



**ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE**  
**Fakulta elektrotechnická**  
**Katedra radioelektroniky**

## **Inteligentní kamerový systém**

**květen 2015**

**Diplomant: Bc. Martin Šimák**  
**Vedoucí práce: Ing. Stanislav Vitek, PhD**

## Čestné prohlášení

Prohlašuji, že jsem zadanou diplomovou práci zpracoval sám s přispěním vedoucího práce a konzultanta a používal jsem pouze literaturu v práci uvedenou. Dále prohlašuji, že nemám námitek proti půjčování nebo zveřejňování mé diplomové práce nebo její části se souhlasem katedry.

Datum:

.....  
podpis diplomanta

## ZADÁNÍ DIPLOMOVÉ PRÁCE

Student: **Bc. Š I M Á K Martin**

Studijní program: Komunikace, multimédia a elektronika  
Obor: Elektronika

Název tématu: **Inteligentní kamerový systém**

### ***Pokyny pro vypracování:***

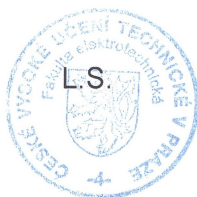
1. Navrhněte inteligentní multi kamerový bezpečnostní systém
2. Systém umožní sledování vytyčeného prostoru, uživatelskou definici perimetru, kritických zón
3. Vstup osoby do definované zóny je možné spojit s akcí (záznam v databázi, e-mail, ...)
4. Systém umožní i pokročilé funkce, sledování a jednoduchou identifikaci osoby nebo počítání průchoďů
5. Systém realizujte pomocí PTZ kamer se síťovým rozhraním

### ***Seznam odborné literatury:***

- [1] Aghajan H., Cavallaro A., Multi-Camera Networks: Principles and Applications, Academic Press, ISBN 978-0123746337
- [2] Song D., Sharing a Vision, Springer, 2009, ISBN 978-3-540-88064-6

Vedoucí: **Ing. Stanislav Vítek, Ph.D.**

Platnost zadání: 31. 8. 2016



prof. Ing. Miroslav Husák, CSc.  
vedoucí katedry

prof. Ing. Pavel Ripka, CSc.  
děkan

V Praze dne 2. 2. 2015

## **Poděkování**

Na tomto místě bych rád poděkoval vedoucímu své diplomové práce Ing. Stanislavu Vítкови, PhD za odborné vedení a za pomoc při návrhu aplikace pro kamerový systém. Dále bych rád poděkoval své rodině za to, že mi umožnila studovat a přítelkyni za podporu při studiu.

## **Anotace:**

Diplomová práce shrnuje poznatky o kamerových systémech a využití knihovny OpenCV pro účely aplikace inteligentního kamerového systému v prostředí Microsoft Visual Studio 2013. Výsledný program je utvořen tak, aby bylo možné na danou práci navázat a pokračovat jejím případným použitím v praxi.

## **Klíčová slova:**

Kamerový systém, kamera, OpenCV, Visual Studio, sledování, detekce

## **Summary:**

The Master's Thesis summarizes the findings about camera systems and the use of the library OpenCV for purpose of the application of the intelligent camera system in Microsoft Visual Studio 2013. The final program is created to make possible to continue in this Thesis and to use it in practice.

## **Index Terms:**

Camera system, Camera, OpenCV, Visual Studio, tracking, detection

# Obsah:

<b>1. Úvod:</b> .....	<b>1</b>
<b>2. Kamerové systémy</b> .....	<b>2</b>
2.1 Analogové kamerové systémy .....	3
2.2 Digitální kamerové systémy .....	4
2.2.1 IP kamery.....	4
2.2.2 Záznamová zařízení .....	6
2.2.3 Zobrazovací zařízení .....	7
2.2.4 Doplnky ke kamerovým systémům .....	9
<b>3. OpenCV</b> .....	<b>10</b>
3.1 Jednotlivé moduly OpenCV .....	10
3.1.1 CxCore .....	10
3.1.2 CV .....	11
3.1.3 CxAux.....	11
3.1.4 MLL .....	12
3.1.5 HighGui .....	12
<b>4. Návrh kamerového systému</b> .....	<b>14</b>
4.1 Technické parametry kamery PT8133W .....	14
4.2 Ovládání kamery pomocí PC .....	15
4.3 Ovládání kamery pomocí URL příkazů .....	18
4.4 Ukázka nejpoužívanějších příkazů.....	19
<b>5. Návrh aplikace pro kamerový systém</b> .....	<b>20</b>
5.1 Hlavní menu .....	20
5.1.1 Zobrazovací okna.....	20
5.1.2 Box s jednotlivými funkcemi .....	21
5.1.3 Box pro vyhledání IP adres .....	28
5.2 Algoritmus pro sledování osob.....	29
5.3 Algoritmy pro klasifikaci osob .....	33
5.4 Blokový diagram .....	35
5.5 Použití aplikace pro sledování pohybu v budově .....	36
<b>6. Závěr</b> .....	<b>39</b>

<b>Zdroje:</b> .....	<b>40</b>
<b>Seznam obrázků:</b> .....	<b>41</b>
<b>Seznam zkratk:</b> .....	<b>42</b>
<b>Seznam příloh:</b> .....	<b>43</b>



# 1. Úvod:

Vzrůstající výpočetní výkon osobních počítačů umožňuje detailnější a rychlejší zpracování kamerového obrazu. Z toho důvodu jsou na vzestupu knihovny počítačového vidění, mezi něž patří i knihovna OpenCV. Intel, který je původním autorem, vytvořil účinný nástroj pro pokročilou práci s obrazem. Použití této knihovny můžeme nalézt v robotice, automobilovém průmyslu, ale i kamerových systémech, kterými se budeme zabývat v této práci.

Ve druhé kapitole se zaměříme na kamerové systémy a jejich rozdělení. Jsou zde popsány jak digitální, tak i analogové systémy s důrazem na podrobnější popis digitálních systémů, které budeme používat v praktické části. Součástí jsou informace o jednotlivých zařízeních tvořící takový systém.

Třetí část je zaměřena na seznámení s knihovnou OpenCV, kterou budeme potřebovat pro propojení jednotlivých kamer s grafickým prostředím Visual Studia a následně využijeme vestavěné funkce při práci s kamerovým obrazem.

Jelikož před samotným programováním aplikace, je nutné se nejdříve seznámit s použitými kamerami VIVOTEK PT8133W, je třetí kapitola věnována jejich popisu. Nalezneme zde jednotlivé vlastnosti, popis programu dodávaný společně s kamerou a podrobný návod, jak ovládat kameru pomocí URL příkazů. Součástí této kapitoly je i výběr vhodného propojení kamery s počítačem.

Poslední kapitola je věnována samotné aplikaci inteligentního kamerového systému. Nalezneme zde popis grafického prostředí a jednotlivých bloků programu. To zahrnuje jednak rozbor algoritmů používaných pro sledování osob, ale i popis jak daný program používat.

Cílem této práce je propojit vývojové prostředí Visual Studio 2013 s knihovnou OpenCV a následně vytvořit aplikaci pro práci s kamerami VIVOTEK PT8133W. Výsledný program by měl být schopný snímat až čtyři kamery současně a společně s grafickým prostředím nabídnout jednoduché sledování pohybujících se objektů či osob.

Takto vytvořená aplikace představuje podpůrné vývojové prostředí pro inteligentní kamerové systémy. Budoucí programátor bude mít připravený program se zajištěnou komunikací s kamerami, ovládním, či nástrojem pro tvoření perimetrů. Po menších úpravách sledovací funkce je možné tímto nástrojem chránit majetek, osoby či sledovat dopravní situaci na silnicích. Díky použití vývojového prostředí Visual Studio, je možné v budoucnu vytvořit také mobilní aplikaci, a tak rozšířit možnosti použití daného systému.

## 2. Kamerové systémy

Kamerový systém představuje spojení kamery spolu s několika dalšími zařízeními, díky kterým můžeme efektivně využít plný potenciál kamery a výpočetní techniky, jenž nás všude obklopuje. Od roku 1941, kdy se objevila první průmyslově vyráběná kamera, uběhl zhruba rok a světlo světa spatřil první kamerový systém vytvořený firmou Siemens AG. Jednalo se o techniku, která zprostředkovávala sledování odpalování raket V-2, za jejíž vývoj byl zodpovědný německý inženýr Walter Bruch. Od té doby se začaly kamerové systémy uplatňovat ve stále širší oblasti lidské činnosti. Zpočátku kamerové systémy sloužily spíše pro soukromé účely, jako například ochrana obchodů, bank, či pro výzkum. Jak se postupem času tyto technologie stávaly dostupnějšími, tak i oblast jejich využití se rozšiřovala. (1)

Počáteční nedůvěru lidí k této novince nebylo jednoduché překlenout. Teprve až v polovině devadesátých let nastal ve Velké Británii kamerový rozmach, kdy do těchto systémů bylo investováno zhruba tři čtvrtiny rozpočtu na boj proti zločinu. Významnou měrou k rozvoji v tomto odvětví přispěly také teroristické útoky, jako například ten, který se stal 11. září v USA. (2)

Většinu kamerových systémů tvoří trojice prvků. Každý kamerový systém je vybaven vstupním zařízením, a to kamerou. Účelem tohoto prvku je zprostředkovat konverzi obrazu mezi snímanou scénou a systémem, který následuje za ní. To se děje pomocí převodu světelného paprsku na elektrický signál v kamerovém snímači. Druhým podstatným prvkem kamerového systému je přenosové vedení, díky kterému můžeme informace z kamery poslat do jiné části kamerového systému a zde ho zpracovat. Poslední částí, která se v dnešní době vyskytuje téměř u všech kamerových systémů, je záznamové zařízení, díky němuž je možné uchovat obrazová data pro pozdější zpracování či analýzu.

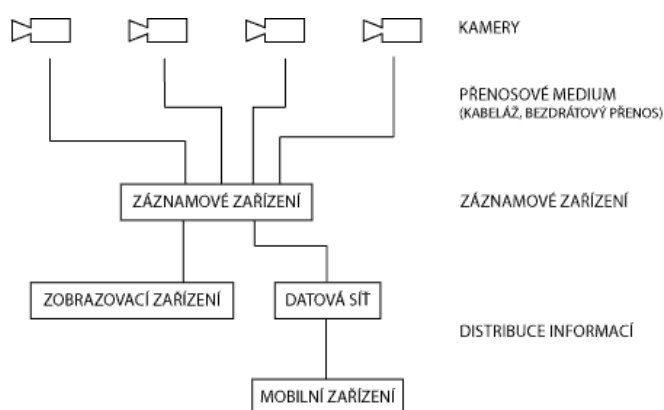


Obr. 1 Ústředna bezpečnostního kamerového systému (3)

V průběhu vývoje nových snímačů pro kamery a všudypřítomné digitalizace se kamerové systémy rozdělily do dvou skupin, které si následně popíšeme.

## 2.1 Analogové kamerové systémy

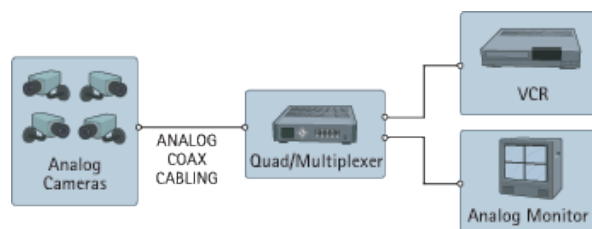
CCTV (Close Circuit Television) neboli uzavřené televizní okruhy je obecné označení pro kamerové systémy. Jak již bylo zmíněno, analogové kamerové systémy jsou používány od roku 1942 a jejich využití přetrvává do dnešních dní. Jedním z důvodů, proč tyto systémy zcela nenahradily modernější digitální systémy je, že firmy nechtějí investovat velké peníze do nákupu kompletní technologie. Místo toho investují nižší částky na obnovení pouze menších součástí tak, aby zajistily kvalitu potřebnou k dalšímu provozu. Na následujícím obrázku vidíme zapojení obecného kamerového systému, který je rozdělen do 4 částí.



Obr. 2 Schéma obecného kamerového systému (4)

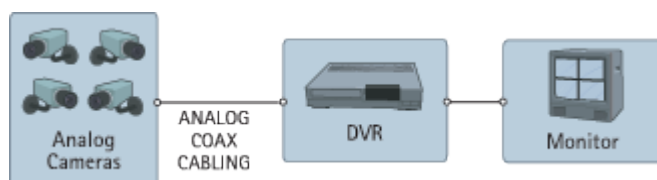
Na prvním místě jsou zde snímací kamery, které představují nejdůležitější část celé soustavy. Při použití levnějších kamer musíme počítat s nízkou kvalitou výstupního obrazu. Pokud mluvíme o analogových kamerách, myslíme tím takové zařízení, jehož výstupní signál je analogový, v takovém případě již neřešíme vnitřní obvody, které mohou obsahovat digitální zpracování. Z důvodu přenosu analogového signálu do ostatních částí je dalším prvkem kamerového systému přenosová linka. Ta je tvořena koaxiálním kabelem s odporem  $75 \Omega$ . Nevýhodou analogových kamer je nutnost použití druhého kabelu, který zajišťuje napájení.

