
TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky a mezioborových inženýrských studií

Studijní program: M2612 – Elektrotechnika a informatika
Studijní obor: 3906T001 Mechatronika

**Vývoj videometrické aplikace
v prostředí OS Linux**

**Development of videometrical application
for OS Linux**

Diplomová práce

Autor:	Tomáš Černý
Vedoucí práce:	Ing. Lukáš Matela, PhD.
Konzultant:	Doc. Ing. Ivan Jaksch, CSc.

V Liberci 10. 9. 2006

TECHNICKÁ UNIVERZITA V LIBERCI

Fakulta mechatroniky a mezioborových inženýrských studií

Katedra měření

Akademický rok: 2005/2006

ZADÁNÍ DIPLOMOVÉ PRÁCE

pro: Tomáše Černého
studijní program: M 2612 – Elektrotechnika a informatika
obor: 3906T001 - Mechatronika

Vedoucí katedry Vám ve smyslu zákona o vysokých školách č.111/1998 Sb. určuje tuto diplomovou práci:

Název tématu:

Vývoj videometrické aplikace v prostředí OS Linux

Zásady pro vypracování:

1. Seznamte se s problematikou zachytávacích karet s čipem Bt878 a se způsobem použití karet v OS Linux
2. Seznamte se s knihovnou Open Computer Vision Library
3. Dle podrobnějších požadavků sestavte aplikaci pro videometrická měření
4. Začněte vhodné způsoby kalibrace při použití různých variant objektivů
5. Ověřte možnosti uplatnění aplikace včetně zhodnocení přesnosti

Rozsah grafických prací: dle potřeby dokumentace

Rozsah průvodní zprávy: cca 40 až 50 stran

Seznam odborné literatury:

- [1] Fischer, J.: Optoelektronické senzory a videometrie, ČVUT FEL Praha 2002
- [2] Hlaváč, V., Sedláček, M.: Zpracování signálů a obrazů, ČVUT FEL Praha 2000
- [3] Mitchell, Oldham, Samuel: Advanced Linux Programming
- [4] Open CV Library <
<http://www.intel.com/technology/computing/opencv/index.htm>>

Vedoucí diplomové práce:

Ing. Lukáš Matela

Konzultant:

Doc. Ing. Ivan Jaksch, CSc.

Zadání diplomové práce:

27.10.2005

Termín odevzdání diplomové práce:

19. 5. 2006

L.S.

.....
Vedoucí katedry

.....
Děkan

V Liberci dne

Prohlášení

Byl(a) jsem seznámen(a) s tím, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 o právu autorském, zejména § 60 (školní dílo).

Beru na vědomí, že TUL má právo na uzavření licenční smlouvy o užití mé diplomové práce a prohlašuji, že **s o u h l a s í m** s případným užitím mé diplomové práce (prodej, zapůjčení apod.).

Jsem si vědom(a) toho, že užití své diplomové práce či poskytnout licenci k jejímu využití mohu jen se souhlasem TUL, která má právo ode mne požadovat přiměřený příspěvek na úhradu nákladů, vynaložených univerzitou na vytvoření díla (až do jejich skutečné výše).

Diplomovou práci jsem vypracoval(a) samostatně s použitím uvedené literatury a na základě konzultací s vedoucím diplomové práce a konzultantem.

10.9.2006

Tomáš Černý

.....

Poděkování

Rád bych zde poděkoval Ing. Lukáši Matelovi, PhD. za všestrannou pomoc při zpracování diplomové práce, zapůjčení studijních materiálů, zachytávací karty a kamery. Dále nesmím zapomenout poděkovat svému zaměstnavateli, Ing. Františku Benčovi, PhD., za poskytnutí pracovního volna při dokončování diplomové práce a své partnerce Evě za vysokou míru pochopení a korekci textu.

Anotace

Videometrie umožňuje eliminovat lidský faktor z měřících a vyhodnocovacích činností. Jedná se o bezkontaktní měření vzdálenosti z parametrů obrazu snímaných kamerou, především o kontrolu správných rozměrů, tvaru a pozice objektů na snímané scéně. Klasické řešení videometrické aparatury sestává z kamery, počítače a vyhodnocovacího softwaru.

Cílem této diplomové práce je vyvinout univerzální vyhodnocovací software, který umožní uživateli názorné proniknutí do problematiky videometrie a následný vývoj včetně realizace konkrétních videometrických úloh. To vše při zachování nízkých pořizovacích nákladů díky použití dostupné a rozšířené zachytávací karty s čipem Bt878 a operačního systému Linux. Samotná aplikace je napsaná v programovacím jazyce C++.

Videometrická aplikace postihuje čtyři hlavní kroky vedoucí ke změření určitého rozměru či k rozhodnutí o tom, zda má měřený předmět požadované vlastnosti.

Kalibrace měřícího systému ve vztahu k použitým objektivům, k úhlu pod kterým bude kamera snímat a ke zkreslení daném perspektivou.

Pořízení obrazu a jeho digitální interpretace v počítači pro další použití. V operačním systému Linux je k tomuto účelu určeno videorozhraní Video4Linux.

Zpracování obrazu, během kterého dojde ke zdůraznění důležitých obrazových informací a naopak k potlačení informací z pohledu měřené problematiky nedůležitých. Jedná se hlavně o nalezení hran a objektů na snímaném předmětu. K tomuto účelu aplikace používá volnou knihovnu počítačového vidění Intel OpenCV, která obsahuje velké množství pokročilých funkcí.

Změření a vyhodnocení vlastností předmětu z připravených obrazových dat. Pomocí statistiky a parametrizace nalezených hran a objektů lze dosáhnout až subpixelové přesnosti určení rozměrů. Ty se následně porovnají s požadovanými a dojde k vyhodnocení.

Klíčová slova:

Počítačové vidění; videometrie; kalibrace kamery; knihovna Intel OpenCV

Resumme

Videometry is a method that enables the elimination of the human factor from measurements and interpretations of processes. It deals with a contactless measurement of a distance from parameters derived from an image taken by a camera and especially with the check of the proper dimensions, shape and positions of the object on the scanned area. A common arrangement of a videometrical apparatus is composed of a camera, a PC and an SW to interpret the obtained data.

The objective of this diploma thesis is to design a multi-purpose evaluating software which would provide to user a comprehensible penetration into the problem of the videometry and a subsequent development including the implementation of a particular solution of a videometrical problem. This should be obtained with low acquisition costs owing to the utilization of the current accessible grabber with a Bt878 chip and the Linux operating system. The actual application is written by C++ programming language.

The videometrical application covers the four main steps contributing to the measurement of a dimension or a decision whether the measured object has the desired properties.

The calibration of measuring systém and its connection to the employed objectives, the angle of the camera reference and the deformation of image caused by the perspective.

The image acquisition and its digital interpretation, which should be instantly used in the PC. For this purpose a video interface Video4L is used in the Linux operating system.

The image processing, during which the important image information is emphasized and the non – essential information smoothed according to the observed properties. The most important objective is to detect the edges and the components of the object. For this purpose the Intel Open CV free source computer vision library containing many advanced functions is used in our work.

The measurement and the interpretation of the properties of the object from the obtained data. Using a statistical and a parametrical description of detected edges

with a high sub-pixel accuracy in the determination of dimensions can be reached. These dimensions are then compared with desired ones and the evaluation is made.

Key words:

Computer vision; videometry; camera calibration; Intel OpenCV library

Obsah

<i>Úvod</i>	<i>11</i>
1 Počítačové vidění	13
1.1 Videometrie	15
2 Požadavky na snímanou scénu	17
2.1 Objektiv	17
2.1.1 Dírková komora a tenká čočka	17
2.1.2 Hloubka ostrosti	18
2.1.3 Speciální typy objektivů	19
2.2 Snímače	21
2.2.1 CCD	22
2.2.2 CMOS	22
2.3 Kamery	22
2.3.1 Analogové kamery	23
2.3.2 Digitální kamery	24
2.4 Osvětlení	24
2.4.1 Zdroje světla	26
2.4.2 Používané způsoby osvětlení	27
2.5 Pozadí	29
3 Požadavky na videometrickou aplikaci	30
3.1 Komerční videometrická aplikace NI Imaq Vision Builder AI	31
4 Zachytávací karta	34
4.1 Karta AverMedia EZCapture s čipem BT878	36
5 Softwarové nástroje	38
5.1 Video4Linux	38
5.1.1 Použití V4L	39
5.2 OpenCV	42
5.2.1 Instalace knihovny a kompilace programů	43
5.3 Vývojové prostředí KDevelop a knihovna QT	45
6 Model perspektivního zobrazení	47
6.1 Matice kamery	48
6.2 Matice vnějších parametrů	49
6.3 Zkreslení skutečného objektivu	49
7 Kalibrace kamery	51
7.1 Kalibrace ze známé scény	51
7.2 Kalibrace z neznámé scény	52
8 Vývoj aplikace	53

8.1	Pořízení obrazu	53
8.2	Volání OpenCV funkcí	53
8.3	Kalibrace kamery	54
9	Závěr	56
	<i>Seznam použité literatury</i>	<i>57</i>
	<i>Seznam obrázků</i>	<i>59</i>
	<i>Seznam tabulek</i>	<i>59</i>
	<i>Seznam příloh</i>	<i>60</i>

Úvod

Diplomová práce se zabývá vývojem videometrické aplikace pro použití v Laboratoři počítačového zpracování obrazu (LDIP) na Katedře měření TUL. Výzkumná činnost laboratoře spočívá v aplikacích metod zpracování a analýzy obrazu a metod počítačového vidění ve výuce a v průmyslu.

Technické i programové prostředky k realizaci přenosu obrazu do počítače dodává mnoho firem. Zpracovávat analogové nebo digitální signály je možné pomocí styčných externích modulů nebo zásuvných karet připojovaných ke sběrnici počítače. Pro efektivní zpracování, vyhodnocení a případnou analýzu obrazu počítačem je vhodné použít speciální videometrický software. Jeho dodavateli jsou například firmy National Instruments (NI Imaq Vision Builder AI), Siemens (Simatic Spectation), Cognex (DVT Framework). Strategie firem dodávajících tento software se v některých případech vyznačuje nepříjemně úzkou podporou pouze vybraných typů externích zařízení a specializací na jediný operační systém. Diplomová práce si klade za cíl zpřístupnit videometrii pro platformu operačního systému Linux. A zároveň se zbavit závislosti na použitých technických prostředcích.

V první části se diplomové práce zabývá obecnými principy videometrie a slouží jako formulace požadavků na vyvíjenou aplikaci. Zároveň poskytuje teoretický základ pro budoucího uživatele, protože dobrých výsledků je dosaženo jen tehdy, jsou-li vhodně nastaveny parametry jak pro pořízení obrazu, tak pro jeho zpracování.

Dále práce detailně popisuje použitý hardware v podobě zachytávací karty založené na čipu Bt878. Zabývá se podporovanými formáty přenosu obrazu dle způsobu připojení kamery (kompozitní/komponentní video), použitých televizních norem (NTSC/PAL) a skládání obrazu při prokládaném řádkování.

V kapitole Softwarové nástroje je nejdříve popisováno videorozhraní Video4Linux, které zprostředkovává komunikaci mezi videometrickou aplikací a zachytávací kartou. Dále lze nalézt seznámení s knihovnou Intel OpenCV a rozdělení na jednotlivé její části z pohledu použití. Nakonec je stručně popsáno vývojové prostředí KDevelop vhodné pro vývoj aplikace v prostředí operačního systému Linux.

Následuje důležitá kapitola zabývající se metodami kalibrace kamery. V ní jsou teoreticky popsány hlavní postupy vedoucí k určení matic perspektivního zobrazení.

V poslední kapitole je diplomová práce zaměřena na postupný vývoj videometrické aplikace v implementaci vybraných algoritmů.

1 Počítačové vidění

Počítačové vidění, někdy též nazývané jako strojové vidění, je rychle se rozvíjející obor průmyslové automatizace, který se uplatňuje hlavně v systémech řízení jakosti. S rozvojem požadavků na absolutní jakost dodávky se nahrazuje statistická kontrola úplnou kontrolou každého vyrobeného kusu. Klasické metody kontroly by v tomto případě vyžadovaly enormní finanční i lidské zdroje. Proto je třeba kontrolu automatizovat.

Současný prudký rozvoj ve sféře spotřební elektroniky v oblasti fotoaparátů a videokamer je zacílený na požadavky široké veřejnosti, ale otevírá i cestu k levným a přitom spolehlivým technickým aplikacím v různých oborech. Hnacím motorem pro urychlení využití obrazové informace je současný stav výpočetní techniky. Cesta, která využívá produkty spotřební elektroniky pro technické účely, má však i své zápory v podobě snížení přesnosti.

Postupy počítačového vidění jsou značně složité. Stěžejním úkolem je interpretace a porozumění obrazových dat, která jsou nejčastěji reprezentována symbolicky. Jádrem pokročilejších postupů jsou metody rozpoznávání, znalostní systémy a techniky umělé inteligence. Tato část počítačového vidění je nazývána vyšší úroveň.

Cílem nižší úrovně počítačového vidění je poskytnout kvalitativní symbolickou informaci potřebnou pro vyšší úroveň. K tomu slouží nejrůznější metody analýzy a zpracování obrazu. Postupy nižší úrovně jsou také bezprostředně použitelné pro odstraňování šumu z obrazu či rozpoznávání jednoduchých objektů v obraze.

Postup zpracování a rozpoznávání obrazu je možné rozložit do posloupnosti základních kroků [1]:

- snímání, digitalizace a uložení obrazu v počítači
- předzpracování
- segmentace obrazu na objekty
- popis objektů
- porozumění obsahu obrazu (klasifikace objektů)

Snímání a digitalizace

Při snímání se většinou převádějí vstupní kvantované optické veličiny na elektrický signál, se kterým se dále pracuje jako se spojitým v čase i úrovni. Vstupní informací může být jas z kamery, intenzita rentgenového záření, ultrazvuk, tepelné záření aj. Snímat se může v jednom nebo více spektrálních pásmech. Digitalizací se vzorkuje a kvantuje vstupní spojitý signál odpovídající monochromatickému obrazu do diskrétního tvaru. Ten je nejčastěji popsán funkcí dvou proměnných tj. souřadnic v obraze. U řádkových kamer to ale bývá jednorozměrný jasový profil.

