sloužící ke konfiguraci připojení ke kameře přes protokol HTTP.
- `StereoDecisionPanel`: Panel, ve kterém jsou uživateli nabídnuty způsoby připojení sekundární kamery.

Panely patřící do dialogu pro časování

- `TimingPanel`: Jedná se o panel, ve kterém uživatel vybírá základní vlastnosti ukládání snímků. Může zde nastavit cestu do složky, kam se budou soubory ukládat, časový interval snímání a název souborů.
- `SchedulingPanel`: Panel, ve kterém je možno nastavit plánování počátku a konce ukládání.
- `TriggeredActivityPanel`: Jedná se o panel, sloužící k aktivaci `Activity Triggered` režimu.
- `CorrectionPanel`: Jedná se o panel, sloužící k aktivaci korekcí barev.

Panel `SummaryPanel` je pro oba druhy dialogů společný a obsahuje pouze textovou informaci o zavření průvodce.

Za zmínku ještě stojí `WaitDialog` a jeho potomci `ConnectionDialog` a `WaitingForClientDialog`, sloužící jako informační boxy při vytváření síťového propojení dvou instancí programu.

5.2.4 Plátno `VideoCanvas`

`VideoCanvas` je plocha oddělená od `JPanel`, sloužící pro zobrazení náhledu videa. Je zde překryta metoda `public void paint(Graphics g)`, sloužící k vykreslení komponenty. V té jsou na vyžádání vykreslovány jednotlivé obrázky, poskytované třídou `ImageComposer`, sloužící ke skládání výsledného obrazu.

5.2.5 Ostatní okna a dialogy

V programu se nachází ještě další dialogy a okna, jejichž princip je třeba vysvětlit. Konkrétně se jedná o třídu `PhotoConfigDialog`, která slouží pro interaktivní nastavení parametrů digitálních zrcadlovek, připojených přes WIA. Toto dialogové okno obsahuje tabulku, na jejíchž řádcích se nachází jednotlivé parametry nastavení, jako je např. clona, čas závěrky či citlivost ISO. Protože je knihovna `Com4J` stále ve vývoji, nelze zatím využívat všech funkcí těchto rozhraní, a proto zde není možné vybírat jednotlivé možnosti nastavené z drop-down seznamu (`JComboBox`), což je docela škoda.

Nápověda

Okno nápovědy je `JFrame`, obsahující stromový seznam prvků nápovědy po levé straně. Zbytek plochy je věnován pro zobrazení obsahu nápovědy, jejíž text ve formátu HTML je zobrazován v needitovatelných `JEditorPane`. Tyto panely jsou opět umístěny v `CardLayout`.

5.2.6 Lokalizace

Program obsahuje statickou třídu `Lang` pro lokalizaci, kterou využívá drtivá většina komponent v GUI. Proto na v UML diagramu na obr. 5.2 nejsou uvedeny propojení na tuto třídu, protože by se schéma stalo velice nepřehledným.

Třída `Lang` obsahuje tři statické metody. Metodu `Locale` `getLanguage()` a `setLanguage(Locale locale)`, sloužící k nastavování jazyka (a jeho zjištění) a metodu `String` `val(String key)`, sloužící k získání jedné lokalizované hodnoty ze souboru `Texts_cs_CZ.properties` nebo `Texts_en_GB.properties`, v závislosti na použitém jazyce.

Ostatní třídy, využívající tuto lokalizační třídu disponují metodou `void changeLanguage()`, ve které na požadavek z třídy hlavního okna změní veškeré své texty.

6 Závěr

Cílem této práce bylo navrhnout a napsat softwarový systém pro časosběrné vizuální snímání pomalých dějů pomocí videokamer či fotoaparátů.

Vyvinutý systém zadání splňuje: Umí snímat pomocí webových kamer, videokamer, kamer přes protokol HTTP a digitálních zrcadlovek značky Nikon přes Windows Image Acquisition. Program byl odladěn na zrcadlovce Nikon D40, testován byl i na Nikon D2X. Zrcadlovky ostatních výrobců nepodporují vzdálené snímání přes WIA, a proto ve vytvořeném systému nejdou použít. Neúspěchem skončily testy zrcadlovek Canon EOS D5 Mark II, Canon EOS Rebel XT_i a Olympus E-510.

Program dále dovede:

- *Snímání ve stereoskopickém režimu:* Snímání probíhá ze dvou libovolných zařízení současně.
- *Activity-triggered snímání:* Snímek se uloží pouze tehdy, bude-li se lišit od předchozího uloženého snímku.
- *Korekce expozice a vyvážení bílé:* Ukládané snímky budou barevně synchronizovány, aby nedocházelo k blikání obrazu.