Třetím prvkem v pořadí za přenosovým médiem je buď analogový multiplexer s kazetovým záznamovým zařízením, nebo digitální záznamové zařízení. U plně analogových systémů je možné v určitých případech použít před záznamovým zařízením multiplexer. Ten zajistí to, že na jednu videokazetu můžeme nahrávat několik datových proudů současně za cenu menšího rozlišení. Příklad takového zapojení vidíme na následujícím obrázku. (4) (5)



Obr. 3 Schéma analogového kamerového systému (5)

Pokud bychom chtěli využít výhod částečně digitálního systému, u kterých výstupy z kamer jsou zapojené do digitálního záznamového zařízení, tak již není nutné používat multiplexer. Ten je obsažen v záznamovém zařízení, které nám díky digitalizaci obrazu nabízí jednak nahrávání několika kanálů v plném rozlišení, ale také absenci videokazety, která je nahrazena pevným diskem. Blokové schéma zmíněného zapojení je ukázáno na následujícím obrázku.



Obr. 4 Schéma částečně analogového kamerového systému (5)

Posledním prvkem, který se vyskytuje u většiny kamerových systémů, je zařízení pro zobrazení kamerových výstupů či informací o aktuálním stavu. V případě analogových systémů se používají klasické CRT (Cathode Ray Tube) monitory, na kterých obsluha daného systému může sledovat jeden výstup ve větším rozlišení či několik výstupů v rozlišení menším. U částečně analogových systémů, jež byly zmíněny výše, se převážně používají LCD (Liquid Crystal Display) displeje, které vynikají poměrem cena ku výkonu. (4) (5)

## 2.2 Digitální kamerové systémy

Jelikož v praktické části budeme využívat výhradně digitální kamerové systémy, budou tyto systémy probrány do větší hloubky. V případě digitálních kamerových systémů téměř všechna zařízení využívají moderní digitální technologie. Jejich úkolem je digitalizace obrazu, zajištění komprese dat do požadovaného formátu a přenos dat na požadovanou vzdálenost s daným maximálním zpožděním a v potřebné kvalitě tak, aby mohla být následně uložena na vhodném záznamovém médiu. V následující části si podrobně popíšeme jednotlivá zařízení digitálních kamerových systémů.

### 2.2.1 IP kamery

Síťové IP kamery jsou záznamová zařízení, která slouží k zprostředkování obrazových dat dalším částem kamerového systému. Kromě obrazu mohou kamery v určitých případech poskytnout i nahrávání zvuku pomocí vestavěného mikrofону, a tak rozšířit pole využití pro dané zařízení. Na rozdíl od analogových kamer je přenos a řízení v daném případě zajištěn pomocí IP sítě. Ta nám poskytuje přenos dat s takovým zabezpečením, které se u analogových systémů nevyskytuje. Oproti analogovým kamerám připojených pomocí koaxiálního kabelu tato zařízení využívají zabezpečený přenos dat pomocí šifrování. Díky tomu není vůbec jednoduché začít odposlouchávat či manipulovat s datovým kanálem, a tak snadno ovlivnit či odstavit celý systém. (6) (7)

Velkou výhodou je možnost použití stávající počítačové sítě, která je schopna zprostředkovávat i jiné služby (internet, poštovní server). Díky tomu je velice jednoduché zakomponovat kamerové systémy do nových staveb, kde jsou tyto sítě již zhotoveny. V ideálním případě je vhodné pro kamerové systémy využít vlastní síť oddělenou od okolí, čímž zajistíme nejen větší přenosovou rychlost, ale i lepší zabezpečení. Každá kamera v takové síti má přidělenou svoji vlastní adresu a obsahuje vestavěný webový server. Na rozdíl od webkamery je možné využít dané zařízení i bez počítače.

U většiny IP kamer se můžeme setkat s mnoha zabudovanými funkcemi, které ulehčují správu systémů a rozšiřují pole jejich využití. Obsahují webové rozhraní, usnadňující práci s jednotlivými kamerami, které nám pomáhá jednoduše měnit většinu nastavení či automaticky ovládat pohyb kamery. Mezi funkcemi určené pro střežení určité oblasti zde můžeme nalézt uživatelskou definici perimetru, automatické sledování pohybu či oznámení narušení daného perimetru spuštěním alarmu. Nechybí zde ani záznam videa nebo fotografie narušitele. Dále mohou být vybaveny digitálními vstupy či výstupy, které se využívají pro připojení externích alarmů, tlačítek, indikačních diod nebo senzorů.



**Obr. 5 Příklad IP kamery s ochranným krytem (8)**

Na výše uvedeném obrázku můžeme vidět příklad běžné IP kamery využívané pro monitorování budov či venkovních prostor. V případě venkovních prostor nastávají situace, kdy je nutné kameru chránit před vlivy nepříznivého počasí, jako je například sněžení, déšť či vítr. Proti slabému dešti je možné kameru chránit speciálním štítem a v situacích, kdy by mohlo do zařízení vniknout větší množství vody, je lepší koupit kvalitní voděodolnou kameru. Nejen proti tepelným vlivům počasí, ale i k ochlazení kamery zahřáté z důvodu velké spotřeby vnitřních obvodů jsou některé kamery vybaveny systémem zvyšující či snižující teplotu v okolí kamery.

Kromě klasických fixních zařízení, které je možné manuálně otočit pouze jedním směrem, zde máme i nepřeberné množství PTZ kamer. PTZ je tvořeno anglickými označeními pro Pan (horizontální otáčení doleva či doprava), Tilt (vertikální náklon nahoru či dolů) a Zoom (přiblížení či oddálení). Vertikálním či horizontálním pohybem kamery můžeme rozšířit oblast pokrytí, kterou je kamera schopna sledovat, a tím snížit celkový počet kamer. (6) (7)

## 2.2.2 Záznamová zařízení

V případě digitálních kamerových systémů je všechen zaznamenaný obraz nutné uložit v digitální podobě. Klasické analogové VCR (Video Cassette Recording) zařízení již není možné pro daný typ dat použít. Pro záznam v potřebné kvalitě a velikosti je nutné zakoupit NVR (Network Video Recorder), jehož příklad můžeme vidět na následujícím obrázku.



Obr. 6 NVR záznamové zařízení (9)

Jedná se o zařízení, které nabízí kromě ukládání záznamu i mnoho dalších užitečných funkcí. Jedním ze způsobů, jak připojit k danému zařízení kameru, je využití síťového konektoru. Ten je ve většině případů propojen s hlavním směrovačem zajišťujícím komunikaci s ostatními zařízeními. Největší výhodou představuje možnost ukládání na pevný disk namísto klasických páskových kazet, které se používají u analogových systémů. Obraz je postupně ukládán na záznamové médium a při dosažení maximální kapacity jsou postupně mazána poslední uložená data a nahrazena novými. Vzhledem k relativně nízké ceně pevných disků je možné uchovávat data nejen po dobu měsíců, ale i několika let, než dojde k jejich smazání. Tak vysoké doby je dosaženo jednak velkou kapacitou úložného prostoru, jednak formátem ukládaných dat, počtem snímků za vteřinu či inteligentním ukládáním obrazu pouze při zaznamenaném pohybu. Pro potřeby delšího záznamu, než umožňuje vnitřní paměť, se používají externí zařízení, která se připojují pomocí USB 2.0 či FireWire.

Za zmínku stojí, že v dnešní době se ještě stále můžeme setkat s digitálními systémy obsahující DVR (Digital Video Recorder). Jedná se o hybridní záznamová zařízení umožňující nejen způsob záznamu zmíněný v minulém odstavci, ale díky BNC konektorům je zařízení schopné nahrávat obraz z analogových kamer.



Obr. 7 DVR záznamové zařízení (10)

Podobně jako kamery, také záznamová zařízení nabízejí kromě zpracování obrazu zvukový záznam pomocí audio vstupů. Dále zde můžeme nalézt pole vstupních konektorů pro připojení alarmů, tlačítek či senzorů což nám umožňuje vytvářet jednodušší systémy s minimem propojovacích prvků. (4) (7)



## 2.2.3 Zobrazovací zařízení

Další velmi důležitou částí digitálních systémů je zobrazovací zařízení. Díky němu může obsluha sledovat dění v jednotlivých částech střeženého objektu. V případě dražších systémů se můžeme setkat s rozsáhlými monitorovacími pulty, které disponují velkým množstvím zobrazovacích zařízení. Výhodou takových systémů je zobrazení jednotlivých kanálů v plném rozlišení. V převážné většině se však můžeme setkat s jednoduchými monitory, které sice poskytují zobrazení několika kamerových výstupů, avšak pouze v podstatně menším rozlišení.

Nyní si jednotlivé displeje rozdělíme podle využívané technologie.

### LCD monitory

Na začátku tohoto století došlo k velké expanzi LCD (Light Emitting Diode) displejů do všech odvětví průmyslu a osobního použití. Princip této technologie tkví v použití tekutých krystalů. Je vytvořena struktura z dvou polarizačních filtrů, mezi nimiž se nachází tekuté krystaly. Průchodem fotonů pocházejících ze zdroje světla umístěného na jedné straně struktury dochází k ovlivnění jejich polarizace. Následný polarizátor pak funguje jako filtr propouštějící pouze fotony, jejichž polarizace byla díky krystalové struktuře pootočená o 90°

Díky této technologii jsou LCD displeje velmi levné a dostupné v každém obchodě. Od těchto zařízení můžeme očekávat velmi nízkou spotřebu elektrické energie a dlouhou životnost. Bohužel u LCD displejů se setkáme pouze s nižšími pozorovacími úhly. Na následujícím obrázku můžeme vidět příklad běžného LCD monitoru. (8)



Obr. 8 LCD monitor (11)

### Plazmové monitory

V roce 1983, tedy zhruba 10 let od výroby prvního LCD displeje, spatřila světlo světa nová technologie založená na pokrocích v oborech zabývajících se plazmatem. Princip plazmových displejů tkví v použití fosforem pokrytých buněk tvořících jednotlivé pixely. Každá z buněk je naplněna vzácnými plyny, které po přivedení napětí na buňky změní skupenství a výsledkem je vznik plazmatu uvnitř buňky. Tato reakce způsobí uvolnění fotonů v podobě ultrafialového záření. K převodu z ultrafialové oblasti na viditelné světlo slouží speciální sloučenina fosforu, a to luminofor, který je na povrchu jednotlivých buněk.

Jejich výhodou je stejně jako u LCD displejů úzký design a úspora prostoru potřebného k umístění zařízení do místnosti. Vyrábějí se především ve větších úhlopříčkách a poskytují velké rozpětí pozorovacích úhlů. Nabízejí barevnou hloubku až 16,7 milionů barev což je více, než dokáže lidské oko rozpoznat. I přes velkou reklamu nejsou tak populární jako LCD displeje. To je dáno především menší životností, vyšší cenou a vyšší spotřebou elektrické energie. (9)

### OLED monitory



Obr. 9 OLED displej (13)

Poslední druh zobrazovacích zařízení, který si zmíníme, jsou OLED (Organic Light Emitting Diode) displeje. V roce 1987 byla firmou Eastman Kodak vyvinutá technologie displejů na bázi luminiscence organických materiálů. Princip spočíval v uložení několika tenkých vrstev organického materiálu mezi průhlednou anodu a kovovou katodu. Po přivedení napětí na tyto 2 desky jsou uvnitř vrstvy vyvolány kladné a záporné náboje, při jejichž spojení dojde k vyzařování dané vrstvy. Díky tomu tyto displeje nepotřebují vlastní podsvícení.

Tyto displeje opět vynikají velmi tenkým designem a přináší celou řadu výhod. Patří mezi ně vysoký kontrast, věrnější podání barev a velmi nízký příkon. Zmínit bychom mohli také až o 30% vyšší účinnost oproti klasickým LCD displejům. Díky použití organických materiálů, které vykazují luminiscenční vlastnosti, jsou navíc tyto materiály ještě velmi pružné, a tak je můžeme využít například jako rolovací displeje či zakřivené televizory. Momentálně k jejich největším nevýhodám patří jejich cena, ale v budoucnu se očekává její podstatné snížení poté, co se zvládne doladit potřebnou technologii výroby. Tyto displeje se zatím využívají převážně u mobilních telefonů, ale díky potenciálu, který mají do budoucna, lze očekávat, že jednou nahradí dosud nejvíce používané LCD monitory. (10)



## 2.2.4 Doplnky ke kamerovým systémům

### Kryty kamer

Kamerové kryty můžeme využít nejen k ochraně kamerových zařízení, ale i k tepelné stabilizaci. Té je docíleno pomocí vnitřního topného tělíska či ventilačního systému.



Obr. 10 Ochranný kryt pro kamery (15)

### Ovládací zařízení

Ovládací jednotky slouží k jednoduchému ovládnání celého kamerového systému využívající PTZ kamery. Většinou se skládají z jednoduché klávesnice pro volbu kamerového kanálu a z joysticku, který slouží k ovládnání horizontální nebo vertikální polohy kamery a také přiblížení. Součástí může být většinou 3 řádkový displej pro zobrazení informací o systému.



Obr. 11 Ovládací panel pro kamerové systémy (16)

### Poplachová tlačítka

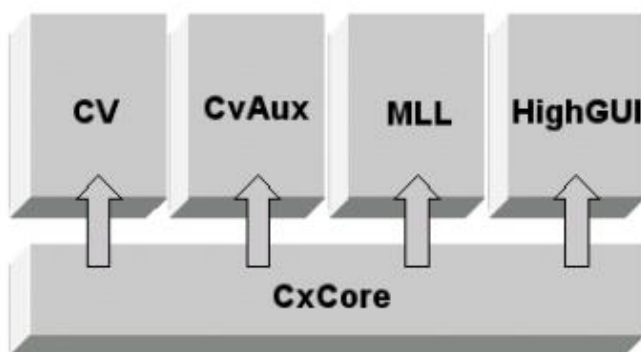
Jelikož kamerové systémy jsou připojeny k síti, je výhodné do takového systému zahrnout i poplachové tlačítko napojené na systém nejbližší policejní stanice. To může využít například zaměstnanec v obchodu pro rychlé a nenápadné přivolání pomoci.