Předzpracování obrazu

Cílem předzpracování je potlačit šum a zkreslení vzniklé nejen při digitalizaci a přenosu dat. Jindy se předzpracování snaží zvýraznit určité rysy obrazu podstatné pro další zpracování a ostatní naopak potlačit. Příkladem může být hledání hran v obraze, tj. obrazových bodů (edgelů) s vysokými hodnotami velikosti gradientu obrazové funkce.

Segmentace

Asi nejsložitějším krokem postupu zpracování je segmentace, která dovolí v obraze najít objekty. Za objekty se považují ty části obrazu, které nás z hlediska dalšího zpracování zajímají. Segmentace představuje rozklad množiny bodů obrazu na podmnožiny podle jejich vlastností na základě znalosti interpretace obrazu [2].

Popis nalezených objektů

Nalezené objekty lze popsat buď kvantitativně pomocí souboru číselných charakteristik a/nebo kvalitativně pomocí relací mezi objekty. Způsob popisu objektů je ovlivněn předpokládaným použitím získaných dat. Za nejjednodušší popis objektů lze považovat např. stanovení plochy, obvodu či těžiště objektů.

Porozumění obsahu

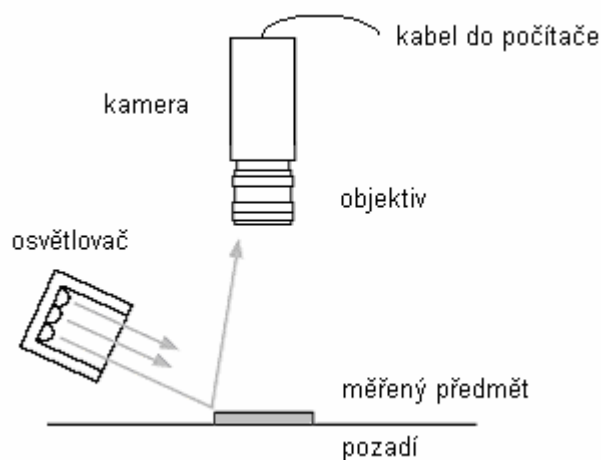
Ve velmi jednoduchém případě můžeme za porozumění považovat klasifikaci objektů v obraze podle jejich velikosti. Jen o málo složitější je klasifikace objektů do několika předem známých tříd, např. na hranaté a kulaté. V obecném případě představuje porozumění interpretaci obrazových dat, o kterých se předem nic nepředpokládá.

Porozumění obrazu je potom založeno na znalosti, tvorbě plánu, cílech, a využití zpětných vazeb mezi různými úrovněmi zpracování.

1.1 Videometrie

Videometrie je postup vedoucí ke změření určité fyzikální veličiny prostřednictvím obrazu, tedy bezdotykové měření pomocí implementace metod zpracování a analýzy obrazu. Dříve se hojně uplatňovala v astronomii, kde je často jedinou možností dozvědět se něco o vzdálených planetách. V současnosti lze spatřit největší rozvoj videometrie v prostředcích automatizace kontrolních činností. Příkladem může být kontrola, zda je v objektu vylisovaný otvor, či zda je správná rozteč mezi děrami. Další možností použití je kontrola úplnosti montážního celku, například kontrola vzájemné polohy různých součástí. Za hlavní výhody videometrie lze považovat:

- objektivitu analýzy
- rychlost vyhodnocení
- možnost regulace výrobního procesu
- možnost uložení výsledků (obrazy i naměřená data)
- automatické výpočty vlastností obrazu
- široký rozsah možností použití jednoho zařízení
- menší množství zaměstnanců, menší únava pracovníků



Obr. 1.: Měřicí sestava

Měřicí sestavu si lze představit pomocí Obr. 1. Je tvořena měřeným objektem, nebo alespoň jeho částí, kolem kterého se obecně nachází pozadí. Světelné paprsky odražené od objektu jsou objektivem promítány na světlocitlivý senzor kamery. Kamera může obraz zpracovat buď samostatně a nebo jej pošle například prostřednictvím zachytávací karty do počítače. V počítači je nutné mít softwarové vybavení pro zpracování a vyhodnocení obrazu.

2 Požadavky na snímanou scénu

Základním předpokladem pro úspěšné provedení obrazové analýzy je správné pořízení obrazu měřeného předmětu. Tento proces je ovlivňován mnoha faktory, z nichž nejdůležitějšími jsou osvětlení, podklad, typ objektivu a kvalita snímacích zařízení. Nejdříve je ale nutné se seznámit s fyzikálními principy vzniku obrazu.

2.1 Objektiv

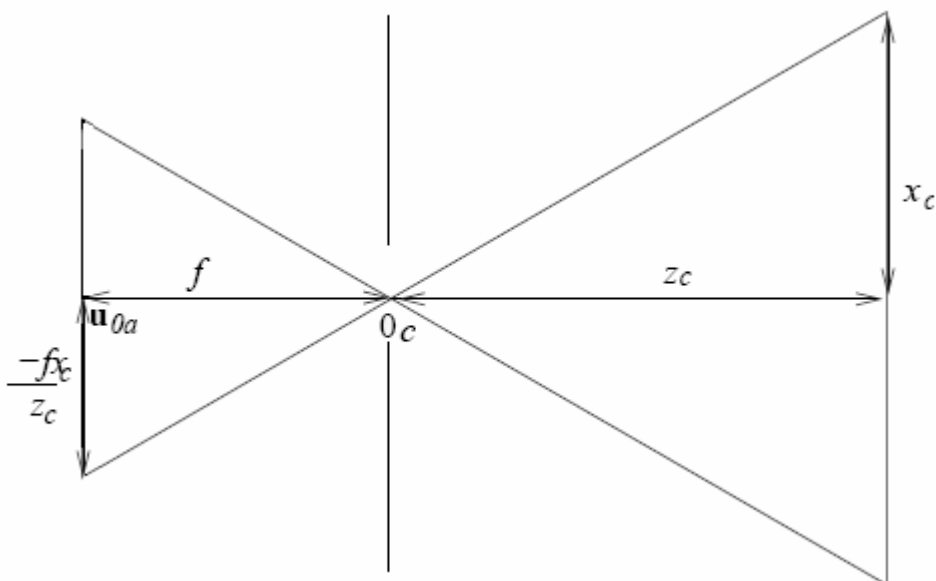
Mezi důležité parametry obrazu lze zařadit jeho ostrost. Je-li obraz neostrý, nelze pak s dostatečnou přesností určit hranice měřeného objektu a tudíž nelze provést ani dostatečně přesné změření jeho rozměrů. Objektiv složí k promítnutí světelného záření na snímač pro převod na elektrický signál. Objektiv by měl co nejvěrněji napodobovat ideální projektivní zobrazování.

2.1.1 Dírková komora a tenká čočka

Dírková komora (angl. Pinhole), je nejjednodušším typem objektivu. Jedná se o jednoduché optické zobrazovací zařízení ve tvaru uzavřené skříňky či prostoru, v jehož jedné stěně je malý otvor, který na protilehlé stěně vytváří převrácený obraz vnějšího prostoru na základě přímočarého šíření světla.

Princip zobrazování dírkové komory znázorňuje Obr. 2. Každý bod na povrchu osvětleného předmětu odráží světelné paprsky všemi směry. Určitou část těchto paprsků dírka propustí. Tyto paprsky pokračují ve své dráze, až narazí na průmětnu, kde vytvoří převrácený obraz předmětu. Protože jde o skutečný středový průmět, mají obrázky v dírkové komoře dokonalé perspektivní podání [1].

Jinou zajímavou vlastností je naprostá hloubka ostroty, která umožňuje na jednom snímku zachytit stejně ostře zároveň předměty velmi blízké i velmi vzdálené.



Obr. 2.: Princip perspektivního zobrazení.

Dírková komora je nereálnou idealizací, protože velmi malou dírkou prochází i málo energie. Další překážkou v použití je potřeba dlouhých expozičních časů. Též je nutné zmínit i ohyb světla způsobující odchýlení směru záření pro objekty srovnatelné velikosti s vlnovou délkou záření. Pouze rovinná zrcadla přesně vyhovují tomuto modelu geometrické optiky.

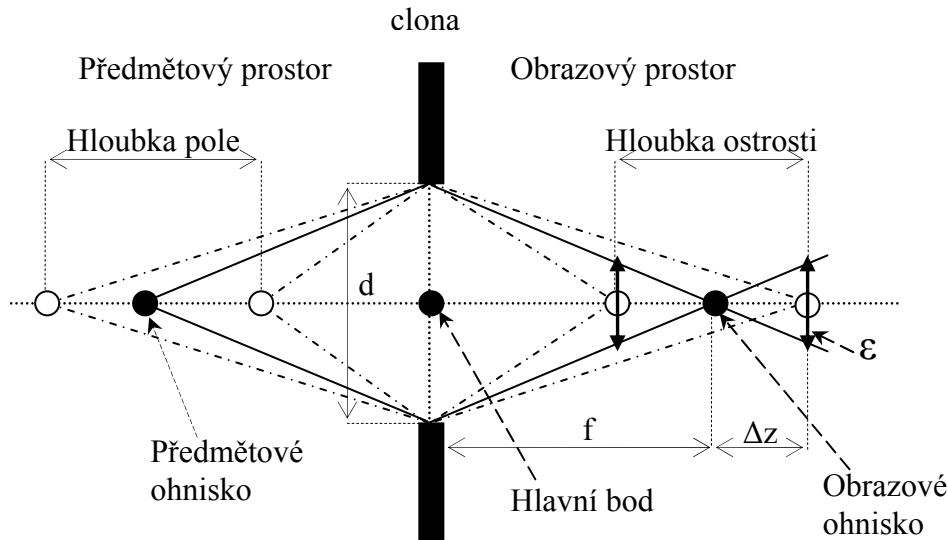
Částečně je princip dírkové komory použitelný pro ideální tenkou čočku, která je nejjednodušším objektivem. Paprsek procházející středem čočky (ohniskem) se neodklání. U správně zaostřené čočky se procházející paprsky jiných směrů lámou tak, že se protínají s paprskem procházejícím ohniskem v hlavním bodě.

Nedostatkem ideální tenké čočky je skutečnost, že soustřeďuje pouze paprsky vycházející z bodu v rovině kolmé k optické ose ve vzdálenosti z_c od středu promítání (viz.Obr. 2), kde f je ohnisková vzdálenost, tj. vzdálenost mezi hlavními body a ohniskem [3].

2.1.2 Hloubka ostrosti

Na rozdíl od dírkové komory, která zobrazuje blízké i vzdálené objekty stejně ostře, jsou reálné objektivy zaostřeny jen na konkrétní vzdálenost, respektive na určité okolí této vzdálenosti.

Posuneme-li obrazovou rovinu mimo obrazové ohnisko, obraz se rozostří. Představme si svazek paprsků vycházejících z předmětového ohniska, viz. Obr. 3. Bod se zobrazí jako malý kruh o průměru ε . Vzdálenost obrazové roviny od obrazového ohniska je Δz , d je průměr clony [1].



Obr. 3.: Hloubka ostrosti a hloubka pole

Pojmy hloubka ostrosti (angl. depth of focus) a hloubka pole (angl. depth of field), vycházejí z představy, že v aplikacích je přípustné, aby obraz byl poněkud rozostřený. Pro digitální obrazy je povolené rozostření úměrné velikostí pixelu. Čím je rozlišení obrazu větší, tím méně může být obraz rozostřen. Hloubka ostrosti je interval povolených posunů obrazové roviny, pro který je průměr kruhu rozostření menší než ε .

V literatuře je zaveden pojem f-číslo objektivu $n_f = f/d$ a udává, nakolik se objektiv liší od dírkové komory. Čím je aperturní clona menší, tím je hloubka ostrosti větší. Z hlediska použití optické soustavy je důležitá hloubka pole na předmětové straně optické soustavy, která udává rozsah přípustných poloh pozorovaného předmětu [1,3,4].

2.1.3 Speciální typy objektivů

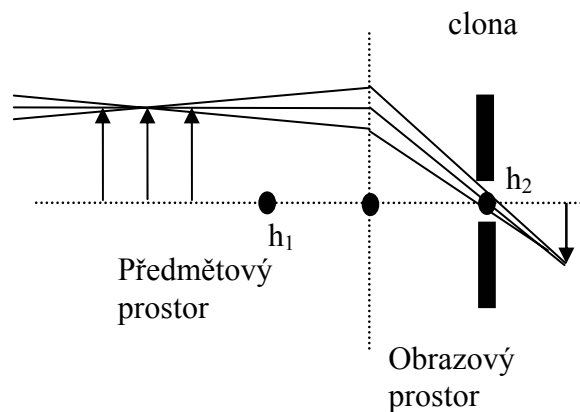
Skutečné objektivy lze z hlediska výpočtů nahradit čočkami, pro něž je možné základní model dírkové komory rozšířit. Hlavní rovina se nahradí dvěma hlavními rovinami. U tlusté čočky máme tedy dva hlavní body, předmětový a obrazový. Paprsek vcházející

do čočky přes předměťový hlavní bod vychází po průchodu systémem čoček z obrazového hlavního bodu pod stejným úhlem. Vzdálenost mezi hlavními body udává efektivní tloušťku tlusté čočky.

Výběr objektivu je dán značnou specifičností měřených objektů, zejména jejich rozměrem a požadovanou přesností měření. Rozlišujeme dva základní typy objektivů, telecentrický a hypercentrický [1].

Telecentrický objektiv

V případě kdy nelze zaručit, aby měřený předmět byl stále v obrazové rovině lze využít vlastností telecentrického objektivu, znázorněného na Obr. 4. Zde je malý aperturní otvor umístěn do obrazového hlavního bodu. Tím se dosáhne toho, že obraz vytvářejí pouze paprsky, které jsou přibližně rovnoběžné s optickou osou. V průmyslových aplikacích získávají telecentrické objektivy na významu. Jejich nevýhodou je, že průměr objektivu musí být přinejmenším tak velký jako měřený předmět a pro větší rozměry jsou objektivy drahé. Výhodné použití nacházejí telecentrické objektivy tam, kde se mění poloha objektu na optické ose nebo je objekt “tlustý” tj. nechceme, aby byl obraz ovlivněn bočními stěnami předmětu [1,3].