Program, který jsem napsal, se tedy stává velmi zajímavou alternativou k existujícímu software. Mezi jeho nevýhody ovšem patří vyšší náročnost na výpočetní výkon procesoru, zvláště pokud je vykreslováno video v HD rozlišení, protože rendering probíhá softwarově.

Plány do budoucna

V budoucnu bych se rád vývoji tohoto programu dále věnoval. Konkrétně bych rád přidal možnost hardwarové podpory pro vykreslování videa a možnost exportu nejen série statických snímků, ale i videa, do kterého se nyní musí převádět snímky v externích programech, jako je např. VirtualDub.

Literatura

- [1] Larson Technologies Inc.: LTI-CIVIL. 2007, [Online; navštíveno 14. 06. 2012].
URL <http://lti-civil.org/>
- [2] Com4J. 2012, [Online; navštíveno 14. 06. 2012].
URL <http://com4j.java.net/>
- [3] Microsoft: Windows Image Acquisition. 2012, [Online; navštíveno 19. 06. 2012].
URL <http://msdn.microsoft.com/en-us/library/windows/desktop/ms630368%28v=vs.85%29.aspx>
- [4] Burger, W.; Burge, M. J.: *Digital Image Processing: An Algorithmic Introduction using Java*. Springer Science+Business Media, 2008, ISBN 978-1-84628-379-6.
- [5] Chaney, M.: Video Frame Rates (24p, 25p, 30p, 60i). 2009, [Online; navštíveno 12. 07. 2012].
URL <http://ddisoftware.com/tech/articles/march-2009-video-frame-rates-%2824p-25p-30p-60i%29/>
- [6] Chylinski, R.: *Time-lapse Photography: A Complete Introduction to Shooting, Processing and Rendering Time-lapse Movies with a DSLR Camera*. 2012, ISBN 978-0-9853757-0-6.
- [7] Čížek, P.: *Prostorové zobrazování*. Diplomová práce, Západočeská univerzita v Plzni, 2005.
- [8] Szeliski, R.: *Computer Vision: Algorithms and Applications*. 2010.
- [9] Wikipedia: Histogram matching — Wikipedia, The Free Encyclopedia. 2011, [Online; navštíveno 12. 06. 2012].

- URL http://en.wikipedia.org/w/index.php?title=Histogram_matching&oldid=443295324/
- [10] Wikipedie: Dirichletův princip — Wikipedie: Otevřená encyklopedie. 2012, [Online; navštíveno 19. 06. 2012].
URL http://cs.wikipedia.org/w/index.php?title=Dirichlet%C5%AFv_princip&oldid=8179951/
- [11] Wikipedie: Konvoluce — Wikipedie: Otevřená encyklopedie. 2012, [Online; navštíveno 13. 06. 2012].
URL <http://cs.wikipedia.org/w/index.php?title=Konvoluce&oldid=8549411/>
- [12] Wikipedie: Paralaxa — Wikipedie: Otevřená encyklopedie. 2012, [Online; navštíveno 19. 06. 2012].
URL <http://cs.wikipedia.org/w/index.php?title=Paralaxa&oldid=8372807/>
- [13] Wikipedie: Výřez — Wikipedie: Otevřená encyklopedie. 2012, [Online; navštíveno 12. 06. 2012].
URL <http://cs.wikipedia.org/w/index.php?title=V%C3%BD%C5%99ez&oldid=8296372/>

Seznam obrázků

4.1	Ukázky histogramů	11
4.2	RGB histogram	11
4.3	Korekce barev a expozice	13
4.4	Zřetěžené mapování	14
4.5	Princip diskrétní dvojrozměrné konvoluce	16
5.1	UML diagram aplikační logiky programu	18
5.2	UML diagram uživatelského rozhraní aplikace	25
A.1	Hlavní okno programu	35

Seznam tabulek

3.1	Možnosti připojení zařízení	9
3.2	Speciální funkce programů	9
A.1	Pořizování snímků	39

A Uživatelská dokumentace

A.1 Požadavky

Pro spuštění programu je třeba mít nainstalováno Java JRE¹ verze 6 nebo vyšší. Pro možnost snímání z webových kamer a kamer připojených přes FireWire je třeba program spouštět na 32-bitové JVM platformě, pro snímání ze zrcadlovek je třeba program spouštět v operačním systému Windows.

A.2 Překlad a spuštění programu

Program je primárně určen pro operační systém Windows. Na ostatních systémech nebyl jeho běh testován a proto nemusí fungovat správně.