Obr. 12 Poplachové tlačítko (17)

## 3. OpenCV

OpenCV je multiplatformní knihovna, která slouží pro potřeby počítačového vidění. Její historie sahá až do roku 1999, kdy byla vyvíjena společností Intel v jazyce C. Od té doby OpenCV nabývá na popularitě a jeho implementace nalezneme v mnoha jazycích. Tato knihovna se skládá z přibližně 400 funkcí, které jsou využívány v mnoha směrech počítačového vidění. Ať už se jedná o bezpečnost, robotiku či detekci v oblasti medicíny, ve všech těchto a mnoha dalších oblastech OpenCV nabízí velké možnosti využití. Tato knihovna je open source, což znamená, že ji může kdokoliv zcela zdarma využít ve svém projektu. Knihovna je rozdělena do 5 velkých bloků. Každý z těchto modulů poskytuje mnoho funkcí, které budou využity ve vytvářené aplikaci.



Obr. 13 Uspořádání knihovny OpenCV (18)

### 3.1 Jednotlivé moduly OpenCV

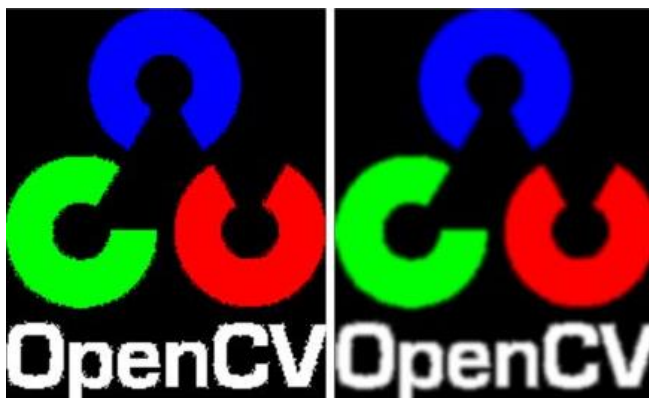
#### 3.1.1 CxCore

CxCore je nejdůležitějším modulem celé knihovny OpenCV, bez něhož nelze žádný další spustit. Nachází se v něm většina základních datových typů a nepoužívanějších funkcí. Jelikož důležitým typem v OpenCV jsou matice, tak zde nalezneme i funkce pro maticovou algebru či jednotlivé transformace.

Datový typ Mat definovaný výše zmíněným modulem představuje paměťové pole, které budeme využívat pro ukládání obrazových dat. OpenCV při nahrávání obrazu určí, zda se bude jednat o černobílý či barevný obraz, čímž můžeme ušetřit spoustu paměťového místa. Tento typ obsahuje také informace o formátu, v jakém jsou vstupní data ukládána. V průběhu programování je důležité nemíchat matice pro černobílý a barevný obraz, což by ve výsledku mohlo mít za následek špatné zobrazení požadovaných dat. Pro práci s tímto datovým typem zde nalezneme funkce pro inicializaci pole, změnu velikosti, přístupu k jednotlivým pixelům, statistickým výpočtům či složitějším diskretním transformacím. (18)

### 3.1.2 CV

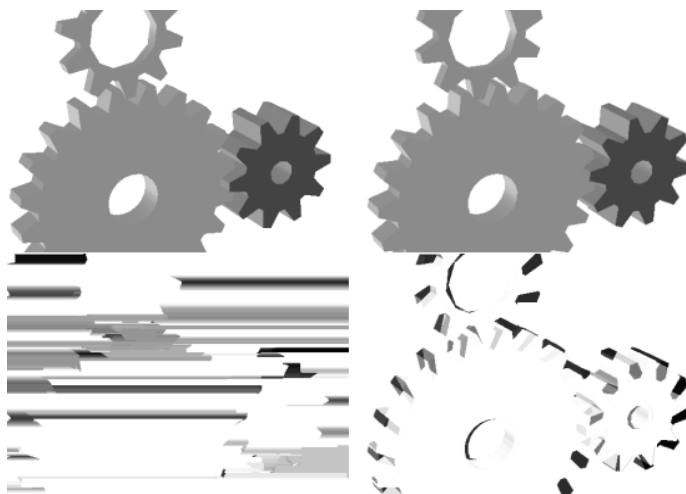
Další v pořadí je modul, který implementuje funkce používané v počítačovém vidění. Díky nim je možné pracovat s obrazem na vyšší úrovni. Jedná se především o morfologické operace, použití filtrů či hranových detektorů. Na následujícím obrázků můžeme vidět použití funkce `GaussianBlur()`, která se používá k rozostření obrazu. To je výhodné především při detekci změn mezi dvěma obrázky, kdy díky tomuto filtru omezíme vliv náhodných chyb kamerového snímače.



Obr. 14 Příklad využití funkce `gaussianBlur()`

### 3.1.3 CxAux

Tento modul rozšiřuje knihovnu o experimentální algoritmy, které nejsou nikde dopodrobna zdokumentovány. Pokud je budeme chtít použít, je nutné zajít hlouběji do samotné knihovny a prozkoumat jednotlivé funkce v hlavičkových souborech. Pro představu tento modul obsahuje sledovací funkce ve 3D prostoru a markovovský model vhodný pro segmentaci obrazu. Na následující obrázku můžeme vidět využití funkce `FindStereoCorrespondence()`, která hledá rozdíly ve 2 vstupních obrazech. V levém spodním vidíme výstupní obraz této funkce a vpravo rozdílový snímek vytvořený pomocí funkce `sub()`. (18)



Obr. 15 Příklad využití funkce `findStereoCorrespondence()` a `sub()`

### 3.1.4 MLL

Předposlední je knihovna MLL (Machine Learning Library). Jedná se o modul strojového učení, který pokrývá statistické klasifikátory a regresní metody. Využití takové funkce můžeme v případě, že po programu požadujeme například rozeznání určitých vzorů. Těmi mohou být auta, lidé, značky nebo jakékoliv jiné objekty. Jejich rozpoznání vyžaduje velké množství obrázků, které obsahují nebo neobsahují dané vzory. Z těch se program daný vzor naučí a podle něj je pak schopný rozeznávat požadované objekty.

### 3.1.5 HighGui

Poslední modul slouží k zajištění komunikace mezi OpenCV a operačním systémem, nad kterým je daný program spuštěn. Obsahuje funkce pro vytvoření grafického rozhraní a práci s obrazem.

#### Grafické rozhraní

V dnešní době většina aplikací používá grafická rozhraní pro zprostředkování informací mezi programem a uživatelem. To je umožněno díky výstupním zobrazovacím zařízením a vstupním zařízením připojených k počítači. Tento modul disponuje funkcí `namedWindow()` pro vytvoření klasického okna aplikace. Jelikož součástí grafických rozhraní jsou i funkce pro práci s klávesnicí či myší, tak i zde nalezneme vše potřebné pro jednoduché ovládání programu. Funkce `waitKey()` nám umožňuje počkat na jednotlivá stisknutí klávesnice a naopak funkce `setMouseCallback()` je volána při použití myši. Na následujícím obrázku můžeme vidět příklad jednoduchého grafického rozhraní vytvořeného pomocí tohoto modulu. Obsahuje tlačítko, textová pole, seznam s výběrem a jednoduché kontextové menu. (18)



Obr. 16 Příklad jednoduchého GUI v OpenCV

Přestože OpenCV disponuje základními funkcemi pro tvorbu grafických rozhraní, nehodí se pro složitější aplikace. V případě větších projektů je programování značně náročné, a proto nebudeme používat daný modul pro tvorbu grafického rozhraní kamerového systému.

### Funkce pro práci s obrazem

Kromě grafického prostředí zde nalezneme i funkce, které se starají o komunikaci s kamerami. K tomu slouží třída `VideoCapture` a její jednotlivé třídni metody. Jako první, pokud budeme chtít pracovat s daty z kamery, si vytvoříme instanci objektu `VideoCapture`. Metoda `VideoCapture::open()` zajistí navázání komunikace s kamerou a umožní nám načítat jednotlivé obrázky pomocí metody `VideoCapture::read()`, která vrací poslední snímek z kamery. Metoda `VideoCapture::get()` nám může poskytnout jednotlivé informace o kameře, jako například počet snímků za vteřinu, rozlišení či formát příchozích dat. Je třeba si kontrolovat, jaké hodnoty parametrů kamery z dané funkce obdržíme. V některých případech nejsou kamery schopné poskytnout relevantní data a funkce vrací nesmyslné hodnoty.

V určitých případech je žádoucí místo kamerového výstupu použít video soubor s vhodným záznamem. Toho můžeme dosáhnout pomocí stejného přístupu jako v předešlé části, jen s tím rozdílem, že místo adresy kamerového kanálu zadáme název souboru. To se používá například v případě testování aplikace, kdy nám kamera neposkytuje vhodná data pro konkrétní simulaci. Kromě čtení videosouborů zde najdeme i funkce pro ukládání videí či jednotlivých obrázků. (18)

## 4. Návrh kamerového systému

V rámci této práce budeme využívat kamerový systém tvořený čtveřicí kamer Vivotek PT8133W. Kamery je nutné vhodně propojit s počítačovým systémem tak, aby nedocházelo ke zbytečné ztrátě dat a přenosová rychlost nebyla silně závislá na počtu aktivních kamer.

### 4.1 Technické parametry kamery PT8133W

Pro záznam obrazu využijeme čtveřicí IP kamer Vivotek PT8133W dodané školou pro účely projektu. Tato kamera disponuje ethernetovým a Wi-Fi rozhraním, díky nimž můžeme čtveřicí kamer připojit k počítači, či využít bezdrátovou síť notebooku. Maximální rozlišení kamery je plně dostačující pro potřeby běžných kamerových systémů. Velkou výhodou představuje možnost otáčení kamery jak v horizontální, tak vertikální rovině, a díky tomu můžeme sledovat pohybující se objekty či prohledávat větší místnosti.

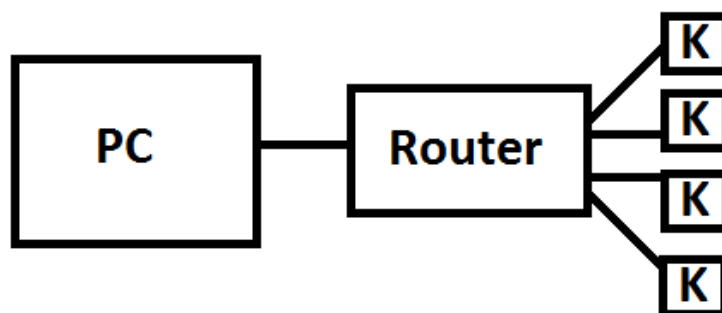


Obr. 17 Kamera Vivotek PT8133W (6)

Typ kamery	PT
Spotřeba	cca 5.1W
Rozsah otáčení	-175°- 175°
Rozsah naklápění	-35 – 90°
Síťové rozhraní	10/100 Mbit/s Ethernet, RJ45
Snímací čipu	¼" CMOS
Rozlišení	Max. 1280 x 800 px
Snímkovací frekvence	Max. 30 snímků/s při max. rozlišení
Komprese videa	MJPEG, MPEG-4, H.264,
Integrovaný objektiv	Ohnisková vzdálenost 3.6 mm,
Horizontální úhel záběru	58.39°
Vertikální úhel záběru	35.58°

Obr. 18 Tabulka technických parametrů kamery PT8133W4.2 Propojení systému s PC (6)

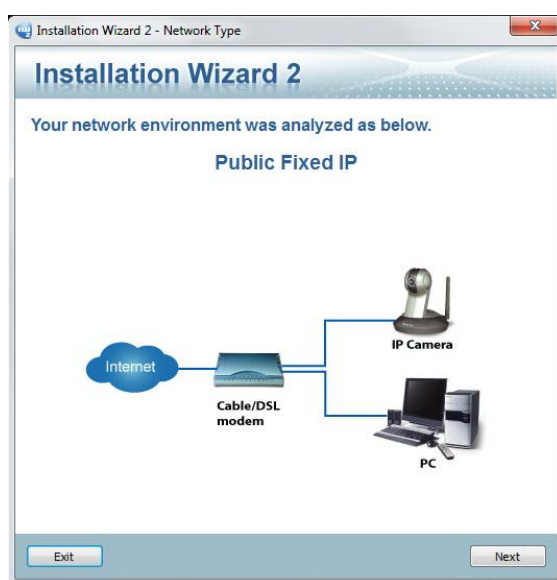
Kamery, které budeme využívat pro inteligentní kamerový systém, disponují dvěma přenosovými médii, a to Wi-Fi a ethernet. Při testování kamer propojených pomocí Wi-Fi začne při větším počtu současně spuštěných kamer docházet k zahlcení přenosového kanálu, což způsobí výrazné zpomalení. Počet snímků za vteřinu klesne na takovou hranici, kdy už by výsledná aplikace mohla mít problémy se správnou detekcí pohybu v obraze. Proto jsem se rozhodl použít propojení jednotlivých kamer pomocí síťového směrovače připojeného k počítači. Tím dosáhneme výrazně vyšší přenosové rychlosti a kamery je možné používat s větším rozlišením. Na následujícím obrázku můžeme vidět schéma zapojení výsledného systému.