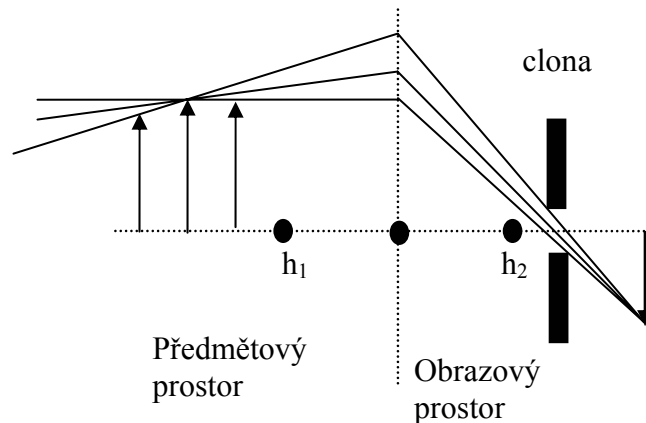


Obr. 4.: Telecentrický objektiv

Hypercentrický objektiv

Posuneme-li aperturní clonu mezi obrazový hlavní bod a obrazovou rovinu, dostaneme hypercentrické zobrazení u kterého jsou dokonce vzdálenější objekty zobrazovány jako větší. Schéma hypercentrického objektivu je uvedeno na Obr. 5. Na rozdíl

od telecentrického objektivu, který díky rovnoběžnosti paprsků pozoruje jen průřez, hypercentrický objektiv vidí průřez a vnější stěnu [1,3].



Obr. 5.: Hypercentrický objektiv

2.2 Snímače

Snímače světelné energie používané v kamerách lze rozdělit podle typu fotoelektrického jevu na [4]:

- fotoemisní snímače využívající vnější fotoelektrický jev. Přijetím fotonu je poskytnut dostatek energie, aby se uvolnil volný elektron. Principu se využívá ve vakuových snímacích elektronkách a fotonásobičích.
- fotovoltaiické a fotovodivostní snímače využívající vnitřní fotoelektrický jev. Energie fotonu způsobí, že elektron opustí svoje valenční pásmo. U fotovodivostních látek se dostane do vodivého pásma. U fotovoltaiických látek je vybuzený elektron zdrojem elektrického napětí.

Skutečný snímač má snímací část, paměťovou část a výstupní obvod. V elementech snímací části se akumuluje elektrický náboj, jehož množství je úměrné intenzitě dopadajícího světla a době osvětlení. Po uplynutí určitého času daného hodinovým taktem se náboj přemístí do paměťové části, která slouží k uchování obrazu až do jeho vyslání z obvodu. Mezitím se integruje náboj dalšího snímku.

V současnosti se používají snímače CCD a CMOS založené na vnitřním fotoelektrickém jevu.

2.2.1 CCD

CCD snímače (angl. Charge Coupled Devices) převádí světelnou energii na elektrický náboj pomocí Schottkyho fotodiod. CCD snímač je analogovým posuvným registrem, který sejmutý náboj přesouvá do vyhodnocovací elektroniky. Nevýhodou CCD snímačů je vzájemné ovlivňování nábojů v sousedních pixelech (angl. blooming), setrvačnost (tj. pomalé odeznívání silně ozářených bodů), malý rozsah intenzit a díky způsobu posouvání náboje nemožnost adresovat jednotlivé pixely. Výroba CCD elementů je komplikovanější a tudíž drahá. Výhodou CCD je nízký šum [4].

2.2.2 CMOS

Rozvoj technologie výroby polovodičových čipů umožnil vyrábět snímače také CMOS technologií a dosáhnout nízké ceny. Zároveň je možné na jednom čipu integrovat kameru i procesor zpracovávající obraz. Výhodou CMOS kamer je větší rozsah intenzit, velká rychlost vyčítání a přístup k pixelům podobně jako k paměťovým buňkám. Zároveň lze snadněji vytvořit čip s větším rozlišením. Nevýhodou je větší šum než u CCD snímačů [4].

2.3 Kamery

Kamera se skládá ze zařízení pro připojení objektivu, vlastního snímače a elektroniky pro převedení signálu ze snímače na nový signál pro další zpracování. Kamery mohou být vybaveny některými funkcemi, které jsou popsány dále [4]. Při použití kamer pro účely měření se doporučuje, je-li to možné, tyto funkce vypnout, aby nedošlo ke zkreslení signálu.

První takovou funkcí je AGC (angl. Automatic Gain Control) – automatické řízení zisku. AGC mění automaticky zesílení kamery, aby se výstup kamery nedostal do saturace, kterému odpovídá napětí o rozsahu $\langle 0,25V; 1V \rangle$. Součástí bloku AGC bývá také korekce intenzity přizpůsobující obraz logaritmické citlivosti oka na jas.

Další funkcí je funkce elektronické závěrky – ES (angl. Electronic Shutter). Ta umožňuje pevně nastavit dobu expozice nábojových prvků CCD, aby se snímač nedostal do saturace. Toto v podstatě znamená snížení citlivosti CCD snímače oproti základnímu lineárnímu režimu. Automaticky je opět vyhodnocování velikosti

videosignálu zajištěno pomocí funkce označené AES (angl. Automatic Electronic Shutter). Výhodou AES je možnost snímání při proměnném osvětlení.

Další funkce tzv. gama korekce je dána minulostí. V obrazové elektronice existovala exponenciální závislost mezi velikostí proudu elektronů dopadajících na luminofor a jeho jasem. Aby se tato nelinearita korigovala používal se obvod zvaný gama korekce. Tato korekce upravuje lineární závislost velikosti videosignálu na osvětlení CCD na nelineární a tím nelinearitu kompenzuje inverzní závislostí. Funkce gama korekce je vestavěna ve všech kamerách s výstupním videosignálem podle TV standardů.

Poslední nastavitelnou funkcí bývá aperturová korekce. Tato slouží ke zvýraznění optických přechodů (černá – bílá) tím, že k videosignálu přičítá jeho derivace. U barevných kamer bývá funkce AWB (angl. Automatic White Balance). Automatické vyrovnání bílé mění poměr barevných složek R a B vůči G.

Podle charakteru výstupního signálu se kamery dělí na analogové a digitální.

2.3.1 Analogové kamery

Analogové kamery generují úplný televizní signál, v němž jsou kromě údajů o intenzitě vertikální a horizontální synchronizační pulsy pro postupné vykreslování obrazu po řádcích. Rozklad obrazu může být s prokládanými řádky jako v televizi nebo s neprokládanými s vyšší snímkovou frekvencí. Potom se jedná o tzv. Progressive scan kamery.

Nejlepšího odstupů signál/šum (SNR) dosahuje kamera v základním lineárním režimu při maximálním osvětlení, obvykle více než 50 dB. Proto je při nižších úrovních osvětlení vhodné, pokud to kamera umožňuje, zesílení kamery zvětšit.

Signál je před vyvedením z kamery upraven hornopropustným (derivačním) filtrem zdůrazňujícím vysoké frekvence. Kompenzuje tak nežádoucí rozmazání při průchodu světla vzduchem a optikou projevující se jako dolnopropustný filtr. Dále je analogový televizní signál veden obvykle koaxiálním kabelem do zachytávací karty umístěné v počítači [1].

2.3.2 Digitální kamery

Digitální kamery se od analogových liší hlavně tím, že obsahují A/D převodník. Výhodou umístění převodníku do kamery je jediné převzorkování. Převodník nejčastěji kvantuje signál do 8 bitů. Na rozdíl od analogového přenosu jsou všechny synchronizační a obrazové signály zcela odděleny. Na vstupu počítače stačí jednoduchá adaptační karta, standardně vyráběná, a tedy levná, která zajišťuje nejenom přenos obrazových dat, ale současně umožňuje i konfiguraci kamery podle povelů z počítače a často dodává do kamery potřebné provozní napětí, aniž by k tomu byl zapotřebí další kabel.

Pro připojení slouží sériová rozhraní IEEE 1394 (FireWire) a USB nebo nejnovější kamerové rozhraní Camera Link. Sběrnice FireWire je vysokorychlostní sériová sběrnice pro izochronní přenos dat rychlostí až 400 Mbitů/s, hodně používaná ve spotřební elektronice, ale i ve výpočetní a automatizační technice. Její předností je izochronní přenos, při kterém jsou data dopravována předem definovanou rychlostí, což je zvláště důležité pro přenos obrazových signálů, které se musí přenést přesně ve stanoveném okamžiku. Rozhraní Camera Link se svou vysokou rychlostí přenosu obrazových dat až 810 MB/s a snadnou systémovou integrací se stalo preferovaným rozhraním pro kamery ve špičkovém segmentu trhu. Standard Camera Link byl zaveden jako nová technika rozhraní, aby bylo možno jednoznačně definovat rozhraní mezi kamerami a obvody pro zpracování obrazu. Jeho hlavní využití se očekává zvláště v kamerách vybavených rychlými snímači obrazu na bázi technologie CMOS, které mají vysoké rozlišení a poskytují daleko větší množství obrazových dat.

Všechno nasvědčuje tomu, že moderní inteligentní digitální kamery s výkonným sériovým rozhraním brzo nahradí v průmyslovém zpracování obrazů klasické kombinace analogových kamer a zachytávacích karet. Hlavním důvodem jsou lepší kvalita obrazu, větší flexibilita v použití a v neposlední řadě také nižší cena komplexního systémového řešení.

2.4 Osvětlení

Kromě volby vhodné kamery a optiky hraje klíčovou úlohu osvětlení zkoumaného objektu. Nevhodná volba osvětlení je častou příčinou pořízení nevhodného obrazu

a nepřesné interpretace po dalším zpracování. Žádný projekt strojového vidění nemůže být úspěšný bez znalostí a zkušeností v oblasti návrhu a volby osvětlovacího systému, který musí být v souladu s požadovaným cílem projektu.

Světlo je svou podstatou elektromagnetické vlnění, jehož vlastnosti lze exaktně popsat radiometrickými veličinami. Pro oblast viditelného světla se však historicky používají veličiny fotometrické: svítivost, světelný tok a osvětlení.

Světlo prochází opticky homogenním prostředím přímočaře. Jeho šíření lze geometricky reprezentovat paprsky, které mohou být v prostoru uspořádány různým způsobem. V osvětlovačích se nejčastěji setkáme se dvěma případy osvětlení [5]:

- difúzní (osvětlení rozptýleným světlem) – objekt osvětlen ze všech směrů;
- kolimované – svazek paprsků, které jsou v ideálním případě rovnoběžné.

Barva světla je důležitým parametrem pro snímání černobílou i barevnou kamerou. V úlohách, kde je nutné věrné barevné podání, se obvykle volí „bílé“ světlo, které se podobá světlu slunečnímu, tj. obsahuje všechny vlnové délky v intenzitách obdoby slunečnímu záření). V monochromatické kameře je úroveň jasu digitálního obrazu ovlivňována kombinací barvy světla a spektrální citlivostí kamery.

Potíže obvykle působí vyšší vlnové délky – červený okraj spektra, zejména neviditelné infračervené záření, které je silně obsaženo v žárovkových zdrojích. Tato část světelného spektra způsobuje horší „kresbu“ objektivu kamery, protože objektiv je korigován pro vlnové délky viditelného světla. Důsledkem může být zkreslený obraz, který nelze použít k zamýšlenému vyhodnocení.

Správnou volbou osvětlení lze z pohledu měřené problematiky zdůraznit důležité obrazové informace a naopak nedůležité potlačit. Například přesvětlením objektivu získáme velmi kontrastní obraz snímaného předmětu, nebo naopak přicloněním zvýšit hloubku ostrosti. Osvětlení objektu se dá měnit volbou zdroje světla, jeho regulací, přiblížením či vzdálením osvětlovače od objektu a úhlem odráženého světla.

Osvětlení objektu, zaclonění objektivu a doba expozice tvoří trojici parametrů, která ovlivňuje úroveň jasu, kontrast a případně barevné podání digitálního obrazu. Pro digitální obraz (optimální z hlediska úlohy strojového vidění) je nutné zvolit optimální kombinaci těchto tří parametrů. V praxi je vzhledem k dalším požadavkům

úlohy nutné nastavit tyto parametry tak, že výsledná kombinace je lepším či horším kompromisem. Proto je volba osvětlovače z hlediska poskytovaného světelného toku vždy součástí celkové koncepce řešení úlohy.

Některé zdroje osvětlení vyzařují světlo časově proměnné. Nejčastějším případem je periodicky proměnné světlo, které vydávají některé zdroje napájené z elektrické rozvodné sítě, např. zářivky. Potom je třeba zajistit, aby celkový osvit, (integrál světelné energie, která po dobu expozice dopadne na snímač) byl pro všechny snímky stejný, jinak by měly různou úroveň jasu. Obvykle se tohoto stavu dosáhne tím, že doba expozice je celistvým násobkem periody [5].

V případě snímání rychlých dějů s krátkými časy expozice lze periodicky proměnné osvětlení snímané scény využít. Potom se jedná o zábleskové či stroboskopické osvětlení. V závislosti na možnostech vzájemného propojení kamery s osvětlovačem synchronizujeme kameru na osvětlovači či naopak.

2.4.1 Zdroje světla

Při řešení videometrické úlohy je nutné se zamyslet nad volbou zdroje světla pro osvětlovače.

Žárovka je jednoduchý, dostupný a levný zdroj světla pro osvětlovače. Používají se hlavně halogenové žárovky, u kterých lze získat vyšší světelný tok a nižší podíl infračerveného záření. Přesto je bohužel často nutné potlačit infračervené záření pomocí tzv. infrazádržného filtru. Velkou nevýhodou halogenové žárovky je rychlé stárnutí spojené se snižováním světelného toku a krátká životnost. Po výměně žárovky je většinou třeba osvětlovač znovu nastavit; protože při výrobě žárovek lze jen velmi těžko udržet parametry jednotlivých kusů v rozumných tolerancích. Do osvětlovačů se většinou používají žárovky s vnitřním reflektorem. Mohou sloužit jako plošné zdroje světla, osvětlení však nebývá rovnoměrné.

Zářivky jsou velmi vhodným zdrojem bílého světla. Z hlediska stárnutí a životnosti na tom však nejsou o mnoho lépe než žárovky. Nevýhodou je i poměrně vysoké napětí potřebné k jejich provozu. Pokud je na závadu jejich periodicky proměnné světlo s frekvencí 100 Hz, lze je napájet z napěťových měničů, které pracují na vyšších kmitočtech. Zářivkovou trubici lze při výrobě tvarovat do spirál nebo meandrů. Může tak sloužit jako základ plošného zdroje světla, tak i pro kruhový osvětlovač.