A.2.1 Překlad programu

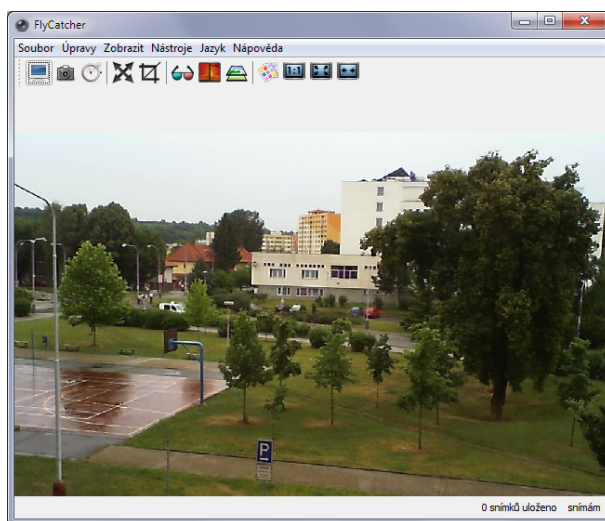
Překlad se provádí pomocí přiloženého Ant skriptu `build.xml` přes target `fat-jar`. Po vytvoření jar souboru je nutno do složky s jar souborem nakošírovat složku s nativními knihovnami `native` a složku `img` s grafikou.

A.2.2 Spuštění programu

V operačním systému Windows je třeba spustit program příkazem `java -jar -Djava.library.path="native/win32-x86"Grabber.jar` nebo spuštěním souboru `start.bat`.

V ostatních operačních systémech se bude spouštěcí příkaz lišit pouze v parametru `java.library.path`, který uvádí cestu ke složce, ve které má JVM hledat nativní knihovny. V systému Linux bude hodnota parametru `native/linux-x86` nebo `native/linux-amd64` (v závislosti na architektuře) a v Mac OS X `native/macosx-universal`.

¹Java Runtime Enviroment



Obrázek A.1: *Hlavní okno programu: Na obrázku lze vidět hlavní okno programu v režimu stereoskopického snímání*

A.3 Obsluha programu

Po načtení programu se zobrazí hlavní okno, viz obr. A.1, na kterém uživatel nalezne všechny základní ovládací prvky programu.

A.3.1 Hlavní okno

Hlavní okno programu se skládá z plochy pro video, menu, nástrojové lišty a dolní stavové lišty.

A.3.2 Snímání

Kliknutím na tlačítko „Nové snímání...“ v menu **Soubor** se uživateli zobrazí Průvodce výběrem zařízení a režimu snímání. Uživatel provede výběr typu zařízení, které bude připojeno jako primární.

Připojení videokamery

Tento typ snímání využívá knihovnu LTI-CIVIL. [1] Pro správné nastavení snímání je třeba vybrat zařízení a poté jeho parametry, kterými jsou barevný formát, velikost snímku a zobrazovací frekvence ve snímcích za sekundu.

Připojení digitální zrcadlovky

Při snímání přes WIA je nutné vybrat zařízení a příponu implicitního formátu, ve kterém fotoaparát snímky ukládá. Typicky JPG či NEF (Nikon). Po zavření snímacího dialogu se zpřístupní položka Konfigurace fotoaparátů v menu Nástroje. Zde je možno měnit základní nastavení fotoaparátu za běhu.

Snímání ze síťové kamery

Pro síťové snímání je třeba zadat URL obrázku, který je periodicky aktualizován. Po zadání adresy uživatel ověří její platnost kliknutím na tlačítko Ověřit.

Stereoskopické snímání

Na další straně dialogu lze přidat druhé zařízení pro stereoskopické snímání a nebo zahájit snímání pouze s jedním zařízením. Stereoskopické snímání lze nastavit dvěma způsoby:

1. *Lokální snímání:* Toto nastavení je preferováno a podporováno všemi typy zařízení. Druhé zařízení (pro pravé oko) lze přidat po zvolení možnosti „Přidat druhé zařízení“.
2. *Síťové snímání:* Toto nastavení je doporučeno použít v případě, že z technických důvodů nelze připojit obě zařízení k jednomu počítači. Nejprve je nutno vytvořit server. Uživatel tedy označí možnost „Vytvořit server“, poté spustí program na druhém počítači, otevře dialog pro nové snímání, vybere zařízení a zvolí možnost „Připojit se jako klient“. Do pole Adresa pak vyplní IP adresu hostitelského počítače, který čeká na připojení klienta a stiskne tlačítko „Hotovo“. Jako klient nelze použít program s připojenou digitální zrcadlovkou, poněvadž se pořízené

snímky ze zrcadlovky dají ukládat pouze na lokální disk. Proto je nutno jej nastavit jako server a nebo připojit lokálně.

A.3.3 Náhled videa

Pokud používáme zařízení, umožňující náhled videa, zpracované snímky se budou automaticky zobrazovat na ploše hlavního okna. Uživatel si může měnit styl zobrazení těchto snímků na obrazovce, aniž by ovlivnil formát ukládaných snímků. K tomuto účelu slouží položky v menu „Zobrazit“.