Obr. 19 Schéma připojení kamerového systému k počítači

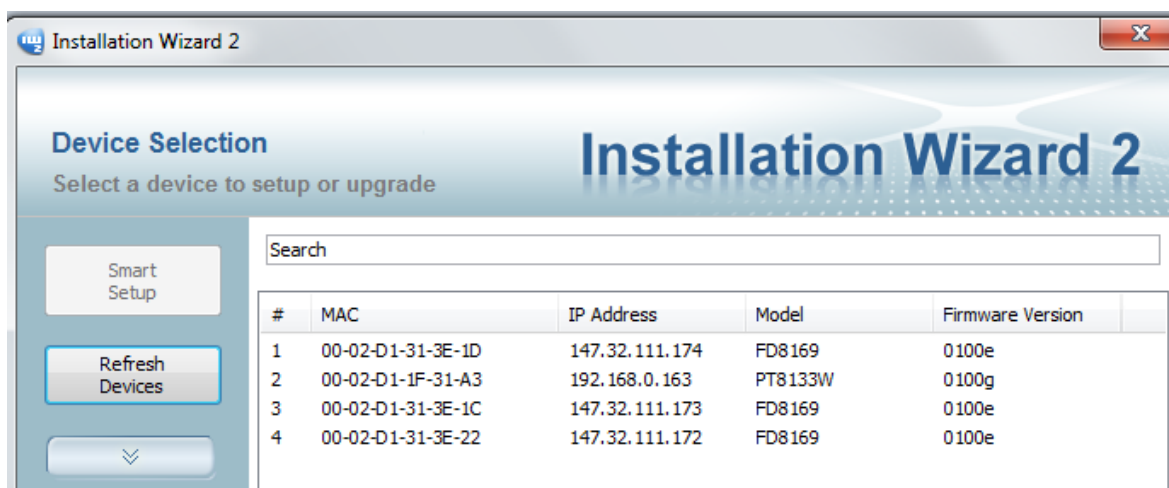
## 4.2 Ovládání kamery pomocí PC

Spolu s kamerou je dodáván software pro její nastavení a následnou obsluhu. Po připojení kamery k počítači pomocí ethernetového kabelu a instalaci programu Installation Wizard 2 si program načte informace o síti, ke které je připojen a poté se dostaneme do hlavnímu menu aplikace.



Obr. 20 Installation Wizard 2 - prohlédávání sítě

Na následujícím obrázku vidíme hlavní menu aplikace Installation Wizard 2.



Obr. 21 Hlavní menu aplikace Installation Wizard 2

Nachází se zde seznam všech kamer připojených do stejné sítě jako počítač. IP adresy jednotlivých kamer budeme poté využívat k jejich adresování ať už ve webovém prohlížeči či vytvořené aplikaci pro tento systém. Můžeme si zde také všimnout tří připojených kamer FD8169, které jsou součástí kamerového systému na Sinkuleho koleji. Dále tato aplikace umožňuje základní nastavení jednotlivých kamer. Nalezneme zde nastavení napájení přes ethernet, nastavení ethernetové či Wi-Fi sítě, ale také například zabezpečení proti cizímu vniknutí pomocí unikátního hesla. Při dvojkliku na některou z kamer aplikace spustí webový prohlížeč s adresou zvolené kamery.

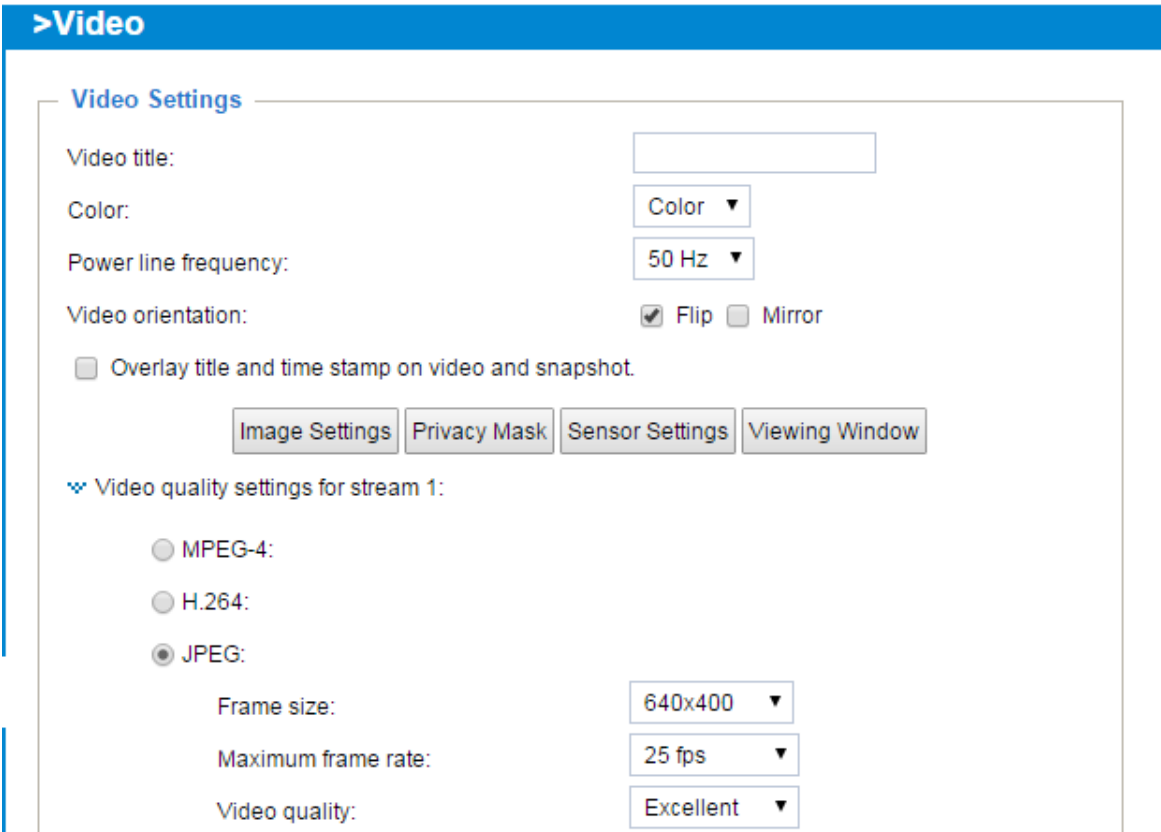


Obr. 22 Webové prostředí IP kamery



Jelikož webové prostředí slouží v našem případě pouze k nastavení základních parametrů kamery, tak zde nebude popsáno do detailů.

Dané prostředí umožňuje ovládat kameru jak ve vertikálním, tak horizontálním směru. V případě že vlastníme ovládací pult připojení k počítači, je možné namísto šipek používat joystick. K nastavení rychlosti otáčení slouží kolonky Pan speed a Tilt speed. Pod záložkou Configuration nalezneme menu, ve kterém je možné změnit veškeré parametry kamery. V našem případě bylo třeba nastavit síťovou frekvenci na 50 Hz pro snímání obrazu bez rušivých obrazců. Dále jsme převrátili obraz pro správné zobrazení na displeji. Pro další nastavení bylo třeba aktivovat pokročilý režim, který umožňuje vybrat vhodnou kompresi obrazových dat. V našem případě jsme použili kompresní metodu JPEG, která měla nejmenší zpoždění. Jelikož práce s obrazem je velice náročná a maximální rozlišení by mohlo aplikaci zpomalit, bylo rozlišení kamery sníženo na 600 x 400 pixelů s frekvencí 25 snímků za vteřinu.



The screenshot displays the 'Video Settings' configuration page. At the top, there is a blue header with the text '>Video'. Below this, the 'Video Settings' section is enclosed in a light gray border. It includes several input fields and checkboxes: 'Video title' (text input), 'Color' (dropdown menu), 'Power line frequency' (dropdown menu set to '50 Hz'), 'Video orientation' (checkboxes for 'Flip' and 'Mirror', with 'Flip' selected), and an unchecked checkbox for 'Overlay title and time stamp on video and snapshot'. Below these are four buttons: 'Image Settings', 'Privacy Mask', 'Sensor Settings', and 'Viewing Window'. A section titled 'Video quality settings for stream 1:' contains three radio buttons for 'MPEG-4:', 'H.264:', and 'JPEG:', with 'JPEG:' selected. Underneath are three dropdown menus: 'Frame size' (set to '640x400'), 'Maximum frame rate' (set to '25 fps'), and 'Video quality' (set to 'Excellent').

Obr. 23 Nastavení výstupního videa z kamery

## 4.3 Ovládání kamery pomocí URL příkazů

K ovládání kamery je možné kromě grafického webového rozhraní použít také speciální URL příkazy, jejichž kompletní seznam se nachází v manuálu přiloženém ke kameře. To poskytuje znalostem uživateli možnost vytvořit si vlastní webové stránky pro ovládání kamery či navrhnout aplikaci, která díky těmto příkazům bude moci kameru plně ovládat. Příkazy můžeme rozdělit do dvou skupin. První skupina je tvořena příkazy pro nastavování jednotlivých parametrů kamery. Za každým takovým příkazem vysílá kamera odpověď, zda došlo k správnému nastavení. Druhou skupinu tvoří příkazy pro zjištění, v jakém stavu se daný parametr nachází. To znamená, že kamera vrací hodnotu požadované proměnné. Znalost syntaxe příkazů je potřebná v následující části, jelikož jednotlivé příkazy budeme využívat v naší aplikaci. Nyní se seznámíme s obecnou syntaxí pro jednotlivé příkazy.

```
http://<servername>/cgi-bin/<subdir>[/<subdir>...].cgi  
[?<parameter>=<value>[&<parameter>=<value>...]]
```

### servername

Zde se doplní IP adresa kamery, kterou můžeme nalézt v seznamu přiloženého programu Installation Wizard 2.

### subdir

Slouží k určení vlastností příkazu. Jednak je zde možné nastavit práva uživatele, jednak zda bude příkaz vracet nějakou proměnnou nebo ji naopak nastavovat.

### parameter

Zde zadáme název parametru, který chceme poslat zpět či nastavit.

### value

Hodnota, na kterou se má daný parametr změnit. V tomto případě pošle kamera zpět informaci, zda byl parametr nastaven.

## 4.4 Ukázka nejpoužívanějších příkazů

### Ovládání natáčení kamery

`http://<servername>/cgi-bin/viewer/camctrl.cgi?channel=0&camid=0&move=<směr>`

Za směr dosadíme left, right, up, down nebo home pro patřičný směr.

### Ovládání indikačních diod kamery

`http://<servername>/cgi-bin/admin/setparam.cgi?system_ledoff=<value>`

Za value dosadíme buď 1 pro vypnutí, či 0 pro zapnutí.

### Ovládání automatického vyvážení bílé barvy

`http://<servername>/cgi-bin/admin/setparam.cgi?videoin_c0_whitebalance=<value>`

Pokud za value dosadíme manual, uloží se aktuální nastavení. Pokud bychom dosadili auto, tak si kamera bude automaticky regulovat vyvážení bílé barvy v závislosti na světelných podmínkách.

### Nastavení rychlosti otáčení

`http://<servername>/cgi-bin/admin/setparam.cgi?camctrl_c0_<směr>speed=<value>`

Pro výběr horizontálního otáčení dosadíme za směr **pan** u vertikální pak **tilt**.

Rychlost v daném směru je dána hodnotou dosazenou za value, která může být v rozmezí od -5 do 5.

### Nastavení dokovací pozice

Tento příkaz se nenachází v žádném manuálu a bylo nutné jej najít ve zdrojovém kódu výše zmíněného webového prostředí kamery.

`http://<servername>/cgi-bin/viewer/camctrl.cgi?channel=0&camid=0&sethome=define`

Díky němu je možné nastavit výchozí pozici, do které se má kamera vrátit při zavolání příkazu pro natáčení kamery s parametrem home.

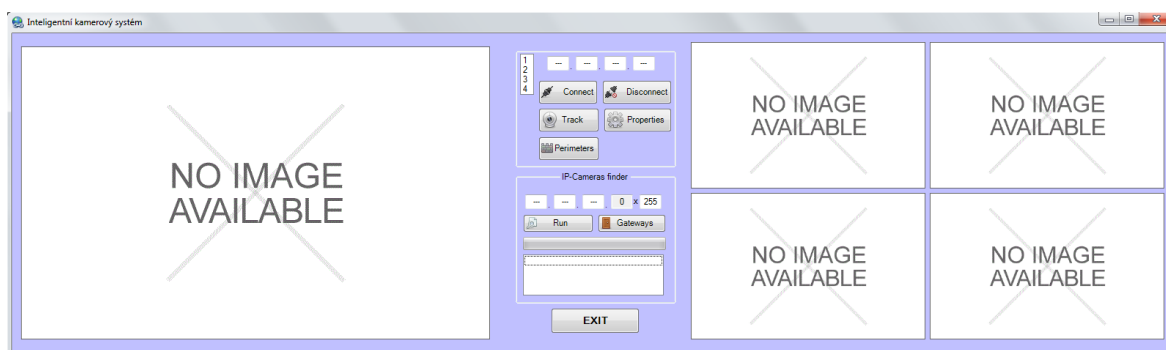
## 5. Návrh aplikace pro kamerový systém

V této kapitole bude popsána praktická část diplomové práce zaměřená na vytvoření aplikace pro inteligentní kamerový systém. Pro návrh aplikace bylo zvoleno prostředí Visual Studio 2013, které nám umožňuje vytvářet aplikace pomocí programovacího jazyka Visual C++. K tvorbě aplikace byly využity volně dostupné knihovny OpenCV a WinHttpWrapper. Díky tomuto prostředí a knihovnám můžeme vytvořit nejen komplexní program, ale také funkční grafické rozhraní, které by v samotném OpenCV bylo velice těžké vytvořit.