Osvětlovače s **LED** jsou v současné době na vzestupu. Diody totiž nepotřebují vysoké napětí a jejich světelný tok se snadno reguluje. Při správném použití je jejich stárnutí pomalé a mají tedy dlouhou životnost. Čip diody LED je v podstatě bodovým zdrojem. Výsledný tvar světelného svazku diody LED závisí na způsobu zapouzdření, což je třeba brát v úvahu při konstrukci osvětlovače. Barevné diody je možné využít díky rozdílné citlivosti světelných senzorů na frekvence barevného spektra. Vyrábějí se i bílé diody spojením barevných.

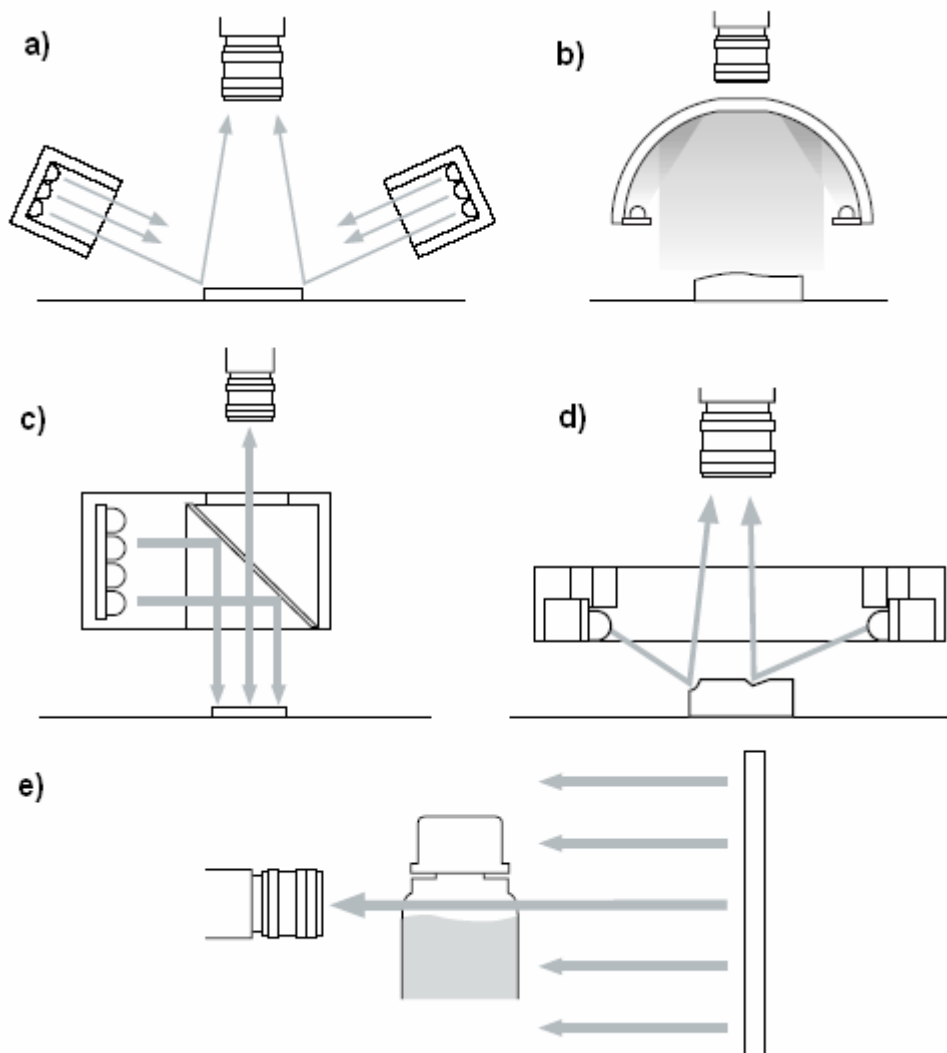
Jak již bylo zmíněno, je pro osvětlení objektů zásadní, aby všechny snímky měly konstantní osvit. Jen za této podmínky je možné očekávat opakovaně shodné řešení úlohy strojního vidění. Tuto zásadu lze snadno dodržet, je-li jediným zdrojem světla správně volený osvětlovač. V průmyslových provozech však téměř vždy působí i jiné zdroje světla: denní světlo a umělé osvětlení provozu. Jejich parametry nelze předvídat, protože závisí na denní a roční době, počasí a mnoha dalších okolnostech.

Nejjednodušším řešením je tyto nežádoucí zdroje světla odstínit. Pokud je to technicky nemožné, použije se osvětlovač s řádově větším světelným tokem, než mají nežádoucí zdroje. V důsledku toho se okolní osvětlení výrazně nepodílí na celkovém osvětlení snímané scény, protože jeho vliv je dán poměrem integrálu jasu okolního osvětlení a integrálu jasu záblesku po dobu otevření závěrky [4].

2.4.2 Používané způsoby osvětlení

Na Obr. 6. jsou představeny používané způsoby nasvícení předmětů pro konkrétní případy použití. V závorkách je uvedeno označení používané v anglicky psané literatuře [6].

Řídicí jednotka osvětlovače je v nejjednodušším případě určena k ručnímu nastavení velikosti světelného toku. Slouží i k pozdější kompenzaci změny osvětlení způsobené stárnutím některých typů světelných zdrojů. Při složitějších úlohách strojního vidění může být vyžadována změna osvětlení během řešení úlohy. Pak je třeba použít řídicí jednotky s vhodným komunikačním rozhraním, která může plynule měnit světelný tok osvětlovačů, vypínat a zapínat sekce světelných zdrojů (LED) v osvětlovačích nebo ovládat celé osvětlovací systémy sestavené z několika osvětlovačů. Zvláštním druhem řízeného osvětlení je také stroboskopické osvětlení, které se používá při snímání pohybujících se objektů [6].



Obr. 6.: Používané způsoby osvětlení.

- a) *Běžné s nepřímým světlem (angl. Bar-type). Pro silné osvětlení nelesklých předmětů.*
- b) *Kupolovité (angl. Dome-type). Difuzní bezstínové osvětlení pro trojrozměrné objekty.*
- c) *Souosé s polopropustným zrcadlem (angl. Coaxial vertical-type). Zdůrazňuje zrcadlové plochy kolmé na osu kamery, ostatní je tmavé.*
- d) *Pod nízkým úhlem (angl. Low angle-type, dark-field). Zdůrazňuje tenké nápisy či nerovnosti na lesklém povrchu.*

- e) *Zadní s protisvětlem (angl. Backlight-type). Časté pro měření vnějších rozměrů plochých předmětů a zviditelňuje neprůhledné části. Pro objekty jejichž tloušťka je nezanedbatelná zajistí kolimátor rovnoměrné paprsky a omezuje ohýbání světla.*

2.5 Pozadí

Na konec této kapitoly je nutné zmínit požadavky kladené na pozadí (podklad). Pozadí by mělo být voleno tak, aby co nejvíce kontrastovalo s analyzovaným vzorkem. Nejvhodnější bývá použití standardního černého nebo bílého podkladu, který je nejen dostatečně kontrastní, ale napomáhá také při definici bílé nebo černé barvy během analýzy. Aby byla zajištěna maximální eliminace stínů v obraze, je možné použít jako podklad semiš, jehož struktura většinu stínů pohltí. Ovšem při mikroskopických analýzách může být struktura tohoto materiálu naopak rušivým elementem. Snímání různobarevných vzorků je možné provádět na barevných podložkách, které mohou již v této první fázi analýzy odstranit některé nežádoucí části tím, že s nimi budou jen velmi málo kontrastovat. Pro analýzy, u nichž je vzorek nasvícen zdola, se jako podklad používají podložní sklička. Sklička mohou být buď čirá nebo v podobě difuzoru.

3 Požadavky na videometrickou aplikaci

Na videometrickou aplikaci je kladeno mnoho často protichůdných požadavků. V ideálním případě univerzálnosti použití na co největší rozsah videometrických úloh je zapotřebí, aby aplikace disponovala co největším množstvím funkcí a jejich vzájemným propojením k docílení užitečnosti. V tomto ohledu je samozřejmě cesta vytváření specializovaných aplikací pro řešení konkrétní videometrické úlohy. Tato cesta přináší výhody v jednoduchosti návrhu a v doladění algoritmů pro dosažení co největší přesnosti.

V případě nasazení videometrické aplikace v průmyslu pro hromadné měření, je žádoucí, aby pro snímání obrazu podporovala co nejvíce snímacích zařízení. Mimo to by měla podporovat zpracování obrazů všech běžných formátů, popř. jejich vzájemné konverze.

Aby výpočty vzdáleností měly jasně definovanou interpretaci ve skutečných rozměrech, je nutné znát rozměry pixelu a zbavit obraz zkreslení vzniklé například nepřesností objektivu, tj. kalibrovat kameru. Pro některé metody zpracování obrazu je praktické je vybavit možností vyznačit oblast zájmu (ang. Region of Interest – ROI). Umožňuje vybrat jen část obrazu důležitou pro danou funkci, protože není vždy žádoucí zpracovávat celý obraz.

Před měřením je vhodné obraz přizpůsobit tak, aby se zvýšila jeho vypovídací schopnost, aniž by se vzniklý obraz výrazně vzdálil od méně názorného originálu. Lze toho docílit například pomocí geometrických a jasových transformací. Do první kategorie spadá například zvětšování/zmenšování, rotace a posun. V obecném smyslu se metody geometrické transformace používají při kalibraci obrazu.

Druhá kategorie je výrazně obsáhlejší a spadají do ní i úpravy barevných modelů jako HSI, HSV, HSL. Dále jsou do kategorie jasových transformací zahrnovány úpravy intenzity barev nebo převod barevného obrazu na šedotónový.

Dále můžeme jmenovat zvyšování kvality obrazu, které zahrnuje postupy úpravou jasu, kontrastu, vyrovnáním intenzit bodů (ekvalizací histogramu) nebo ostřením hran, potlačení šumu pomocí průměrování a dalších filtrů pro vyhlazení či zvýraznění zrnitosti v obraze. Další důležitá funkce je označována jako prahování (angl. treshold), kdy je obraz převeden na binární. Následuje práce s binárními obrazy jako například

aplikování matematické morfologie (eroze, dilatace, otevření a uzavření) a logických operátorů (and, or, xor, not).

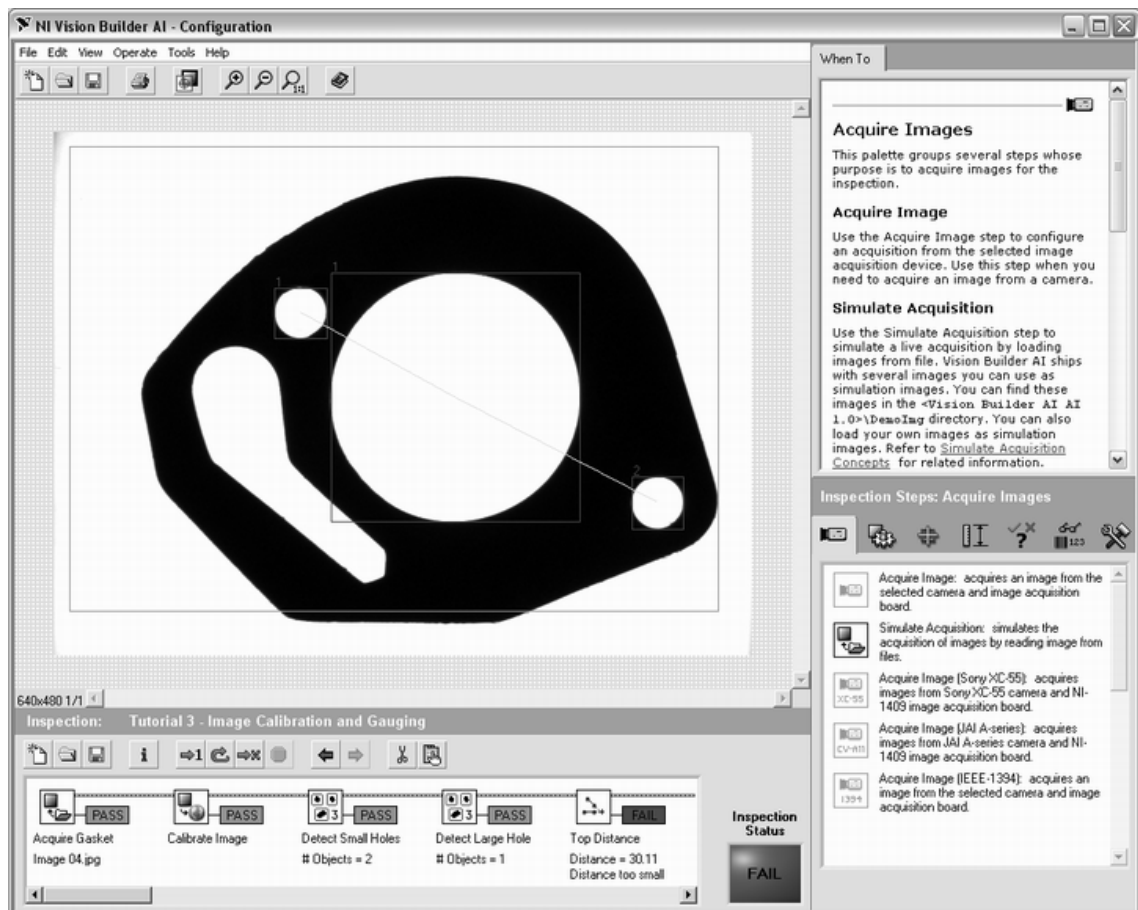
Na upravený obraz je možné používat různé metody nalezení hran a objektů. Nalezení objektu je kromě prahování možné podle předlohy (pomocí konvoluce) a podle spojitě oblasti podobného jasů okolí. Hrany je možné hledat pomocí Sobelova a Houghova operátoru, které jsou založené na hledání obrazových bodů s vysokými hodnotami velikosti gradientu obrazové funkce

Nalezené objekty a hrany je poté nutno parametrizovat, vytvořit matematický model, a provést samotné měření.

Z uživatelského hlediska by měla být videometrická aplikace intuitivní a příjemná na ovládání, tj. přehledná se snadnou editací. Vždy by měla zobrazovat původní i upravovaný obraz. Důležitá je vybavenost nástroji jako je lupa, zobrazení měřítka, ruční měření barvy a polohy kurzoru, relativní vzdálenosti mezi dvěma kurzory, návrat a opakování posledního úkonu. Nalezené objekty či hrany je vhodné zdůrazňovat pomocí kreslení bodů, úseček, obdélníků a kružnic. Je samozřejmostí, aby aplikace vytvořila měřicí protokol, jak pro jedno tak i pro sérii měření, pro které budou statisticky vyhodnoceny výsledky.