Základní nástroje

- *Zobrazit video*: Slouží pro zobrazení nebo skrytí videa na obrazovce. Pokud je video zobrazeno, program potřebuje více výkonu CPU pro jeho rendering.
- *Barevný náhled*: Zapne či vypne černobílé zobrazení videa.

Nástroje k formátu zobrazení stereoskopického videa

- *Anaglyf*: Video bude zobrazeno ve formě červeno-azurového anaglyfu.
- *Vedle sebe*: Video z kamer budou zobrazena vedle sebe.
- *Překrytí*: Video z kamer se budou překrývat. Toto nastavení slouží převážně k eliminaci vertikální paralaxy, vzniklé natočením kamer vůči sobě.

Nástroje pro změnu velikosti videa

- *Původní velikost*: Video bude zobrazeno v původní velikosti bez ohledu na velikost okna programu.
- *Roztáhnout video*: Video bude roztáhnuto tak, aby vyplnilo celou oblast okna.
- *Minimální velikost*: Pokud se video nevejde do okna, zmenší se.

A.3.4 Časování

Kliknutím na tlačítko „Nastavit časování...“ v menu **Soubor** se uživateli zobrazí dialogové okno ve stylu průvodce. Na první straně tohoto dialogu musí vyplnit snímací interval a cestu ke složce, kam se budou zachycené snímky ukládat. Rovněž je zde možno nastavit jméno ukládaných souborů. Každý uložený soubor bude obsahovat nastavené jméno souboru, pořadové číslo snímku a identifikační značku, zda se jedná o levý či pravý obrázek a příponu souboru.

Na další straně je možno nastavit čas spuštění ukládacího procesu. Uživatel si může vybrat mezi třemi způsoby:

1. *Spustit hned*: Ukládání snímků začne okamžitě po zavření dialogu.
2. *Spustit po uplynutí časového intervalu*: Od zavření dialogového okna se začne odečítat čas, po jehož uplynutí se začnou snímky automaticky ukládat. Tento čas je možno sledovat ve stavové liště programu.
3. *Spustit v konkrétní datum a čas*: Ukládání snímků začne ve stanovený čas. Uživateli bude zobrazen zbývající čas do startu stejným způsobem jako v minulém případě.

Na následující straně uživatel vybere způsob ukončení snímání. Podobně jako v předchozí části dialogu si může vybrat mezi několika možnostmi:

1. *Zastavit manuálně*: Snímání musí ukončit uživatel.
2. *Zastavit po uložení zadaného počtu snímků*.
3. *Zastavit po uplynutí časového intervalu*.
4. *Zastavit v konkrétní datum a čas*.

Na další straně průvodce lze aktivovat tzv. *Activity Triggered* režim. Nový snímek, připravený k uložení, se uloží pouze tehdy, pokud se liší od předchozího uloženého snímku minimálně o prahovou hodnotu. Prahová hodnota je udávána v procentech a je doporučeno ji nastavit v řádu jednotek procent.

Poslední strana slouží ke korekci obrazu. Ta odstraní blikání obrazu při změně expoziční hodnoty a vyvážení bílé. Tuto funkci je doporučeno zapnout,

Typ zařízení	Automatické poskytování	Jednotné ukládání
Kamera (CIVIL)	Ano	Ano
Zrcadlovka (WIA)	Ne	Ne
Sít'ová kamera	Ano	Ano

Tabulka A.1: *Pořizování snímků:* Zařízení s automatickým poskytováním snímků umožňují zobrazit své snímky v hlavním okně programu; zařízení s jednotným ukládáním zpracovávají své obrázky standardním způsobem a lze na ně aplikovat filtry.

pokud je ke snímání používáno zařízení s automatickým nastavením expozice a vyvážení bílé. Rovněž je funkci doporučeno aktivovat při stereoskopickém snímání ze dvou kamer s odlišnými senzory, aby došlo ke sjednocení barevného podání obrazu z obou kamer. Referenční vahou se rozumí hodnota, s jakou se bude na barevném podání ukládaného obrázku podílet barevná struktura referenčního snímku. Vyšší hodnoty lépe redukuje blikání, ale nově ukládané obrázky budou náchylnější k dočasnému poklesu dynamického rozsahu snímku.

A.3.5 Možnosti ukládání

Základní nastavení ukládání se neprovádí v průvodci časováním, ale v samostatném dialogu, přístupným z menu **Nástroje**. Zde je možno zvolit formát ukládaného obrázku (za předpokladu, že se nejedná o snímání ze zrcadlovky) a způsob uložení obrázků při stereoskopickém snímání.