Celý program je napsán ve třech hlavičkových souborech. Nejdůležitějším z nich je Cameras.h, který obsahuje hlavní menu, funkce pro odesílání emailu a komunikaci s kamerou. Dále zde najdeme jádro algoritmu pro sledování osob. Následuje hlavičkový soubor Perimeter.h, který definuje okno pro nastavení, ukládání a nahrávání dat potřebných pro určení kritických zón. Poslední soubor Properties.h obsahuje funkce pro odesílání URL příkazů, natáčení kamer a základní informace o jednotlivých kamerách.

### 5.1 Hlavní menu

Hlavní menu představuje jádro celé aplikace. Nachází se zde prvky pro zobrazování výstupního obrazu z kamer, box s tlačítky pro ovládání celé aplikace a box s nástrojem vyhledávání jednotlivých IP adres.



Obr. 24 Hlavní menu aplikace inteligentního kamerového systému

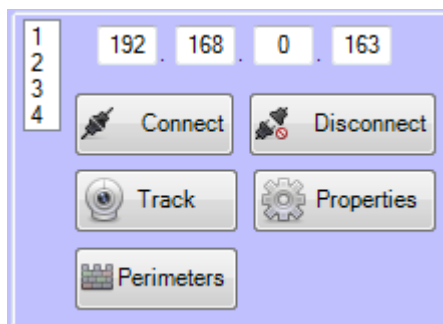
#### 5.1.1 Zobrazovací okna

Na výše uvedeném příkladu úvodní obrazovky si můžeme povšimnout pětice ploch pro zobrazení kamerových výstupů. Na pravé straně je umístěna čtveřice oken, která slouží pro méně detailní zobrazení všech připojených kamer současně o polovičním rozlišení. V případě nového spuštění aplikace či odpojení kamery od systému pomocí tlačítka Disconnect, jsou všechna okna inicializována textem NO IMAGE AVAILABLE. Levé okno slouží k zobrazení jednoho kamerového výstupu v plném rozlišení. Pokud při běhu aplikace chceme zobrazit daný kamerový výstup na velké zobrazovací okno, stačí jednoduše kliknout na jedno z malých oken.

Jelikož obrazová data jsou snímána pomocí knihovny OpenCV, je nutné pro jejich použití ve Visual Studio využít konverzní funkci Bitmap().

## 5.1.2 Box s jednotlivými funkcemi

Program je koncipován tak, aby po zapnutí uživatel připojil jednotlivé kamery, nastavil parametry daných kamer, označil kritické zóny pro sledování a zapnul sledovací mód. Všechny tyto funkce nalezneme v boxu na následujícím obrázku.



Obr. 25 Box s jednotlivými funkcemi programu

Po levé straně je umístěn seznam s čísly, pod které je možné danou kameru zaznamenat. Jelikož systém je koncipován pro maximálně 4 kamery, tak zde nalezneme čísla od 1 do 4. Pokud budeme chtít pokračovat zmáčknutím jednoho z 5 tlačítek, nejdříve musíme označit odpovídající číslo, pod jakým chceme danou akci vyvolat. Jednotlivé funkce aplikace si následně popíšeme.

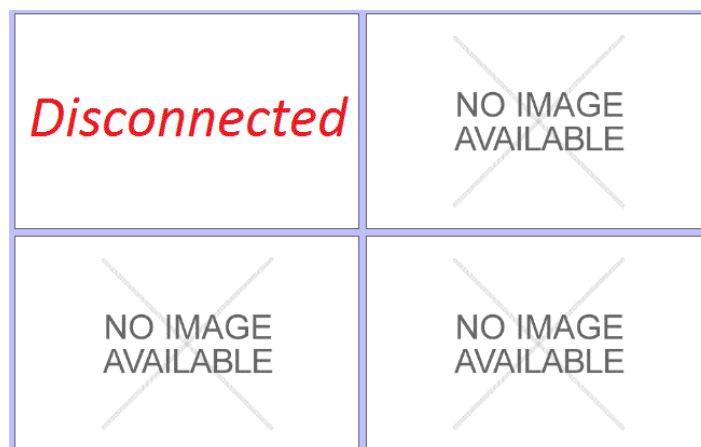
### Connect

Prvním krokem ke zprovoznění kamerového systému je připojení kamery a následné snímání obrazu. Pokud dosud neznáme IP adresy jednotlivých kamer, můžeme si je zjistit, ať už pomocí zmíněného Installation Wizard 2 nebo pomocí nástroje pro vyhledávání kamer zabudovaného v aplikaci. Poté co jsme si již zjistili adresy jednotlivých kamer, můžeme přejít k samotnému navázání komunikace. Jako první vybereme ze seznamu číslo, pod které chceme danou kameru uložit, a opišeme adresu do čtyř kolonek k tomu určených. K zabránění uživatelských chybám zde byla implementována ochrana, která dovoluje psát pouze číslice a umožňuje použití tlačítka Backspace pro mazání. Při následném stisku tlačítka Connect se zahájí navázání komunikace a výsledek této operace je znázorněn obrázkem v pravém dolním rohu. Příklady, kdy došlo či nedošlo ke spojení, můžeme vidět na následujícím obrázku, kde v levé části je případ odpojené kamery od systému.



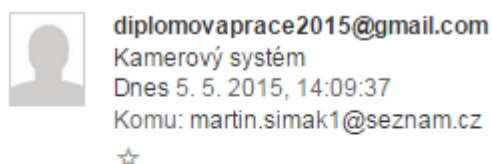
Obr. 26 Příklad neúspěšného a úspěšného připojení kamery

Pokud spojení proběhne úspěšně, je obraz automaticky zobrazován na jednu z obrazových oblastí v pravé části aplikace. V případě, že nastane situace, kdy je spojení navázáno správně, ale aplikace nedostává žádná obrazová data, tak program uživateli tuto skutečnost oznámí velkým nápisem *Disconnected* v příslušném obrazovém bloku. To může nastat v situacích, kdy je přerušeno spojení mezi kamerou a aplikací nebo pokud je přenos příliš pomalý. S tím se setkáme při použití přenosu přes Wi-Fi síť, která není vhodná pro přenosy takto velkých datových toků u zapojení s více kamerami. Situaci s připojenou kamerou s následným odpojením můžeme vidět na následujícím obrázku.



Obr. 27 Příklad násilného odpojení kamery

Poté během několika málo vteřin aplikace odešle informační zprávu o odpojení kameře. Součástí e-mailu je číslo dané kamery a čas, v jakém došlo k výpadku, který se může lišit od času doručení e-mailové zprávy. To je výhodné především u systémů, které nad sebou nemají trvalý dohled dispečinku. Na následujícím obrázku můžeme vidět příklad e-mailové zprávy při násilném odpojení kamery.



Došlo k odpojení kamery číslo 1  
5.5.2015 14:09:26

Obr. 28 E-mail oznamující přerušení spojení s kamerou

Při opětovném navázání spojení aplikace pokračuje v předešlé činnosti. Na tomto místě však může vzniknout menší problém. Jakmile je kamera fyzicky odpojena od aplikace, není možné ukončit funkci, která se stará o snímání jednotlivých obrázků. Tato chyba je dána vnitřním uspořádáním objektu VideoCapture z knihovny OpenCV. Chyba byla nahlášena skupině zastupující tuto knihovnu a v následujících verzích bude pravděpodobně opravena. Díky použití více vláknového programování však tato chybová situace nebrání zbytku systému v běžné činnosti, přesto je třeba na ni brát zřetel při psaní programu.

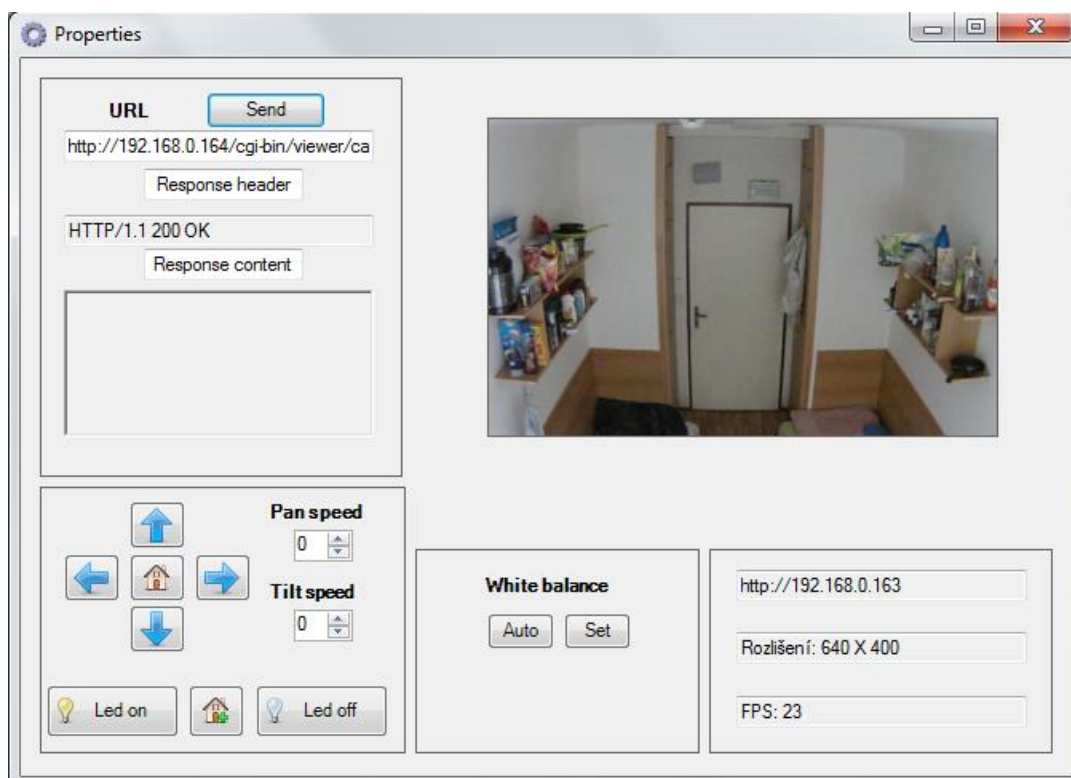
## Disconnect

Pokud již nechceme danou kameru používat, opět vybereme ze senamu číslo pod kterým je uložena a stiskneme tlačítko Disconnect. Následně se volaná funkce `button4_Click()` postará o nastavení příznaku `uvolnit[]` a `tracking[]` do logické 0. Díky použití příznaku je odpojení kamery zajištěno i v případech výpadku spojení. V situaci, kdy je pro danou kameru spuštěn sledovací mód, je také bezpečně ukončen.

V předešlém případě byla naznačena chyba OpenCV týkající se programového odpojení kamery, která v průběhu přenosu dat byla fyzicky odpojena, a program bude čekat, dokud se spojení opět neobnoví. Pokud bychom chtěli místo dané kamery připojit jinou, nebylo by to možné, jelikož Visual Studio nepodporuje ukončení vlákna, které čeká na obrázek od kamery, která již není k systému připojena. Při použití tlačítka Disconnect aplikace pouze počká na opětovné navázání komunikace a poté ukončí dané vlákno.

## Properties

Následující okno slouží k ovládání jednotlivých kamer a nastavení důležitých parametrů. Jak už bylo zmíněno, používáme vertikálně a horizontálně ovládané kamery. K jejich ovládání je třeba odeslat příslušný URL příkaz, po jehož odeslání kamera odpoví, zda byl proveden správně. Pro tyto účely byl použit hlavičkový soubor `WinHttpClient.h`, který obsahuje funkce pro odesílání a přijímání URL zpráv. Samozřejmostí tohoto okna je plocha pro zobrazení kamerového obrazu.



Obr. 29 Okno Properties

V levém horním rohu je umístěn box pro manuální odesílání zpráv. Obsahuje textové pole pro zadání konkrétního příkazu a tlačítko send pro jeho odeslání. Dále zde máme dvě pole pro zobrazení obdržené zprávy od kamery. Jedno pro zobrazení hlavičky a druhé pro obsah zprávy.

V levém spodním rohu jsou obsaženy nejpoužívanější příkazy, které se skrývají za jednotlivými tlačítky. Jsou zde šipky pro otáčení kamerou s domečkem pro vrácení na původní pozici. Do kolonek pan speed a tilt speed je možné zadat rychlost otáčení od -5 do 5. Tlačítka Led on a Led off slouží k zapnutí či vypnutí indikačních diod jednotlivých kamer. Jako poslední zde najdeme domeček se zelenou značkou plus, který nastaví domovskou pozici, na kterou se kamera vrátí při kliknutí na domeček umístěný výše.