3.1 Komerční videometrická aplikace NI Imaq Vision Builder AI

Videometrická aplikace NI Imaq Vision Builder AI byla vyvinuta firmou National Instruments [7]. Je to konfigurovatelné vývojové prostředí, které dovoluje především měřit, propojovat a vyhodnocovat informace obsažené v obrazu. Umožňuje uživateli jednoduše zpracovávat obrazy a aplikovat vytvořené sledy funkcí sériově na velké množství obrazů bez nutnosti jich přeprogramování. V tomto programu je možné vyzkoušet různé strategie pro zpracování obrazu, okamžitě zobrazit výsledky provedených změn parametrů obrazu, otestovat jednotlivé funkce nebo jejich série i prohlížet si výsledky každého kroku při obrazové analýze. Použité funkce jsou ukládány do skriptu, který je možné jednoduchým způsobem převést na aplikaci a vytvořit virtuální měřicí přístroj.



Obr. 7.: Pracovní plocha programu NI Imaq Vision Builder AI.

Pracovní plocha programu je rozdělena na 3 části, viz Obr. 7. Největší část tvoří plocha pro zobrazování jednotlivých mezikroků zpracovávaného obrazu. Ve spodní části je umístěna posloupnost kroků. Kroky na sebe navazují a jsou představovány ikonami. Pod nimi se nachází název prováděné operace, informace o její úspěšnosti a v případě měřících operací i číselný výsledek. Poslední část se nachází vpravo od předchozích a umožňuje vybírat a vkládat připravené operace do prováděného skriptu. Tyto operace jsou rozdělené do 7 kategorií popsanych níže, které v zásadě vycházejí z rozdělení v kapitole 1.

- **Načtení obrazu** z kamery nebo ze souboru
- **Předzpracování** v podobě kalibrace kamery (viz níže) a Image Assistant. Jedná se o samostatnou aplikaci určenou pro předzpracování obrazu a jeho následné použití pro samotné měření. Umožňuje vyzkoušení a použití nejrůznějších metod zpracování obrazu a zdůraznění potřebných informací. Například lze

uvést různé filtry, matematickou morfologii, barevné a šedotónové prahování nebo hranové detektory.

- **Nalezení objektů** nebo jejich částí pro následující krok.
- **Měření vzdáleností a úhlů.**
- **Zjištění přítomnosti objektů** nebo jejich částí.
- **Čtení čárových kódů nebo textu.**

Vyhodnocení měření.

Při výběru operace se objeví jednoduchá nápověda a dialogové okno s několika záložkami pro vyplnění potřebných údajů. V první je možné nastavit základní informace jako název prováděného kroku či výběr oblasti zpracovávání (ROI). V dalších bývá volba parametrů měření jako například výběr prahu na uvedeném histogramu. Poslední záložka slouží k nastavení limitních hodnot pro rozhodnutí, zda měřený objekt splnil požadavky na něj kladené v konkrétním kroku měření. Toto nastavení je velmi důležité při hromadném měření, aby bylo možné oddělit například špatný výrobek od dobrých. Na základě této informace může vznikat měřicí protokol a také může být vyveden signál sériovým portem pro ovládání informačního majáčku či nějakého třídícího mechanismu.

Kalibraci kamery je zde možné provést třemi způsoby. Nejjednodušší je založen na pouhém nastavení rozměru pixelu v ose x a y. Druhý způsob umožňuje definování perspektivního zkreslení pomocí čtyř korespondujících bodů v obraze a reálném světě. K těmto bodům v obraze je nutné zadat jejich vzájemnou polohu ve skutečných rozměrech na měřené rovině a úhel, který svírá optická osa kamery s měřenou rovinou. Poslední a nejpropracovanější metodou je sejmutí obrazu kalibračního obrazce na měřené rovině. Kalibrační obrazec je tvořen maticí bodů o známé vzdálenosti. Tato metoda navíc umožňuje potlačení zkreslení vlivem objektivu.

Na konci této kapitoly je nutné uvést, že program IMAQ Vision Builder přímo spolupracuje pouze s grabberem od firmy National Instruments. Při použití jiného grabberu je nutné nejprve vytvořit snímek měřené scény v programu spolupracujícím s daným grabberem a poté tento snímek načíst ze souboru. To ovšem znemožňuje provádění automatizovaného měření.

4 Zachytávací karta

Zachytávací karty (angl. frame grabber) slouží jako rozhraní mezi analogovou kamerou a počítačem. Jejich základní funkcí je digitalizace video signálu. Umožňují tedy připojit k počítači kameru a pomocí softwarových rozhraní pracovat s jejím obrazovým výstupem.

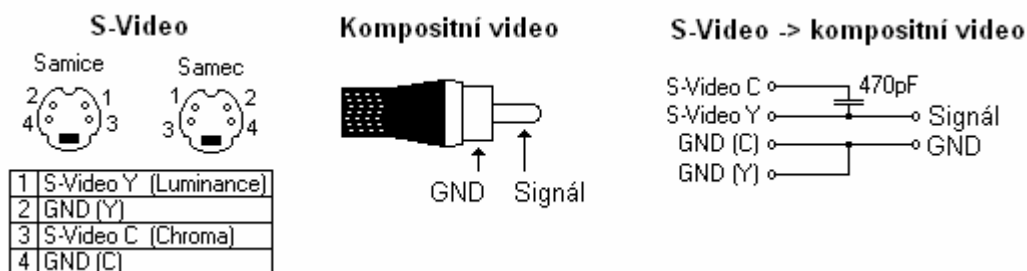
Zachytávacích karet je mnoho druhů a jejich možnosti jsou dány převážně parametry kamer k nim připojených. Nebereme-li v úvahu digitální kamery, nejčastěji používané analogové kamery vysílají dle některé z televizních norem.

V Tab. 1 je uveden přehled parametrů používaných televizních norem. Snímková frekvence je dána použitou frekvencí veřejné elektrické sítě v Evropě (50Hz) a USA (60Hz). Historicky se nejdříve objevily normy černobílé, které byly později rozšířeny na barevné při zachování zpětné kompatibility. Řádky, které jsou zdánlivě navíc oproti rozlišení, slouží jako synchronizační pro správné demodulování signálu [4].

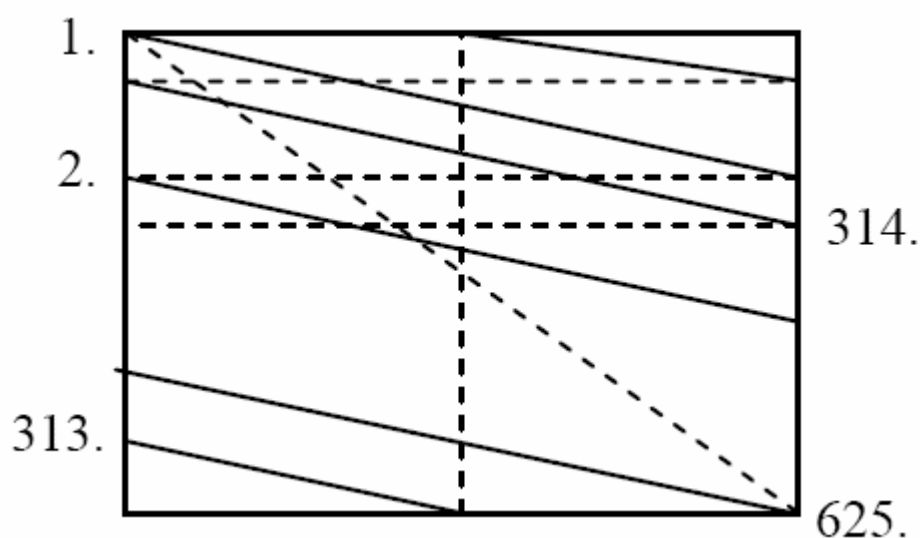
Tab. 1.: Přehled parametrů používaných televizních norem

TELEVIZNÍ NORMA	OBRAZ	REGION	SNÍMKOVÁ FREKVENCE	POČET ŘÁDKŮ	ROZLIŠENÍ
RS-170 (EIA)	černobílý	USA	60 Hz	525	648x484
CCIR	černobílý	Západní Evropa	50 Hz	625	768x576
NTSC	barevný	USA	60 Hz	525	648x484
PAL	barevný	Západní Evropa	50 Hz	625	768x576
SECAM	barevný	Francie/Východní Evropa	50 Hz	625	768x576

Nejčastěji se používá kompozitní video, kde je videoinformace nesena jedním signálem o úrovni $1V/75\Omega$. Na kratší vzdálenosti a pro zvýšení přesnosti se může použít video komponentní, které přenáší zvlášť jasovou (Y) a barevnou složku (C), viz Obr. 8. Kamera vyčítá z CCD snímače celý snímek najednou se snímkovou frekvencí, ale lichý a sudý pulsnípek jsou posílány prokládaně s pulsnímkovou frekvencí dle Obr. 9.



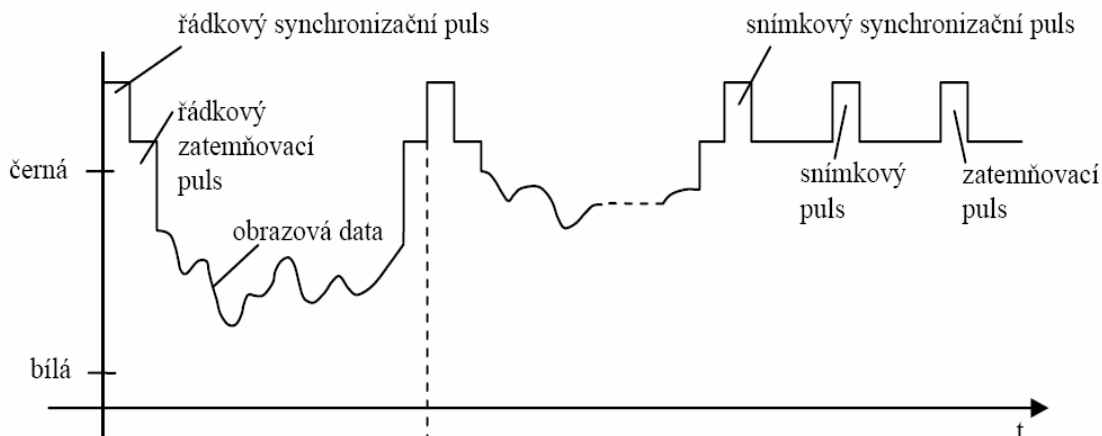
Obr. 8.: Přehled zapojení konektorů a jejich konverze



Obr. 9.: Způsob skládání obrazu při prokládaném řádkování

V jednoduchých kamerách bývá z ekonomických důvodů počet fotoelementů nižší než požaduje norma. V tomto případě se vlastně jedná o analogovou interpolaci. Informace z jednoho řádku je AD převodníkem zachytávací karty vzorkován na nastavené rozlišení.

Řídící impulsy se nepřenášejí žádnými dalšími kanály, ale vkládají se ve formě synchronizačních impulsů do videosignálu. Na začátku každého televizního řádku je řádkový (horizontální) synchronizační impuls a řádkový zatemňovací impuls. Pro případ černobílého obrazu je situace znázorněna na Obr. 10. Nejsou tedy synchronizovány jednotlivé pixely v řádku. U obyčejných kamer se nelze úplně spolehnout ani na časování, a proto se může stát, že jednotlivé řádky vůči sobě takzvaně plují. Anglicky se toto označuje jako jitter [1].



Obr. 10.: Průběh televizního signálu

4.1 Karta AverMedia EZCapture s čipem BT878

Tato karta je osazena čipem BrookTree BT878, který je velmi rozšířený a najdeme jej i na většině televizních karet. V linuxové platformě je čip BT878 podporován v softwarovém rozhraní Video4Linux. To je dokonce přímo implementováno v novějších verzích Linuxového jádra a poskytuje dobrou možnost komunikace s tímto hardware. Fotografie karty je uvedena na Obr. 11. Karta je vybavena vstupy jak pro kompozitní (NTSC/PAL/SECAM), tak pro komponentní video (S-VIDEO). Vlastnosti karty jsou dány převážně možnostmi čipu BT878. Jedná se o kompletní jednočipové řešení pro zachytávání zvuku a videa pro sběrnici PCI s podporou DMA pro nízké odezvy [8].



Obr. 11.: Karta AverMedia EZCapture s čipem BT878

Tato karta odporuje dekódování všech běžně používaných televizních norem a to NTSC/PAL/SECAM a jejich starších černobílých verzí až do rozlišení 768x576. Doplněním o TV tuner lze jednoduše zpřístupnit televizní vysílání počítači ať už v integrované podobě televizních karet nebo externě například prostřednictvím videorekordéru. Bez ohledu na použití barevné nebo černobílé kamery, zachytávací karta vždy demoduluje signál z kamery jako tříkanálový barevný obraz.

Karta AverMedia EZCapture byla zvolena zejména pro svou finanční dostupnost a dobrou softwarovou podporu. Druhý aspekt je velmi důležitý pro jiné operační systémy než je MS Windows. Jedná se spíše o spotřební elektroniku než o vědecký přístroj a proto není možné očekávat vyšší přesnosti.

5 Softwarové nástroje

5.1 Video4Linux

Video4Linux (dnes již ve verzi 2, označované V4L2) je aplikační programové rozhraní (API) Linuxu pro načítání obrazu a zvuku ze zařízení jako jsou televizní, rádiové a zachytávací karty. Toto rozhraní představuje ovladače hardwaru, které jsou implementované jako moduly jádra videodev, který zprostředkovává API dalším programům. Video4linux ovladače nepodporují vícenásobné čtení nebo zapisování stejných datových proudů ze zařízení. Pokud si jeden proces zpřístupní tento systémový prostředek, může s ním pracovat pouze on sám a ostatní konkurenční procesy musí čekat na jeho uvolnění. Toho se dosahuje například použitím softwaru, který již běží v uživatelském režimu [9].