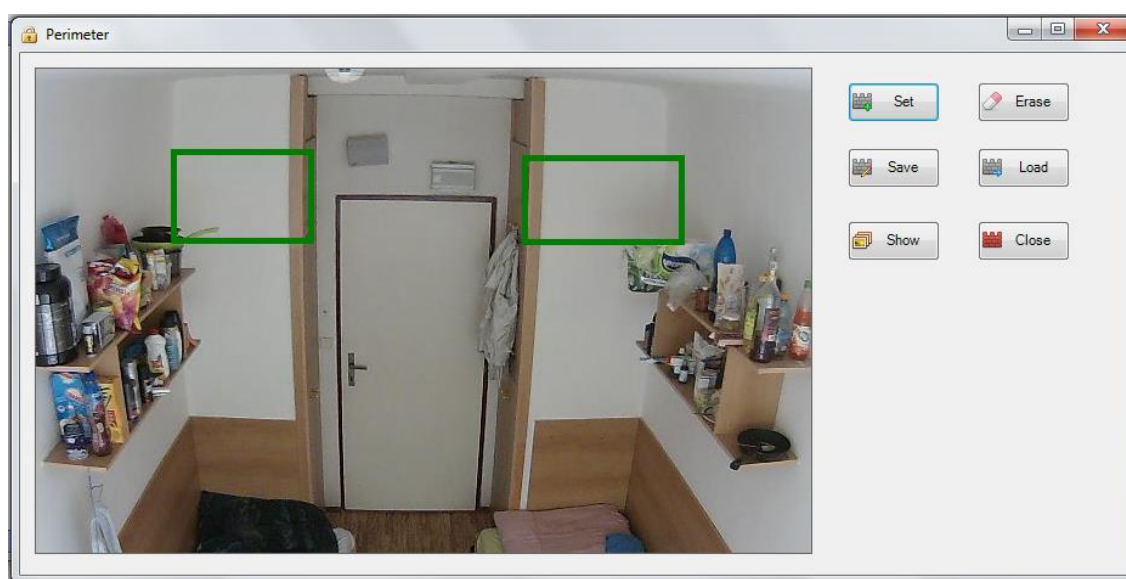
Prostřední spodní box obsahuje dvě tlačítka pro nastavení automatického vyvážení bílé. Při automatickém režimu zvoleném pomocí tlačítka Auto je vyvážení bílé plně řízeno kamerou. Při stisknutí tlačítka Set se uchováni aktuální nastavení a kamera již tento parametr nemění.

Jako poslední zde najdeme základní informace o kameře. Mezi ně patří webová adresa pro přihlášení do rozhraní prohlížeče, rozlišení obrazových dat a snímková frekvence, která je počítána při běhu programu a vypisována do tohoto textového pole.



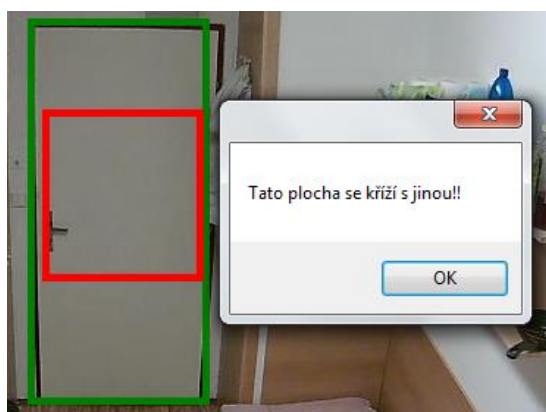
## Perimeters

Jedním z bodů zadání je definice perimetrů kritických zón. Okno na následujícím obrázku je určeno právě k tomuto účelu. V levé části se nachází oblast jednak pro zobrazení obrazu z kamery, jednak pro označení kritických zón pomocí myši. Po stisku tlačítka Set, přejdeme do procesu definice těchto částí obrazu. Následně vytvářením čtvercových ploch si označíme potřebné oblasti. Ty je možné smazat pomocí tlačítka Erase nebo uložit pomocí tlačítka Save. Pokud jsme si již provedli definici s následným uložením, je možné pomocí tlačítka Load nahrát všechny informace do aplikace a následně pomocí tlačítka Show nahrané oblasti zobrazit. Bez zavření okna tlačítkem Close či křížku není možné vyvolat nové okno pro stejnou či jinou kameru z důvodu ochrany před simultánním zápisem do jednoho souboru, v němž jsou data uložena.



Obr. 30 Okno pro definici prítických zón

Program disponuje funkcí Prekrytí(), která kontroluje, zda nedošlo k definici již definované oblasti nebo její části. V případě nesprávné definice program upozorní uživatele chybovou hláškou a umožní mu pokračovat v inicializaci jednotlivých zón.



Obr. 31 Chyba při špatné definici

Takto definované zóny nám poslouží v následující části ke sledování označených objektů či míst se zákazem vstupu.

## Track – sledovací mód

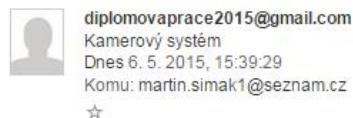
Jelikož se má jednat o inteligentní kamerový systém, byla zde implementována funkce pro sledování objektů v určitém prostoru využívající námi definované kritické zóny. Pokud jsme úspěšně připojili kameru, po stisknutí tlačítka Track je spuštěn nástroj pro sledování pohybujících se objektů. Na následujícím obrázcích můžeme vidět příklad takového použití ve čtyřech situacích. Objekty pohybující se v obraze jsou orámovány zelenými obdélníky. V případě narušení perimetru je k orámování objektů využito červené barvy. Takové narušení je signalizováno jednak spuštěním alarmu, ale také e-mailovou zprávou s fotkou daného objektu. Následně je obrázek narušitele ještě uložen do souboru Intruders.



Obr. 32 Ukázkové situace sledovacího systému

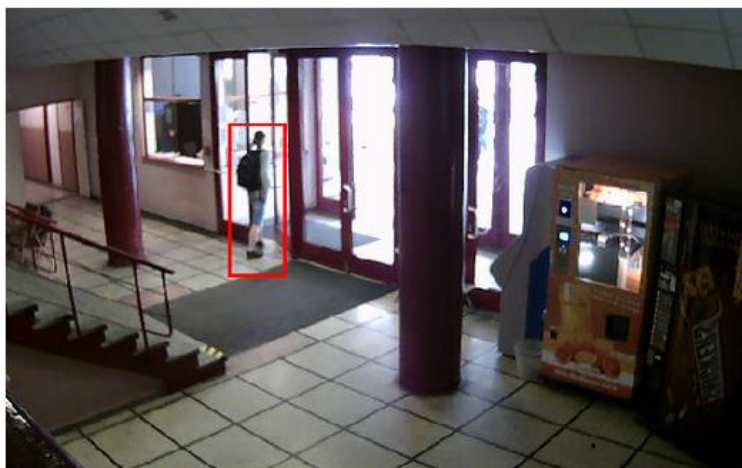
Můžeme si povšimnout, že program je schopný úspěšně identifikovat jednotlivé osoby. Problém nastává, pokud se osoba vyskytuje poblíž jiného pohybujícího se objektu, jako jsou například vstupní dveře v pravého horním obrázku nebo stíny okolo pohybujících se postav.

Na následujícím obrázku můžeme vidět příklad obdržené e-mailové zprávy při narušení perimetru.



Došlo k narušení perimetru  
6.5.2015 15:39:17

Přílohy



intruder.jpg (95 kB) Stáhnout

**Obr. 33 E-mailová zpráva informující o narušení perimetru**

Obdržení takové informace je výhodné v situacích, kdy sledujeme prostory s částečným zákazem vstupu. Díky obdržené zprávě můžeme posoudit, zda daná osoba je skutečným narušitelem a případně přivolat policii.

Jelikož odeslání e-mailové zprávy může být zpožděno, je kromě aktuálního obrazu přidána informace o době vzniku kritické události.

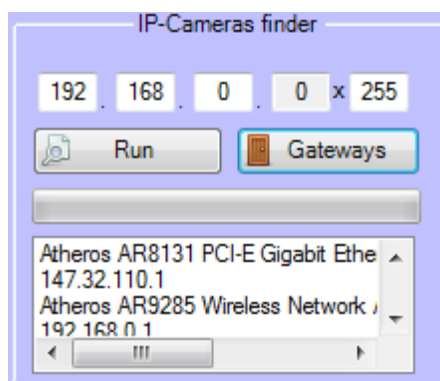
V roce 2015, kdy jsou na vzestupu chytré telefony s mobilním internetovým připojením, představuje tento nástroj jeden z nejrychlejších způsobů jak informovat vlastníka objektu o narušení hlídaného perimetru. Využití toho nástroje nemusí být omezeno pouze pro tyto případy. Funkci je možné upravit ke konstantnímu odesílání kamerového obrazu do příslušné e-mailové schránky, a tak si zajistit neustále informace o dění v jednotlivých prostorách. Při potřebě neustálého dohledu nad kamerovým systémem, by bylo výhodné vytvořit k počítačové aplikaci ještě aplikaci mobilní. Což představuje jednu z možností v pokračování na této práci.

### 5.1.3 Box pro vyhledání IP adres

Jak již bylo zmíněno, pro připojení kamer k aplikaci musíme znát IP adresu daného zařízení. Tu je možné dohledat pomocí Installation Wizard 2 nebo níže uvedené části programu k tomu určené. Oba následující nástroje využívají více vláknového programování, a díky tomu jejich spuštění nijak výrazně neovlivní běh celé aplikace.

#### Hledání síťových bran

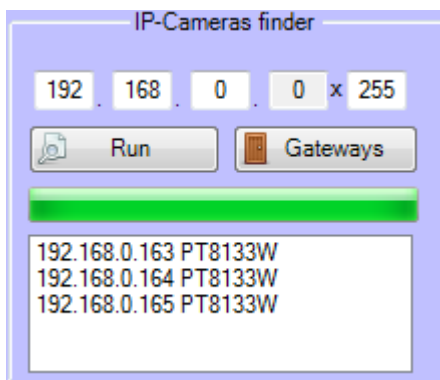
Adresový prostor je dán IP adresou hardwaru, ke kterému je kamerový systém připojen. Jelikož by otestování všech adres zabralo příliš mnoho času, je zde implementován nástroj pro určení vstupních bran počítače a jejich IP adres. Na níže uvedeném obrázku vidíme použití tohoto nástroje pomocí tlačítka Gateways. Program následně získá informace od operačního systému o aktivních síťových prvcích a vypíše jejich jména a IP adresy do textového pole.



Obr. 34 Příklad vyhledání aktivních bran počítače

#### Hledání IP adres

K vyhledání kamer připojených k danému síťovému prvku opíšeme jeho IP adresu získanou v předešlém nástroji. Po stisknutí tlačítka Run je automaticky prohledán adresový prostor o 255 prvcích a do textového pole jsou vypsány IP adresy jednotlivých kamer. Z důvodů větší náročnosti na výpočetní výkon bylo nutné prohledávání jednotlivých adres pozdržet o 1ms. Díky tomu můžou ostatní části aplikace běžet bez zbytečného zpomalení. Tato výhoda je vykoupena zhruba pětinasobnou dobou prohledávání, která se pohybuje okolo 5 vteřin.



Obr. 35 Příklad vyhledání jednotlivých IP adres



## 5.2 Algoritmus pro sledování osob

Při stisku tlačítka Track se spustí pro danou kameru nástroj sledující osoby. Jedná se pouze o obecný příklad, jak řešit daný problém. Pro konkrétní použití je třeba vhodné navolit jednotlivé parametry a uzpůsobit algoritmus daným podmínkám. Jednotlivé kroky tohoto algoritmu si následně popíšeme

### Příprava pozadí

V bezpečnostních systémech nás většinou zajímá detekce a klasifikace lidských postav či nežádoucích objektů. Abychom byli schopni detekovat jak statické, tak pohybující se osoby, je nutné vybrat vhodnou metodu. Výběr je dán nejen povahou pozadí, ale i například úhlem pohledu kamery. Jelikož v našem případě potřebujeme zaznamenávat pohyb v obraze ve čtyřech kanálech najednou, musíme se také zaměřit na výpočetní náročnost použitých algoritmů. OpenCV má implementovanou kvalitní metodu MOG (Mixture of Gaussians), která porovnává jednotlivé snímky a určuje pozadí. Bohužel výkon počítače by v našem případě nestačil pro plynulé sledování všech kanálů. Proto jsem se rozhodl použít metodu odečítání snímků, která je jednak snadno implementovatelná a zároveň nemá velký vliv na rychlost aplikace. Podstata metody spočívá v uložení snímku pozadí na začátku sledování a následném odečítání s aktuálním obrázkem. Uložení snímku je zajištěno funkcí `setBackground()`, kterou je možné použít nejen na začátku sledování, ale po menších úpravách i v průběhu sledování.

Předtím než přejdeme k dalšímu kroku, jsou jednotlivé snímky rozmazány pomocí funkce `GaussianBlur()`, která odstraní z obrazu přebytečný šum. Ten může být dán nedokonalostí kamerového snímače, ale také například slabým pohybem listů či jiných drobných objektů.



Obr. 36 Vlevo obrázek pozadí a vpravo aktuální snímek

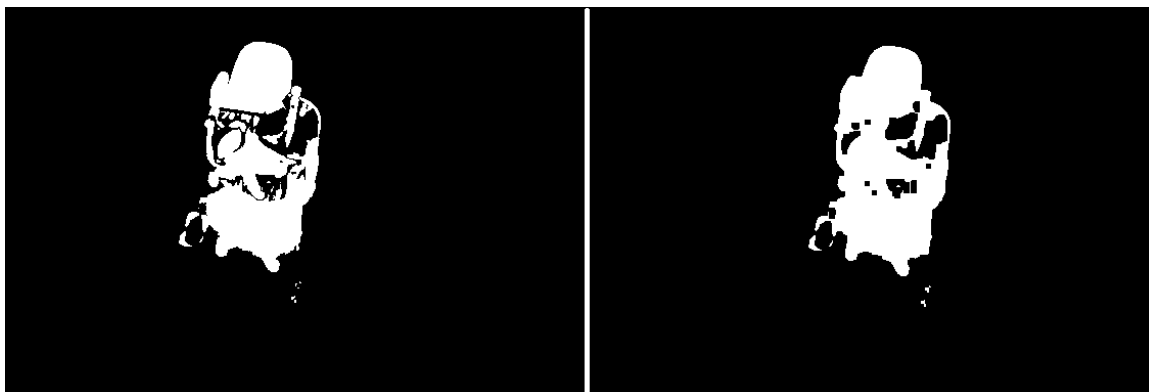
Pokud máme připravený snímek pozadí a aktuální snímek z kamery, následuje vytvoření rozdílového obrazu

## Rozdílový obraz

Dalším krokem, při kterém ve výsledném obrázku získáme informace o pohybu objektů je vytvoření rozdílového obrazu. Ten získáme tak, že vhodným způsobem od sebe odečteme jednotlivé snímky. Barevné obrázky jsou tvořeny třemi maticemi. Jedna pro červenou, druhá pro modrou a třetí pro zelenou barvu. Tady se nám nabízejí dvě možnosti, jak od sebe odečítat 2 různé snímky. Jednak můžeme vzít jednu stejnou barvu obou obrázků, odečíst je od sebe, a tak získat rozdílový obraz, který však není tak kvalitní jako v případě druhého způsobu. V našem případě byla použita druhá metoda, která je založena na stejném principu, ovšem s použitím všech tří barevných vrstev. Z takto získané trojice rozdílových obrázků vytvoříme jeden, kde výsledný pixel pro dané souřadnice má hodnotu buď 0, která představuje, že se pixel nezměnil, nebo 255 určující změnu či pohyb na daném pixelu. O tom jak moc rozdílné musí dva pixely být, aby se jednalo o změnu, rozhoduje následující funkce.

```
if ((pixel1-pixel2)<25){  
    pixel3=0;  
}  
else{  
    pixel3=255;  
}
```

Tomuto procesu se říká prahování, které nám zajistí data, z kterých můžeme snadno rozpoznávat jednotlivé objekty. Výsledek prahování můžeme vidět na levé straně následujícího obrázku.