Rozhraní V4L zprostředkovává komunikaci se zařízeními prostřednictvím souborů, které vytváří v adresáři `/dev` dle Tab. 2. Nejedná se soubory v pravém slova smyslu, ale o jakési zjednodušení, aby bylo možné se zařízeními pracovat podobně jako se soubory. Každé podporované zařízení je tedy představováno vlastním souborem, ze kterého lze načítat a zapisovat potřebné údaje. Na tomto principu je založen celý OS Linux.

Tab. 2.: Přiřazení souborů k V4L zařízením.

Jméno	Číslo zařízení	Funkce
<code>/dev/video</code>	0-63	Rozhraní pro zachytávání videa
<code>/dev/radio</code>	64-127	AM/FM rádiová zařízení
<code>/dev/vtx</code>	192-223	Čipy teletextového rozhraní
<code>/dev/vbi</code>	224-239	Surová VBI Data (Intercast/teletext)

K výměně dat mezi ovladačem zařízení a aplikačním programem je nutno mít standardní formáty obrazových dat, které budou obě strany stejně interpretovat. Video4linux obsahuje několik takovýchto standardních formátů, které jsou všechny

nekomprimované. Existují dvě základní skupiny formátů obrazových dat, RGB a YUV. Liší se od sebe v metodách reprezentace obrazových signálů. RGB reprezentuje formát, který jednotlivé pixely vytváří složením tří základních barev (červené, zelené, modré). Naproti tomu ve formátu YUV představuje Y jasovou a U, V barevné složky pixelu. Mezi RGB a YUV existuje přímý převodní vztah daný rovnicemi:

$$Y = (0.257 * R) + (0.504 * G) + (0.098 * B) + 16 \quad (1)$$

$$U = (0.439 * R) - (0.368 * G) - (0.071 * B) + 128 \quad (2)$$

$$V = -(0.148 * R) - (0.291 * G) + (0.439 * B) + 128 \quad (3)$$

5.1.1 Použití V4L

Při programování komunikace s V4L zařízeními je nutné znát jeho pravidla a hlavní přenášené struktury. Postup je rozdělen do následujících kroků [10]:

- Otevření zařízení (souboru)
- Volba video a audio vstupu, volba video standardu a nastavení dalších vlastností
- Volba formátu dat
- Volba vstupně výstupní metody
- Přenos aktuálních dat
- Zavření zařízení

V následujícím textu budou vysvětleny některé zásady správného použití.

Dotaz na schopnosti zařízení VIDIOCGCAP

Volání **VIDIOCGCAP** je používáno pro zjištění schopností video zařízení. Příkazu `ioctl` je poslána proměnná strukturního typu `video_capability` (popsána níže). Tato struktura je naplněna příslušnými hodnotami a vrácena. Obsahuje následující informace:

<code>name[32]</code>	Název tohoto zařízení
<code>type</code>	Druh zařízení
<code>channels</code>	Počet rádio/tv kanálů

audios	Počet audio zařízení
maxwidth	Maximální šířka získaného obrazu v pixelech
maxheight	Maximální výška získaného obrazu v pixelech
minwidth	Minimální šířka získaného obrazu v pixelech
minheight	Minimální výška získaného obrazu v pixelech

Příklad volání funkce:

```
{
int fd;

struct video_capability capability;

fd = open("/dev/video0", O_RDWR)

ioctl(fd, VIDIOCGCAP, &capability)

...
}
```

Nejdříve pomocí příkazu *open* otevřeme soubor zařízení */dev/video0* v módu *O_RDWR* – čtení a zápis. V proměnné *fd* je uloženo číslo zařízení. Poté zavoláme funkci *ioctl*, která nám vrátí ve struktuře „*capability*“ vlastnosti zařízení.

Načtení obrazu

Pro načtení obrazu ze zařízení je potřeba nejprve namapovat soubor zařízení do paměti RAM. Tím se data, která ze zařízení vystupují, kopírují do této oblasti paměti. Tato data potom v každém okamžiku reprezentují obraz, který zařízení snímá [10].

Nejprve použijeme příkazu *VIDIOCGMBUF*. Tento příkaz vrátí velikost bufferu paměti, kterou potřebujeme a offset paměti uvnitř bufferu pro každý obrázek. Počet obrázků, které zařízení podporuje záleží na daném zařízení a může to být pouze jeden.

Dalším krokem je pomocí příkazu *mmap* propojit výstup video zařízení s oblastí v paměti. Poté příkazem *VIDIOCMCAPTURE* dáme signál ke zkopírování dat ze zařízení do namapované oblasti paměti. Příkaz *VIDIOCSYNC* synchronizuje snímky – zajišťuje, že k pokusu o zpracování dalšího snímku dojde až v době, kdy je současný snímek už zpracován.

Příklad namapování zařízení

```
struct video_mmap device_map;

struct video_mbuf buffer;

unsigned char * map_;

ioctl(fd,VIDIOCGMBUF,&buffer);

map_ = (unsigned char*) mmap(0, buffer.size, PROT_READ,
MAP_SHARED,fd,0);
```

Nastavení parametrů obrazu (pokračování příkladu)

```
device_map.format = pal;

device_map.frame = 0;

device_map.width = 640;

device_map.height = 480;
```

A nakonec příkaz k zachycení obrazu a synchronizace

```
ioctl(fd,VIDIOCMCAPTURE,&device_map);

ioctl(fd,VIDIOCSYNC, &device_map.frame);
```

V oblasti RAM, na kterou ukazuje proměnná *map_*, je nyní uložen obraz, který zařízení načetlo.

5.2 OpenCV

Knihovna OpenCV (celým názvem Open Computer Vision Library) nabízí vysokoúrovňové funkce pro počítačové vidění a zpracování obrazu. Původně byla vyvíjena více než dvaceti vědci převážně ruské národnosti sponzorovanými společnostmi Intel. Knihovna je volná s otevřeným kódem a tím je zajištěný její budoucí vývoj díky mnoha přispěvatelům. Licenční podmínky ji umožňují používat i v komerčních aplikacích. Je multiplatformní a tedy nezávislá na operačním systému a použitém hardwaru. Pro OS Windows je dodávána jako dynamicky linkovaná knihovna (DLL) a v OS Linux je představována jako sdílený objekt (shared object SO). Je optimalizovaná pro použití v aplikacích závislých na zpracování v reálném čase. Obsahuje více než 350 funkcí a několik datových typů [11].

OpenCV obsahuje rozhraní pro napojení na knihovnu IPP. IPP (Intel performance primitives) je komerční nízkoúrovňová knihovna pro podporu zpracování signálů a obrazů vyvinutá společností Intel, jejíž hlavní přínos spočívá v optimalizacích pro instrukce procesorů Intel jako je MMX (MultiMedia eXtensions) a SSE (Streaming Single instruction multiple data Extensions). Spojení obou knihoven dává uživateli velmi silný softwarový nástroj pro použití v počítačovém vidění. IPP ovšem podléhá komerční licenční politice společnosti Intel a dnes díky vývoji běžných kompilátorů nepřináší takový užitek [11].

OpenCV používá pro práci s obrazy a pro jejich uložení v paměti datovou strukturu `IplImage` vypůjčenou z knihovny IPL taktéž vyvinutou firmou Intel. Její detailní popis je uveden v příloze č.1. Je v ní například uveden počet pixelů, jejich orientace, počet kanálů (jeden pro šedotónový obraz a 3 pro barevný) a jejich bitová hloubka.

Knihovna OpenCV je složena již ze 6 modulů nazvaných: CV, CVaux, CXcore, Highgui, CVcam, ML a jejich detailnější popis, který najdeme v následujících odstavcích, čerpá z originální dokumentace.

- **CV** - přináší hlavní funkcionalitu celé OpenCV knihovny a je zaměřen na zpracování obrazu. Konkrétně lze rozdělit do následujících oblastí: Nalezení hran a rohů, geometrické transformace a následné interpolace, morfologické operace a filtrování, podpora histogramů a jejich úpravy jako normalizace a ekvalizace, nalezení předlohy v obraze, kalibrace kamery a 3d rekonstrukce.

- **CVaux** - zabývá se vytvářením 3D modelu světa snímaným pomocí dvou kamer, tzv. stereoviděním. Hledáním korespondenčních bodů ze dvou blízkých obrazů a jejich morfigem. Kalibrací dvou kamer a podporuje rozpoznávání pomocí Markovových Modelů.
- **CXcore** - obsahuje datové struktury a základní funkce pro jejich správu, podporu pro lineární algebru, vykreslování textu a křivek do obrazu, manipulaci s obecnými soubory a práci s chybami.
- **Highgui** - obsahuje základní podporu pro zobrazování v grafickém uživatelském prostředí a interakci s myší a klávesnicí, nahrávání a ukládání obrazu na disk, umožňuje pořizování obrazu z kamer prostřednictvím V4L a Firewire.
- **CVcam** - knihovna CVcam je určena pro zpracování proudu videodat z digitálních kamer.
- **ML** - Tato knihovna v originále nazvaná Machine Learning se stala součástí OpenCV verze 0.99 a věnuje se vyšší úrovni počítačového vidění. Je vybavena třídami a funkcemi pro statistickou klasifikaci představovanou například neuronovými sítěmi.

5.2.1 Instalace knihovny a kompilace programů

Jelikož není instalace knihovny *OpenCV* triviální a ani není popsána v dokumentaci je nutné ji podrobně popsat a zároveň předat mé zkušenosti. Dále je potřeba uvést, že pro následující kroky je nutné být přihlášen jako administrátor systému (root).

Knihovna je pro operační systém Linux dodávána formou balíčku zdrojových kódů, je tedy nutno jí před použitím zkompilovat. Kompilace přináší výhodu v optimalizaci softwaru na konkrétní hardware. Pro úspěšnou kompilaci je také nutné, aby byly k dispozici následující knihovny. Pro zobrazování oken se zpracovávanými obrázky byla staršími verzemi *OpenCV* vyžadována knihovna *Motif*, vyvinutá ještě v dobách komerčního Unixu, kterou je možné nahradit knihovnami *OpenMotif* nebo *Lesstif*. U *OpenCV* verze 0.97 a výše je použito knihovny *Gtk+*, na které je postaven správce oken *GNOME*. Dále je ještě zapotřebí mít knihovnu *ffmpeg*, která ale není bezpodmínečně nutná v případě, že nebudeme používat funkce pracující s videem. U uvedených knihoven se může stát, že jsou součástí některých distribucí operačního

systemu Linux. V opačném případě je nutné je vyhledat na internetových stránkách jejich tvůrců a doinstalovat.

Samotná instalace knihovny OpenCV spočívá ve stažení zkomprimovaného balíčku na adrese [12] a jeho rozbalení například v adresáři `/tmp`. Poté je třeba zadat do příkazového řádku 3 následující příkazy, jejichž provádění zabere několik minut.

```
./configure CFLAGS=-I/kde/najdu/ffmpeg/include CPPFLAGS=-  
I/kde/najdu/ffmpeg/include LDFLAGS=-L/kde/najdu/ffmpeg/lib  
  
make  
  
make install
```

V případě, kdy není nutno pracovat s videem stačí první příkaz `./configure` zadat bez parametrů s cestou ke knihovně `ffmpeg`. Po úspěšném nainstalování najdeme všechny části knihovny `OpenCV` v adresářích `/usr/local/lib` a `/usr/local/include`.

Nakonec je potřeba operačnímu systému sdělit, kde se tyto knihovny nacházejí, aby je bylo možné volat z dalších programů. Toto je možné provést několika způsoby vycházejícími z rozmanitosti jednotlivých distribucí operačního systému. Proto doporučuji podle potřeby vyzkoušet všechny způsoby postupně tak, jak jsou uvedeny [13].

1. Je potřeba v adresáři `/etc/ld.so.conf.d/` vytvořit soubor `opencv.conf`, ve kterém bude cesta ke knihovně (standardně `/usr/local/lib`) a spustit `ldconfig`.
2. Přidat cestu ke knihovně do souboru `/etc/ld.so.conf` a spustit `ldconfig`.
3. Přidat cestu ke knihovně do systémové proměnné `LD_LIBRARY_PATH` zadáním následujících příkazů:

```
LD_LIBRARY_PATH=/kde/je/nainstalováno/lib/cv:${LD_LIBRARY_P  
ATH }
```

```
export LD_LIBRARY_PATH
```

4. Obdobně přidáním cesty do systémové proměnné `PKG_CONFIG_PATH`.

```
PKG_CONFIG_PATH=/kde/je/nainstalováno/lib/pkgconfig:${PKG_CONFIG_PATH}
```

```
export PKG_CONFIG_PATH
```

Ověření provedeme pomocí příkazů (odpověď systému je pod nimi):

```
pkg-config --cflags opencv
```

```
$ pkg-config --cflags opencv  
-I/kde/je/nainstalováno/opencv/include/opencv
```

```
pkg-config --libs opencv
```

```
$ pkg-config --libs opencv -L/com/softs/opencv-cvs/cvs-dev-i686/lib -lcxcore -lcv -lhighgui -lcvaux
```

Úspěšnost instalace je možno ověřit zkompileváním přiložených vzorových příkladů, které se nacházejí v balíčku v podadresáři `/samples/c`. Nejjednodušeji to lze provést pomocí příkazu `sh ./build_all.sh`, který zkompileje všechny příklady. A nebo je možné zkompilevat příklady samostatně pomocí následujících příkazů:

```
gcc `pkg-config --cflags opencv` `pkg-config --libs opencv`  
-o příklad příklad.c
```

```
gcc `pkg-config --cflags --libs opencv` -o příklad  
příklad.c
```

```
gcc -I/usr/local/include -L/usr/local/lib -lopencv -  
lhighgui -lstdc++ příklad.c -o příklad
```

5.3 Vývojové prostředí KDevelop a knihovna QT

QT je programovací základ (framework) od firmy Trolltech, který řeší časté problémy při vývoji aplikací. Obsahuje mnoho předdefinovaných tříd objektů a funkcí, které umožňují rychlou a příjemnou tvorbu aplikací v C++. Navíc aplikace napsané v QT lze

kompilovat a použít na všech obvyklých platformách: Windows, Mac OS, Linux a jiných unixových systémech [14].

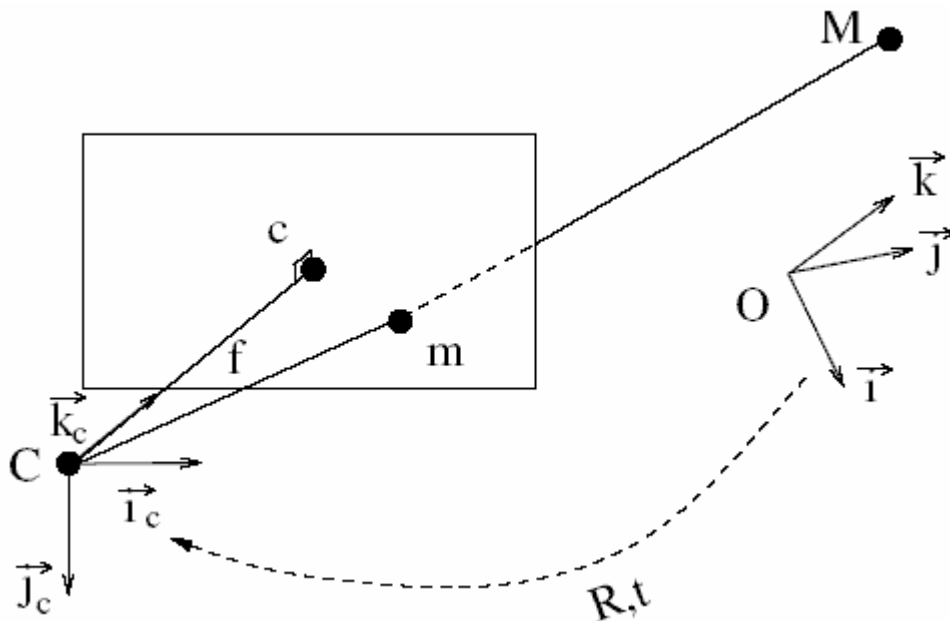
KDevelop je open-source integrované programové prostředí určené k vytváření aplikací pro KDE (rozšířený správce oken v OS Linux). Umožňuje však psaní, kompilování, ladění programů určených nejen pro KDE. Jeho výhodou je snadný vývoj aplikací s použitím knihoven QT, na nichž je KDE postaven [15].

V novějších verzích KDevelopu je integrován QT Designer určený ke snadnému grafickému návrhu uživatelského prostředí metodou táhni a pusť. Umožňuje programátorům návrh rozvržení jednotlivých oken, dialogů, nabídek a panelů nástrojů, které jsou třídami QT. Podobný návrh formulářů je znám z vizuálních vývojových prostředí pro Windows jako je např. Delphi nebo Microsoft Visual Studio. Zde vytvoříme jednotlivé formuláře jako třídy. Ty potom použijeme v KDevelop.

6 Model perspektivního zobrazení

V této kapitole je popsán matematický model perspektivního zobrazení, který používá knihovna OpenCV a jsou vysvětleny vlastnosti jednotlivých parametrů. V anglické literatuře se používá název Pinhole Camera Model, protože vychází ze středového zobrazení dírkové komory (angl. Pinhole). V obecné formě je také podrobně popsán v literatuře [16], kde jsou navíc definovány 4 souřadné systémy:

1. Souřadnice vnějšího světa — index w .
2. Kamerové souřadnice — index c , počátek ve středu štěrbině.
3. Obrazové souřadnice — index i , osy rovnoběžné s kamerovými souřadnicemi.
4. Affinní souřadnice obrazu — index a , osy u, v, w , kde osa u nemusí s osami v, w svírat pravý úhel.



Obr. 12.: Model perspektivního zobrazení, kde $(O, \vec{i}, \vec{j}, \vec{k})$ je světový systém souřadnic, $(C, \vec{i}_c, \vec{j}_c, \vec{k}_c)$ je kamerový systém souřadnic, M je bod ve 3D světě a m jeho projekce do obrazové roviny.

Vytvoření obrazu je definováno jako projekce 3D světa do obrazové roviny dle Obr. 12 [17,18]. Vztah mezi souřadnicemi bodu $M = [X, Y, Z]^T$ v euklidovském souřadném systému a jeho projekce do obrazového bodu $m = [u, v]^T$ je popsán rovnicí č. 4.

$$s\tilde{m} = P\tilde{M} \quad (4)$$

Body $\tilde{m} = [u, v, 1]^T$ a $\tilde{M} = [X, Y, Z, 1]^T$ jsou vyjádřením bodů m a M v homogenních souřadnicích, s představuje měřítko a P je projekční matice 3x4. Matice P definuje projekční transformaci pomocí 11 parametrů a lze ji rozložit na:

$$P = K[R | t] \quad (5)$$

kde:

- K je 3x3 matice kamery nebo také vnitřních parametrů (angl. camera matrix, intrinsic parameters).
- $[R | t]$ je 3x4 rotačně-posuvná matice někdy též nazývaná maticí vnějších parametrů (angl. extrinsic parameters). Popisuje transformaci mezi světovým a kamerovým souřadným systémem.

6.1 Matice kamery

Matice kamery je definovaná rovnicí č. 6 a popisuje vlastnosti kamery pomocí pěti vnitřních parametrů.

$$K = \begin{bmatrix} \alpha_u & s & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (6)$$

Koeficienty α_u a α_v jsou úměrné ohniskové vzdálenosti f a platí pro ně: $\alpha_u = k_u f$, $\alpha_v = k_v f$, přičemž k_u a k_v jsou číselným vyjádřením počtu pixelů na jednotku vzdálenosti zvláště pro osu u a v . Průsečík optické osy a obrazové roviny $c = [u_0, v_0]^T$, ideálně se nachází ve středu obrazu, je vyjádřen v obrazových souřadnicích. Koeficient s vyjadřuje zkosení, které je nenulové pouze v případě, že osy u a v nejsou na sebe kolmé.

6.2 Matice vnějších parametrů

Matice vnějších parametrů $[R | t]$ popisuje natočení a pozici kamery. Je tvořena rotační maticí R a posuvným vektorem t . Používá se pro popis pohybu kamery kolem statické scény, nebo obráceně pro pohyb objektu před kamerou.

$$[R | t] = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \quad (7)$$

Pomocí rovnice č. 5 lze transformovat bod M_w ve světových souřadnicích do bodu $M_c = [x_c, y_c, z_c]$ v kamerových souřadnicích.

$$M_c = RM_w + t \quad (8)$$

Úpravou lze spočítat polohu hlavního bodu C ve světových souřadnicích. Musí platit $0 = RC_w + t$, potom $C_w = -R^{-1}t = -R^T t$. Za použití rovnic (5-8) je možné rovnici (4) vyjádřit v affinních souřadnicích tvarem:

$$z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = KR(M_w - t) = \begin{bmatrix} \alpha_u & s & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} \quad (9)$$

6.3 Zkreslení skutečného objektivu

Světlo vstupuje do kamery soustavou čoček, tj. objektivem. Oproti středovému modelu dírkové komory jsou skutečné objektivy nelineární a při projekci prostoru do obrazu způsobují zkreslení. To může být kompenzováno také nelineární transformací souřadnic. Na výslednou deformaci obrazu mají největší vliv radiální a tangenciální složka zkreslení.

Model definuje affinní transformaci souřadnic:

$$x' = x_c / z_c \quad (10)$$

$$y' = y_c / z_c \quad (11)$$

$$x'' = x'(1 + k_1 r^2 + k_2 r^4) + 2p_1 x' y' + p_2 (r^2 + 2x'^2) \quad (12)$$

$$y'' = y'(1 + k_1 r^2 + k_2 r^4) + p_1(r^2 + 2y'^2) + 2p_2 x'y' \quad (13)$$

kde:

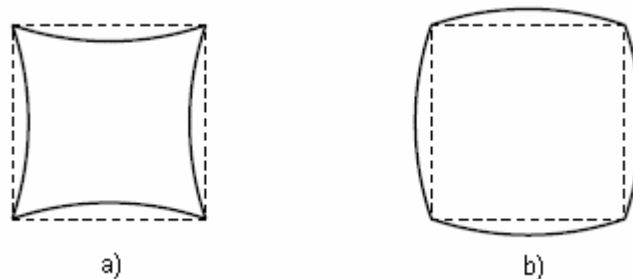
$$r^2 = x'^2 + y'^2 \quad (14)$$

potom je možné kompenzované souřadnice obrazového bodu vyjádřit:

$$u = \alpha_u x'' + u_0 \quad (15)$$

$$v = \alpha_v y'' + v_0 \quad (16)$$

Radiální zkreslení představují koeficienty k_1 , k_2 a tangenciální koeficienty p_1 , p_2 . Ostatní koeficienty zkreslení nejsou v OpenCV uvažovány. Stejně jako vnitřní parametry je zkreslení závislé pouze na vlastnostech objektivu. Záporné hodnoty koeficientů k_1 a k_2 se projevují jako poduškové zkreslení (angl. pincushion distortion), kladné naopak jako soudečkové zkreslení (angl. barrel distortion), viz Obr. 13. Koeficienty p_1 a p_2 se uplatní v případě, kdy není obrazová rovina kolmá na optickou osu.



Obr. 13.: Typy radiálního zkreslení a) poduškové, b) soudečkové.

Radiální zkreslení není uváděno výrobcí v technické dokumentaci kamer či objektivů. Pro účely měření je nezbytné zkreslení kompenzovat kalibrací kamery.

7 Kalibrace kamery

Kalibrace kamery je postup k určení číselných hodnot koeficientů matice kamery K , matice vnějších parametrů $[R|t]$ a koeficientů zkreslení. Díky znalosti těchto parametrů je možné rekonstruovat obraz a zbavit se nechtěného zkreslení. Kalibrace kamery je rozdělena na dvě základní úlohy [16].

7.1 Kalibrace ze známé scény

V případě, že je známa relativní poloha bodů M ve 3D světě, potom je jejich projekce do bodů m obrazové roviny jednoznačně určena rovnicí 4. Projektivní matice P lze rozepsat do tvaru:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (17)$$

Matice P je tvořena 12 neznámými m_{11} až m_{34} . Úpravou rovnice 14 dostaneme dvě lineární rovnice pro každý bod známé scény (rovnice 15).

$$\begin{bmatrix} x & y & z & 1 & 0 & 0 & 0 & 0 & -ux & -uy & -uz & -u \\ 0 & 0 & 0 & 0 & x & y & z & 1 & -vx & -vy & -vz & -v \\ & & & & & & & & \vdots & & & \vdots \\ & & & & & & & & & & & -m_{34} \end{bmatrix} \begin{bmatrix} m_{11} \\ m_{12} \\ \vdots \\ m_{34} \end{bmatrix} = 0 \quad (18)$$

Pro vyřešení této rovnice je tedy potřeba znát polohu šesti bodů ve 3D světě a šest odpovídajících obrazových bodů.

Z matice M je poté možné vyjádřit vnitřní a vnější parametry pomocí rovnic 19 až 21.

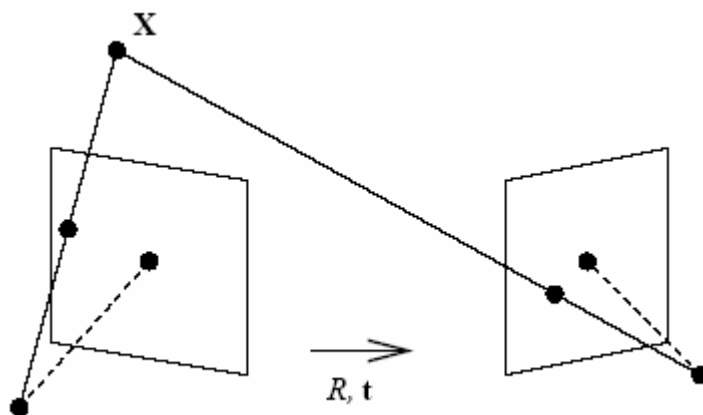
$$M = [KR | -KRt] = [A | b] \quad (19)$$

$$A = KR \quad (20)$$

$$t = -A^{-1}b \quad (21)$$

7.2 Kalibrace z neznámé scény

Tento způsob kalibrace se použije v případě, kdy nejsou známy parametry scény. K tomu je potřeba více pohledů na kalibrační obrazec a předpoklad neměnnosti vnitřních parametrů kamery, viz. Obr. 14. Tento způsob kalibrace pokrývá určení vnitřních a vnějších parametrů.



Obr. 14.: Kalibrace kamery z neznámé scény.

Kalibrovat kameru z neznámé scény lze dvěma způsoby:

1. Ze známého vzájemného pohybu mezi minimálně dvěma pohledy.
2. Z neznámého vzájemného pohybu. Tento způsob je také nazýván samokalibrace. Je potřeba minimálně tří pohledů na kalibrační obrazec.

Kalibrace je založena na snímání kalibračního obrazce o známých rozměrech. Ten může být tvořen pravidelnou maticí bodů, průsečíky mezi navzájem kolmými rovnoběžkami v jednom a druhém směru, nebo nejčastěji šachovnicí. Řešení úlohy kalibrace z neznámé scény je ovšem výrazně složitější [16].

8 Vývoj aplikace

Aplikace byla vyvíjena v OS linux, konkrétně v distribuci Fedora 5 pro správce oken KDE 3.5 za použití programového vybavení, které je jeho součástí. Jelikož je prostředí KDE dodáváno i v jiných distribucích a nebo je možné ho doinstalovat, je předpoklad, že videometrická aplikace bude spustitelná na většině klonů OS Linux

8.1 Pořízení obrazu

System Fedora 5 rozpozná správně zachytávací kartu jako zařízení pro zpracování televizního signálu po jejím nainstalování do PC. Při puštění počítače se načte modul jádra bttv, který funguje jako rozhraní systému pro komunikaci s kartou s čipem BT878 nebo BT848. Potom lze zařízení adresovat pomocí virtuálního souboru /dev/video0.

V C++ programu pak lze použít zařízení podobně, jako soubor. Příkazem `fd = open("/dev/video0", O_RDWR)` zařízení inicializujeme a do proměnné „fd“ typu `int` uložíme číslo, které operační systém použije pro adresaci tohoto souboru.

Teď už můžeme použít funkci `ioctl` a zavolat některou z funkcí rozhraní Video4Linux, abychom připravili zařízení na čtení dat. Kdykoli potom použijeme příkazy `V4L_VIDIOCMCAPTURE` a `VIDIOCSYNC`, dojde k překopírování dat ze zařízení do namapovaného místa paměti. Konkrétní příklad kódu najdeme v příloze č.: 2.