Obr. 37 Příklad prahování a použití morfologických operací

Jelikož i po této úpravě je obraz hodně zrnitý a bílé plochy netvoří velké celky pro určení objektů, je nutné použít morfologické operace. To jsou speciální funkce, které nafukují či smršťují bílé oblasti, čímž je možné dosáhnout odstranění malých ploch a z jednotlivých nespojených částí vytvořit jednu velkou. Výstup těchto funkcí můžeme vidět na pravé straně výše uvedeného obrázku.

## Hledání kontur

Nyní, když máme připravený obraz, můžeme přejít k použití složitějších funkcí. Dosud jsme pouze operovali s jednotlivými pixely, které představují pohyb v obraze, ale nejsou v aplikaci uloženy jako jednotlivé objekty. K tomu slouží operace, které se říká hledání kontur. Jejím úkolem je najít bílé plochy, vytvořit síť bodů z obvodových pixelů a uložit je jako nový objekt. Nejenže můžeme pak jednotlivé objekty vykreslit, ale také jednoduše spočítat jejich těžiště.

Pro hledání kontur použijeme funkci `findContours()`, která z předešlého snímku vytvoří seznam objektů, s nimiž můžeme jednoduše pracovat. Pokud jsme správně nastavili parametry morfologických funkcí, tak námi sledované objekty by měly být snadno rozpoznatelné například podle velikosti. V opačném případě je nutné upravovat parametry tak dlouho, dokud nedosáhneme přijatelného výsledku. Na následujícím obrázku vidíme vykreslen pouze objekt, který splňuje námi požadovanou velikost.



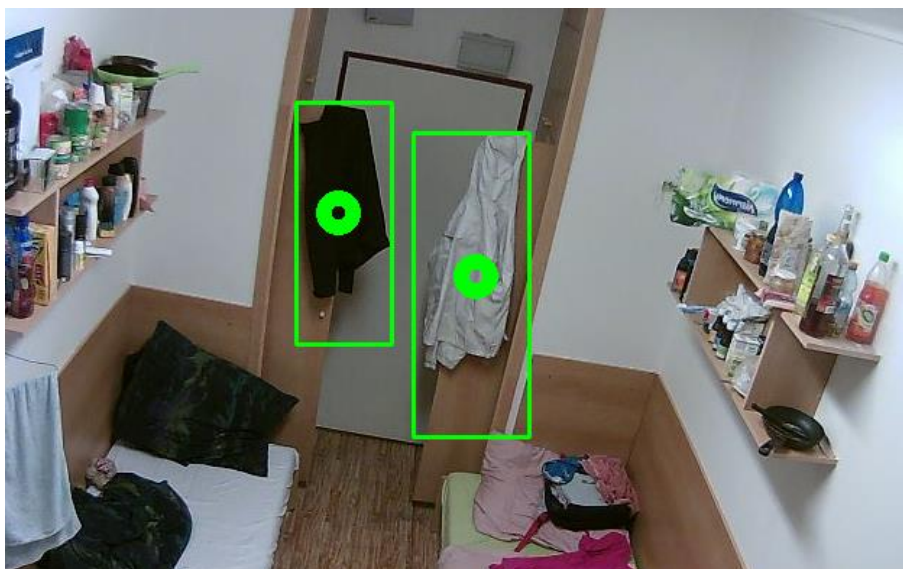
Obr. 38 Vykreslení objektu nalezeného pomocí funkce `findContours()`

V této fázi končí hledání objektů a nastupuje jejich identifikace. Způsobů, jak dojít k tomuto výsledku, je více a je nutné vždy myslet na výpočetní náročnost jednotlivých algoritmů. V našem případě byly použity co nejjednodušší funkce s relativně dobrými výsledky při správném nastavení jednotlivých parametrů.

## Počítání objektů

Další částí procesu sledování je počítání jednotlivých objektů. Jelikož se může stát, že v každém snímku dostaneme různý počet objektů, je nutné pro každý z nich definovat určitou pozici a tu sledovat, abychom se vyvarovali několikanásobnému započítávání.

Proto byl vytvořen algoritmus, který v každém cyklu porovná aktuální seznam objektů se seznamem z předešlého cyklu. Jeho součástí je výpočet geometrického středu, kterým jsou jednotlivé objekty definovány. Podstatou celého procesu je porovnat každý objekt z předešlého snímku a určit jeho vzdálenost s objektem nalezeným na následujícím obrázku. V případě menší než definované vzdálenosti jsou objekty považovány za stejné. V opačném případě je seznam rozšířen o další sledovaný objekt. Výsledek této operace i s vypočteným středem jednotlivých objektů je naznačen na následujícím obrázku.



Obr. 39 Ukázka výpočtu geometrických středů jednotlivých objektů

Takto vytvořený seznam můžeme jednak použít ke sledování jednotlivých objektů, jednak k porovnání s kritickými zónami. Bohužel tento algoritmus je velice obecný a postrádá řešení pro případy, kdy se jednotlivé objekty vyskytují přímo u sebe či opouštějí obrazový prostor. V takových případech nebude seznam souhlasit se skutečným počtem objektů vyskytujících se v daném snímku. Zde se nabízí prostor pro další vylepšení a pokračování v práci.

Budoucímu programátorovi doporučuji definovat plochy, ve kterých bude probíhat identifikace nově přicházejícího objektu, či objektu, který takový prostor opouští. Dále bych doporučil zaměřit se na situace, kdy se dané plochy k sobě přibližují či rozdělují.



## 5.3 Algoritmy pro klasifikaci osob

V předchozí kapitole jsme si ukázaly, jak je možné ve snímaném obraze rozeznat jednotlivé objekty a následně je sledovat. V této práci byli dále vytvořeny dva příklady pro klasifikaci lidských osob, jejichž funkce je popsána níže.

Prvním příkladem je identifikace obličeje z oříznuté plochy detekovaného objektu. K tomu využijeme již připravené kaskádní klasifikátory s různými klasifikačními daty implementované v OpenCV. Úkolem těchto klasifikátorů je prozkoumat celý obrázek a nalézt jednotlivé přechody stínů. Můžeme si povšimnout, že na lidském obličeji je několik přechodů s tmavými a světlými částmi. Jeden takový je například na čele, které odráží velké procento světelného záření. Jelikož se můžeme setkat s různými velikostmi obličejů vzhledem k různé vzdálenosti od sledovaného objektu, musí klasifikátor provést mnoho operací, aby všechny našel. Jelikož by taková funkce mohla velmi zpomalovat celou aplikaci, umožňuje daný klasifikátor nastavit tak, aby nehledal příliš malé nebo velké tváře.

Na následujícím obrázku je příklad použití obličejového klasifikátoru s dvěma různými knihovnamí. Zleva lbpcascades a zprava haarův kaskádní klasifikátor.



Obr. 40 Zleva příklad použití klasifikátoru lbpcascades a zprava haarův kaskádní klasifikátor

Na obou obrázcích byly detekovány obličeje. Jelikož světelné podmínky mají podstatný vliv na úspěšnost detekce, tak v první případě nebyl klasifikátor schopný rozeznat oči. OpenCV dále nabízí nejen detekci obličejů, ale také například úsměvu nebo státních poznávacích značek.

Z důvodu větší náročnosti pro výpočet nedoporučuji používat tyto funkce příliš často, pokud se nejedná například o systém pro udělení vstupu do daného perimetru. Naopak tyto metody je vhodné použít v případě, kdy jsme úspěšně rozeznali lidskou osobu a její obličej je nutné uložit do databáze.

Jako druhý byl pro klasifikaci objektů vybrán HOG (Histogram of Oriented Gradients) klasifikátor, který je implementovaný v OpenCV. Ten je možné se správnou knihovnou použít pro hledání osob. V tomto případě opět využijeme připravený seznam objektů, kde každý jednotlivý objekt prověříme tímto klasifikátorem. Ve výsledku se může stát, že v daném obrázku nalezneme i několik pozitivních nálezů patřící stejnému člověku. V takovém případě je potřeba odfiltrovat přebytečné nálezy a vybrat pouze jeden z nich. Tato metoda je závislá nejen na světelných podmínkách, velikosti postavy, ale také na postoji, který člověk zaujímá. Ve většině případů nám stačí danou osobu identifikovat pouze jednou a poté sledovat její pohyb.

Na následujícím obrázku můžeme vidět použití této metody, kdy došlo k úspěšné identifikaci osoby. Jako u předešlé metody, ani tato není perfektní a její úspěšnost je dána především vhodným použitím jednotlivých algoritmů a správném umístění kamer, které by neměly osoby snímat z velké výšky.

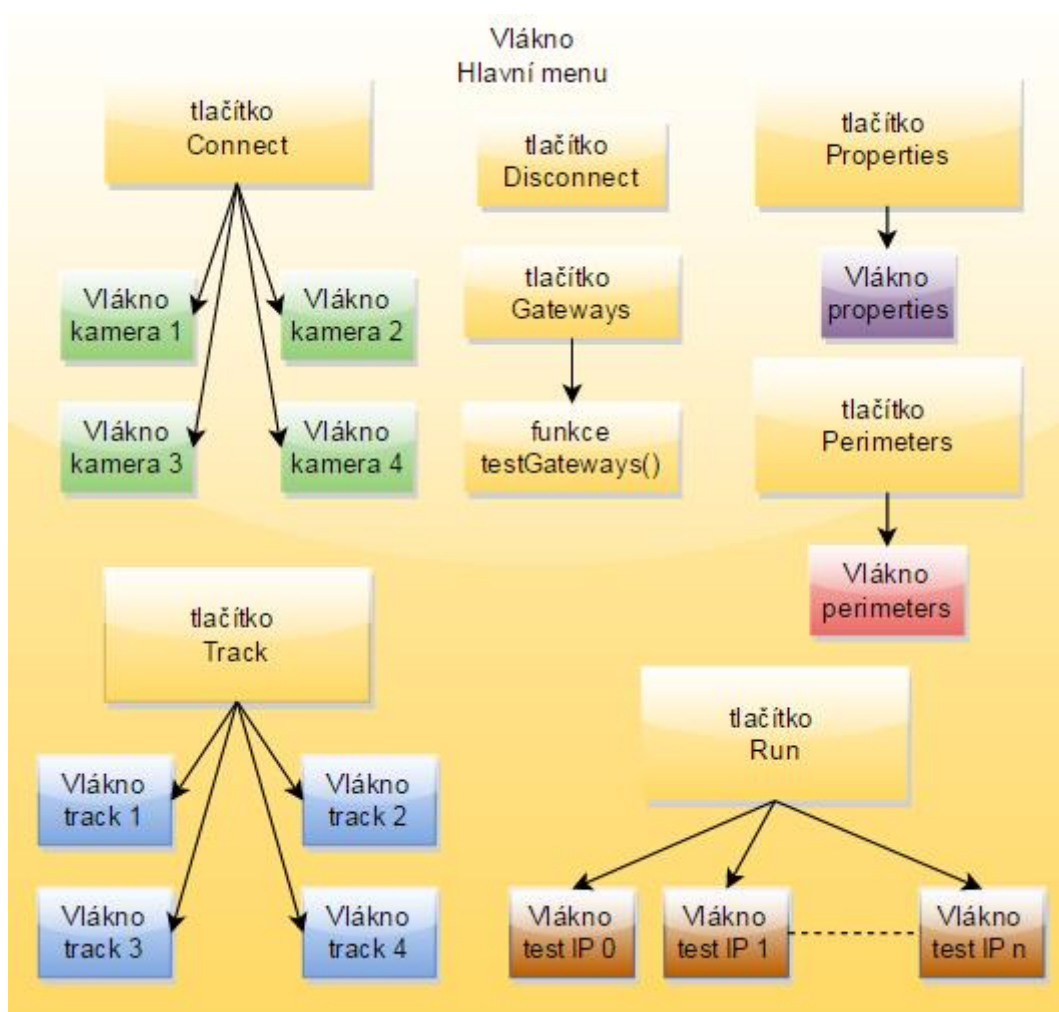


**Obr. 41** Příklad úspěšné detekce lidské postavy pomocí histogramu orientovaných gradientů

Funkce obsahující tuto klasifikační metodu je nejvíce náročným algoritmem v celém projektu. Z toho důvodu je nutné ji používat s rozvahou a neplýtvat výpočetním výkonem.