8.2 Volání OpenCV funkcí

Před voláním jednotlivých funkcí knihovny OpenCV, je nutné do zdrojového kódu vložit její hlavičkové soubory:

```
#include <cv.h>
#include <cvaux.h>
#include <highgui.h>
#include <cxcore.h> // není nutné - nachází se v cv.h
```

Dále je potřeba před kompilováním programu předat linkeru cesty k souborům OpenCV knihovny, například:

```
-I/usr/local/include -L/usr/local/lib -lopencv -lhighgui -
lstdc++
```

Pro lepší orientaci při používání funkcí OpenCV knihovny bylo výhodné si osvojit jmennou konvenci. Každá funkce nebo datový typ začíná písmeny cv a dále její název popisuje, co je možné od ní očekávat, viz příklad:

`cvAkceCílMód(...)`

Akce = co chceme udělat (např. set, create)
Cíl = s čím budeme pracovat (např. contour, polygon)
Mód = volitelně jakým způsobem (např. argument type)

např.: `cvCreateImage(...)`

Podobně je pro práci s datovými typy zavedeno značení:

`cvMat`

`CV_<bitová hloubka>(S|U|F)C<počet kanálů>`

S = Signed integer
U = Unsigned integer
F = Float

např.: `CV_8UC1` znamená 8-bitovou (0-255) jednokanálovou matici,
`CV_32FC2`

8.3 Kalibrace kamery

Pro kalibraci kamery je OpenCV knihovna vybavena několika funkcemi od jednoduchých až po komplexnější. Samotná knihovna je postavena tak, že využívá pro komplexnější funkce ty jednodušší.

Například funkce pro subpixelové nalezení rohu (`cvFindCornerSubPix`) je použita pro hledání bodů, kde se vzájemně dotýkají tmavé čtverce kalibračního obrazce (šachovnice). Tato funkce se jmenuje `cvFindChessboardCorners` a vrací souřadnice obrazových bodů dotyku tmavých čtverců v neceločíselných hodnotách. Tyto body je poté možné pomocí funkce `cvDrawChessboardCorners` vykreslit do původního obrazu a zkontrolovat jejich shodu.

V OpenCV knihovně se využívá neceločíselných souřadnic bodů velmi často. Je to dáno nejen tím, že aplikováním matematických výpočtů na celočíselné hodnoty původního obrazu dostáváme hodnoty neceločíselné, ale také subpixelovou přesností některých

funkcí. Po aplikování nejrůznějších funkcí tedy dostáváme nové obrazy s body s neceločíselnými souřadnicemi. Aby bylo možné tyto nové obrazy zobrazit, je nutné obraz přepočítat do celočíselných hodnot pomocí jasové interpolace. K tomu slouží funkce `cvGetQuadrangleSubPix` za použití bilineární interpolace.

Pro funkce spojené s kalibrací obrazu je použito modelu perspektivního zobrazení popsaného v kapitole 6. Z nalezených bodů šachovnice a jejich odpovídajících bodů ve 3D světě lze pomocí funkce `cvCalibrateCamera2` spočítat matici vnitřních parametrů. Matici vnějších parametrů lze poté spočítat pomocí funkce `cvFindExtrinsicCameraParams2`.

Pro kompenzaci zkreslení byla použita funkce `cvUndistort2`, která pomocí radiálních a tangenciálních koeficientů přepočítá obraz. Ve stejném duchu pracuje i funkce `cvInitUndistortMap`, která spočítá mapu (look-up tabulku). Mapu je potom možné aplikovat pomocí funkce `cvRemap` na další obrazy a zrychlit výpočet.

9 Závěr

Videometrická Aplikace je určena pro operační systém Linux a umožňuje tak snížit závislost na komerčních aplikacích, navíc je možné ji libovolně měnit a dále vyvíjet podle konkrétních potřeb.

Pořízení obrazu je navrženo tak, aby umožnilo použití spotřební videotechniky, a to zejména za použití zachytávací karty založené na čipu Bt878, jelikož je tento čip přítomen na většině dnes vyráběných televizních kartách. Aplikace umí pracovat i se soubory běžných obrazových formátů a na ně aplikovat nejrůznější metody obrazové analýzy vedoucí k nalezení hran či objektů.

Uživateli tak umožní si vše vyzkoušet a podle toho navrhnout samotné měření konkrétní úlohy. Aplikace je tedy určena jak pro výuku a vývoj, tak i pro měření s tím, že nelze očekávat velké přesnosti.

Aplikace podporuje kalibraci optické aparatury. Umí se přizpůsobit použitému typu objektivu, úhlu snímání. K tomu používá různé transformace perspektivy, soudečkového a poduškového zkreslení.

Teoretická část diplomové práce se zabývá vysvětlením pojmů jako Počítačové vidění a Videometrie a dále se snaží seznámit uživatele se základními pravidly při tvorbě videometrické úlohy. Samotný vývoj aplikace je v diplomové práci popisován hlavně ve způsobu práce s knihovnamí Video4Linux a OpenCV, co všechno umí a jak s nimi pracovat.

Seznam použité literatury

- [1] Hlaváč, V., Sedláček, M.: Zpracování signálů a obrazů, Skriptum ČVUT FEL, Praha, 2000.
- [2] Špatněl, M.; Beran, V.: Obrazové segmentační techniky, přehled existujících metod, studijní dokumentace VUT v Brně, 2006, nalezeno: 15.2.2006
<http://www.fit.vutbr.cz/~spanel/segmentace/#_Toc125769338>
- [3] Přednáškové materiály ČVUT v Praze zabývající se objektivy
<http://measure.feld.cvut.cz/groups/edu/osv/optika/Priklady_index.html>, nalezeno: 8.5. 2006.
- [4] Fischer, J.: Optoelektronické senzory a videometrie, Skriptum ČVUT FEL, Praha , 2002.
- [5] Havle Otto: Osvětlovače pro systémy strojového vidění, Automatizace (48), číslo 7-8, strana 456
- [6] Černý Tomáš: Návrh a realizace osvětlovače z vysoce svítivých LED diod, ročníkový projekt, TUL, 2004.
- [7] Internetová stránka firmy national Instruments, výrobce videometrické aplikace ImaqVision Builder AI
<<http://www.ni.com/vision/vbai.htm>>, nalezeno: 8. 11. 2005
- [8] Internetová stránka výrobce věnovaná zachytávací kartě AverMedia EZCapture
<http://www.avermedia.com/cgi-bin/products_capture_ezcapture.asp>, nalezeno:10.6. 2006.
- [9] Schimek M., H.: Video for Linux Two API Specification
<<http://v4l2spec.bytesex.org/v4l2spec/v4l2.pdf>>, nalezeno: 14. 12. 2005
- [10] Cox, A.: Video for Linux Programming
<<http://developer.osdl.org/dev/opendriversonline/documentation/docbook/docbook-2.6.16.7/html/videobook/>>, nalezeno: 25.1.2006.

- [11] Internetová stránka firmy Intel věnovaná projektu OpenCV
<<http://www.intel.com/technology/computing/opencv/index.htm>>,
nalezeno 8. 6. 2006.
- [12] Internetová stránka věnovaná projektu OpenCV, možnost stažení open source knihovny <<http://opencvlibrary.sourceforge.net/>>, nalezeno: 8. 11. 2005.
- [13] Internetová stránka věnovaná problematice sdílených knihoven v OS Linux
<<http://phoenix.inf.upol.cz/linux/prog/sdilene-lib.html>>, nalezeno: 3. 3. 2006.
- [14] Internetová stránka k aplikačnímu rozhraní grafického uživatelského rozhraní knihoven QT
<<http://www.trolltech.com/products/qt/index>>, nalezeno: 25.1.2006.
- [15] Uživatelský manuál vývojového prostředí KDevelop
<<http://docs.kde.org/development/en/kdevelop/kdevelop/>>, 25.1. 2006
- [16] Šonka, M., Hlaváč, V., Boyle, R.D.: Image Processing, Analyses and Machine Vision. PWS, Boston, USA, druhé vydání, 1998.
- [17] Internetová stránka zabývající se problematikou kalibrace kamer.
<<http://homepages.inf.ed.ac.uk/cgi/rbf/CVONLINE/entries.pl?TAG252>>,
nalezeno: 4. 7. 2006
- [18] Lepetit, V.; Fua, P.: Monocular Model-Based 3D Tracking of Rigid Object: A Survey, Computer Graphic and Vision, Vol.1, No 1(p.1-89), 2005

Seznam obrázků

Obr. 1.: Měřicí sestava	15
Obr. 2.: Princip perspektivního zobrazení.	18
Obr. 3.: Hloubka ostrosti a hloubka pole	19
Obr. 4.: Telecentrický objektiv	20
Obr. 5.: Hypercentrický objektiv	21
Obr. 6.: Používané způsoby osvětlení.....	28
Obr. 7.: Pracovní plocha programu NI Imaq Vision Builder AI.	32
Obr. 8.: Přehled zapojení konektorů a jejich konverze.....	35
Obr. 9.: Způsob skládání obrazu při prokládaném řádkování.....	35
Obr. 10.: Průběh televizního signálu.....	36
Obr. 11.: Karta AverMedia EZCapture s čipem BT878	36
Obr. 12.: Model perspektivního zobrazení, kde $(O, \vec{i}, \vec{j}, \vec{k})$ je světový systém souřadnic, $(C, \vec{i}_c, \vec{j}_c, \vec{k}_c)$ je kamerový systém souřadnic, M je bod ve 3D světě a m jeho projekce do obrazové roviny.....	47
Obr. 13.: Typy radiálního zkreslení a) poduškové, b) soudečkové.	50
Obr. 14.: Kalibrace kamery z neznámé scény.....	52

Seznam tabulek

Tab. 1.: Přehled parametrů používaných televizních norem.....	34
Tab. 2.: Přiřazení souborů k V4L zařízením.....	38

Seznam příloh

Příloha č. 1: IPL datová struktura obrazu

Příloha č. 2: Příklad kódu komunikace se zachytávací kartou

Příloha č. 3: CD

Příloha č. 1: IPL datová struktura obrazu

```
IplImage
|-- int  nChannels;      // Number of color channels
(1,2,3,4)
|-- int  depth;        // Pixel depth in bits:
|                                     //   IPL_DEPTH_8U,
IPL_DEPTH_8S,
|                                     //
IPL_DEPTH_16U,IPL_DEPTH_16S,
|                                     //
IPL_DEPTH_32S,IPL_DEPTH_32F,
|                                     //   IPL_DEPTH_64F
|-- int  width;        // image width in pixels
|-- int  height;       // image height in pixels
|-- char* imageData;   // pointer to aligned image
data
|                                     // Note that color images
are stored in BGR order
|-- int  dataOrder;    // 0 - interleaved color
channels,
|                                     // 1 - separate color
channels
|                                     // cvCreateImage can only
create interleaved images
|-- int  origin;       // 0 - top-left origin,
|                                     // 1 - bottom-left origin
(Windows bitmaps style)
|-- int  widthStep;    // size of aligned image row
in bytes
|-- int  imageSize;    // image data size in bytes
= height*widthStep
|-- struct _IplROI *roi;// image ROI. when not NULL
specifies image
|                                     // region  to be processed.
|-- char *imageDataOrigin; // pointer to the
unaligned origin of image data
|                                     // (needed for correct
image deallocation)
|
|-- int  align;        // Alignment of image rows:
4 or 8 byte alignment
|                                     // OpenCV ignores this and
uses widthStep instead
|-- char colorModel[4]; // Color model - ignored by
OpenCV
```

Příloha č. 2: Příklad kódu komunikace se zachytávací kartou

```
/* ***** */
/* Popis obsahu souboru: */
/* Komunikace s grabberem */
/* ***** */

#include "grab.h"

using namespace std;

int signaled=0;

GRAB::GRAB(int width, int height,
           int input, // kompozitní nebo komponentní vstup
           int norm, // PAL nebo NTSC
           int pal, // VIDEO_PALETTE_RGB24, VIDEO_PALETTE_GRAY,
           const char * VIDEO_DEV // /dev/video
           )
{
    map_=NULL;
    memset(device, 0, 100);
    memcpy(device, VIDEO_DEV, strlen(VIDEO_DEV));
    fd=-1;
    mmapflags = PROT_READ|PROT_WRITE;

    // otevření zařízení
    if (-1 == fd && -1 == (fd = open(device,O_RDWR)))
    {
        cout << "nemohu otevrit zarizeni" << device << endl;
        exit(1);
    }

    // zjištění parametrů
    if (-1 == ioctl(fd,VIDIOCGCAP,&capability))
    {
        perror("ioctl VIDIOCGCAP");
        exit(1);
    }

    cout << "grab.cc: GRAB(): Capabilities: " << endl;
    cout << "name\t\t" << capability.name << endl
    << "type\t\t" << capability.type << endl
    << "width\t\t" << capability.maxwidth << endl
    << "height\t\t" << capability.maxheight << endl
    << endl;

    // nastavení parametrů
```

```

channel.norm = norm;
channel.channel = input;
cout << "nastavuji normu = " << channel.norm << endl
<< "          kanal = " << channel.channel << endl
<< endl;

if (-1 == ioctl(fd,VIDIOCSCHAN,&channel))
    perror("ioctl VIDIOCSCHAN");

// namapování paměti
if (-1 == ioctl(fd,VIDIOCGMBUF,&gb_buffers))
    perror("ioctl VIDIOCGMBUF");

map_ = (unsigned char*) mmap(0,gb_buffers.size,
mmapflags,MAP_SHARED,fd,0);
if ((unsigned char*)-1 == map_)
    perror("mmap");

// příprava na grabování
// VIDEO_PALETTE defined in videodev.h
gb1.format = pal; // Typ barevné palety;
gb1.frame = 0;
gb1.width = width;
gb1.height = height;

cout << "grabuji " << width << "x" << height << "
images" << endl;
}

GRAB::~GRAB()
{
    signaled=1;
    close(fd);
}

unsigned char * GRAB::grab()
{
    if(map_ == NULL)
    {
        cout << "video zarizeni není pripraveno.\n";
        exit(-1);
    }

    // Grab frame
    if(!signaled)
    {
        ioctl(fd,VIDIOCMCAPTURE,&gb1);
    }
}

```

```
        if (-1 == ioctl(fd,VIDIOCSYNC, &gb1.frame))
        {
            perror("ioctl VIDIOCSYNC");
            exit(1);
        }
    }
    return map_;
}
```


Příloha č. 3: **Obsah přiloženého CD**

- Závěrečná zpráva
- Program „Videometrická aplikace“
- Zdrojový kód programu
- Knihovna OpenCV
- Ovladač zachytávací karty pro OS Windows
- Offline verze vybrané literatury v elektronické podobě