## 5.4 Blokový diagram

Jak již bylo zmíněno, program je schopný rychle vyhodnocovat jednotlivé funkce díky použití více vláknového programování. Následující diagram vytvořený online na internetovém portálu [www.draw.io](http://www.draw.io), popisuje, jak jsou volány nejdůležitější části programu z hlediska jednotlivých vláken. Nalezneme zde bloky označené jako tlačítko, představující funkce pro správu uživatelských příkazů. Označení vlákno nám říká, že daná funkce je volána jako nezávisle pracující část programu. Můžeme si povšimnout, že pro čtení dat z jednotlivých kamer jsou použita 4 nezávislá vlákna a stejně tak je to i v případě sledování pohybu v jednotlivých obrázcích, pro které mohou být spuštěna také 4 další vlákna. To nám zajistí chod aplikace a sledování objektů v reálném čase.



Obr. 42 Blokový diagram hlavních funkcí

## 5.5 Použití aplikace pro sledování pohybu v budově

Výsledná aplikace, která byla vytvořena, představuje ideální prostředí pro testování monitorovacích algoritmů pro kamerové systémy. Jelikož v ní jsou implementovány i jednoduché sledovací procedury pro nastínění budoucího vývoje této aplikace, je tato kapitola věnována ukázkovému příkladu použití programu v praxi. Modelová situace byla vytvořena ve vstupní hale na Sinkuleho koleji. Na přiloženém CD nalezneme celkem čtyři videa s ukázkou algoritmu pro sledování pohybujících se objektů.

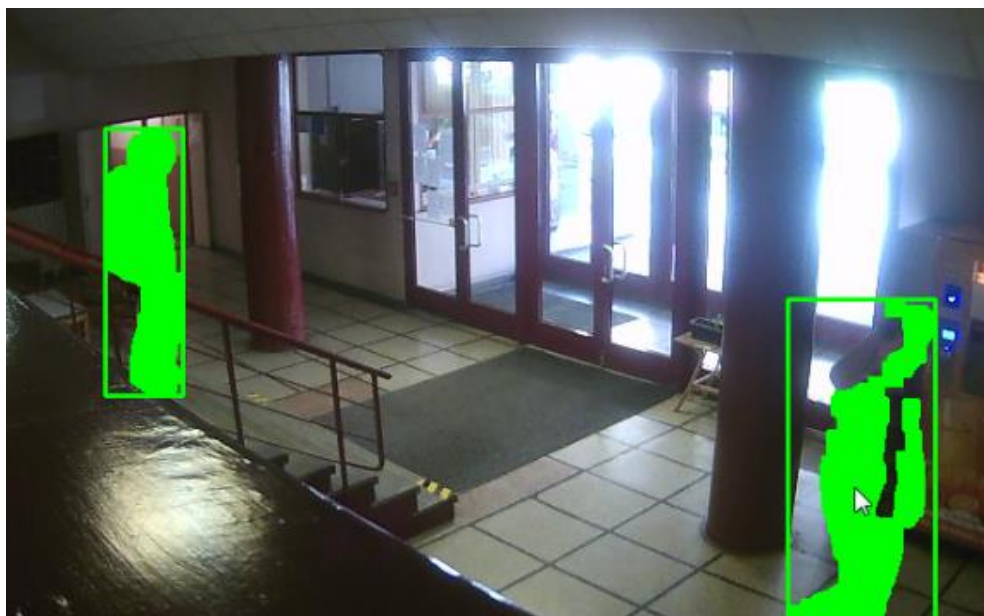
Na prvním videu můžeme vidět pohyb jedné osoby, jejíž oblečení není příliš kontrastní a světelné podmínky také nejsou ideální. Při vhodném nastavení jednotlivých parametrů programu byla osoba úspěšně identifikována, a vyplňující kontura odpovídala pohybující se postavě. V obraze se objevují přebytečné stíny, které jsou pouze částečně eliminovány použitým algoritmem.



Obr. 43 První video aplikace kamerového systému

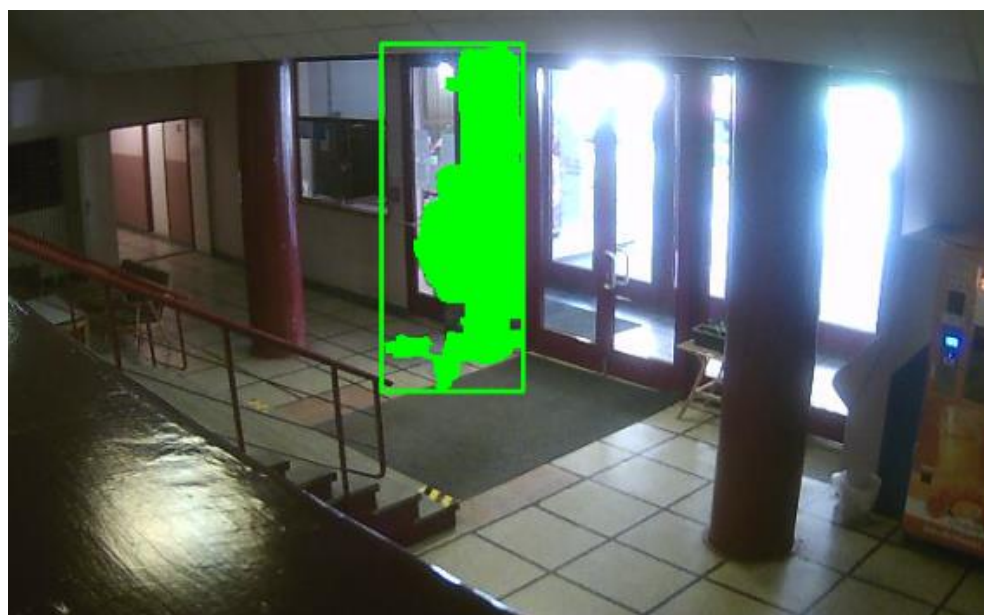
Pro dosažení lepších výsledků, by bylo nutné upravit proces hledání kontur a kromě metody odečítání dvou snímků se porozhlédnout po nějaké doplňkové metodě a jejich kombinací dosáhnou vyšší citlivosti na objekty, které se příliš neliší od pozadí.

Následující druhé video obsahuje modelovou situaci, ve které se nacházejí dvě osoby. V průběhu videa si můžeme povšimnout spojení dvou objektů v jeden. Tento problém je nutné řešit složitějšími algoritmy, které mohou být předmětem dalšího pokračování v této práci. Pokud jsou objekty od sebe dostatečně vzdáleny, je program schopný detekovat a rozeznat neomezené množství osob či pohybujících se objektů.



Obr. 44 Druhé video aplikace kamerového systému

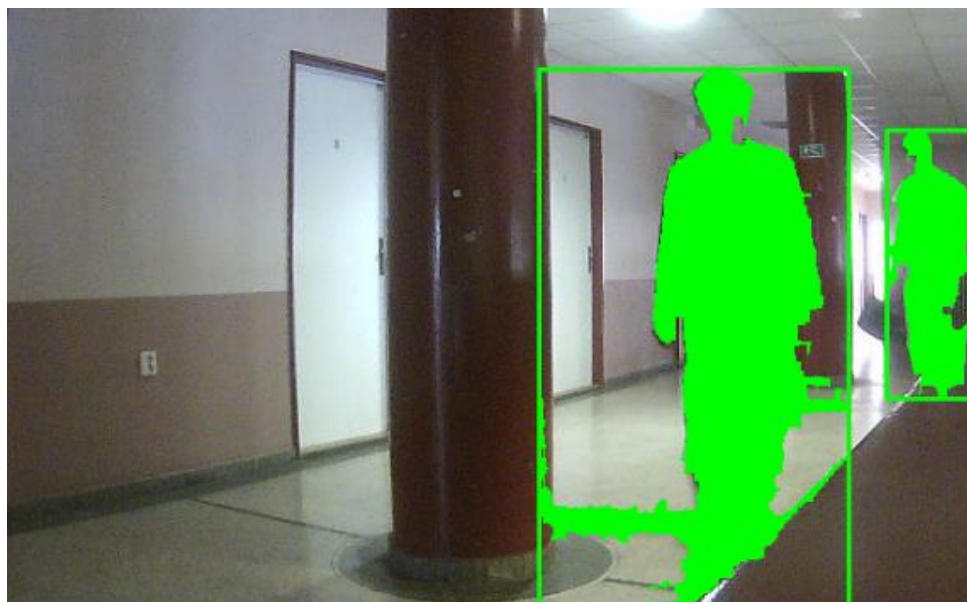
Předposlední video modeluje situaci průchodu osoby vstupními dveřmi. Zde si můžeme povšimnout reakce systému na pohyb dveří, které jsou detekovány společně s procházející osobou. Pro odstranění tohoto problému by bylo nutné použít sofistikovanější algoritmy.



Obr. 45 Třetí video aplikace kamerového systému



Poslední video je natočeno na chodbě umístěné za vstupními dveřmi. Opět modeluje situaci se dvěma pohybujícími se objekty. Video bylo vytvořeno pro demonstraci nechtěných stínů v případě nevhodného osvětlení místnosti. Jednotlivé objekty vrhají velké stíny, které není možné použitým algoritmem výrazně omezit.



Obr. 46 Čtvrté video aplikace kamerového systému

Z jednotlivých příkladů je zjevné, že v případě praktického použití je nutné upravit sledovací algoritmus vždy pro konkrétní situace. Různé vzdálenosti odpovídají různým velikostem objektů, které je nutné filtrovat. Přesto i v případě použití relativně jednoduchých algoritmů můžeme dosáhnout dobrých výsledků.

## 6. Závěr

Cílem této diplomové práce bylo vytvořit aplikaci s grafickým prostředím pro práci s kamerovým systémem tvořeným kamerami VIVOTEK PT8133W. V teoretické části jsou shrnuty poznatky o kamerových systémech, jejich rozdělení a využití pro bezpečnostní sektor. Dále je zde popsána knihovna OpenCV a její jednotlivé moduly. V praktické části jsme se seznámili s použitými kamerami, dodaným softwarem a ovládáním kamer pomocí URL příkazů. Navržený systém byl propojen s počítačem a spojen s vytvořenou aplikací.

V rámci praktické části se podařilo vytvořit funkční aplikaci pro navázání komunikace a přenos obrazu současně ze čtyř kamer. Použití více vláknového programování umožňuje aplikaci kromě snímání obrazu, implementovat téměř jakýkoliv algoritmus vhodný pro sledování objektů či osob. Tento bod byl pro práci stěžejní a vzhledem k bezproblémovému chodu aplikace, bude její vývoj pokračovat.

Součástí programu jsou podpůrné nástroje pro snadné připojení kamery. Díky efektivnímu prohledávání adresového prostoru sítě, je aplikace schopna bez zpomalení programu zjistit a zobrazit připojená zařízení.

Pro otestování funkčnosti aplikace byl vytvořen algoritmus pro detekci, sledování a rozpoznání lidské postavy. Součástí je i uživatelská definice perimetrů s možností uložení informací do souboru a následné nahrání zpět do aplikace. Ideální spojení představuje použití definovaných perimetrů společně s detekovaným pohybem. Vstup osoby do kritické zóny je možné spojit se spuštěním alarmu či odesláním emailové zprávy společně s fotkou narušitele na danou adresu.

V rámci diplomové práce bylo zjištěno slabé místo v zabezpečení kamerového systému na Sinkuleho koleji. Kolejní síť je připojena ke kamerovému systému v posilovně, jehož webové rozhraní sice bylo zabezpečeno heslem, ovšem RTSP kanál, který se také využívá k přenosu dat, byl nezabezpečeny. Díky tomu kdokoliv znalý mohl sledovat jednotlivé záběry z kamer, což vyústilo v pozastavení nahrávání na síťové médium. Správce systému byl informován a kamerový systém zabezpečen.

Dále byla zjištěna chyba v knihovně OpenCV, kdy jednotlivé funkce pro správu síťových kamer způsobí zastavení čtecího vlákna, dokud není kamera opět připojena. V takovém případě je nutno vlákno násilně ukončit, což je v programování nepřípustné. Chyba byla nahlášena komunitě spravující OpenCV a v příštích verzích bude pravděpodobně opravena.

## Zdroje:

1. Closed-circuit television. *Wikipedie*. [Online] Wikimedia Foundation, 2015. [Citace: 24. 4 2015.] [http://en.wikipedia.org/wiki/Closed-circuit\\_television](http://en.wikipedia.org/wiki/Closed-circuit_television).
2. Michaela, Tokárová. *Kamerové systémy na veřejných prostranstvích – zajištění bezpečnosti nebo ohrožení soukromí?* Praha: UK 2014 : Diplomová práce, Univerzita Karlova v Praze, Právnická fakulta, Katedra ústavního práva.
3. \*NEW\* SECURITY CAMERA MONITORING SERVICES. *havenprotection*. [Online] HAVEN PROTECTION, 2015. [Citace: 24. 4 2015.] <http://havenprotection.com/uncategorized/security-camera-monitoring-services-8/>.
4. Kamerové systémy. *Ladinn*. [Online] ELKOV elektro a.s., 24. 5 2014. [Citace: 24. 4 2015.] [http://www.ladinn.cz/ostatni/technika/kamerovy\\_system.html](http://www.ladinn.cz/ostatni/technika/kamerovy_system.html).
5. Evolution of video surveillance systems. *Axis*. [Online] Axis Communications, 2015. [Citace: 24. 4 2015.] [http://classic.www.axis.com/products/video/about\\_networkvideo/evolution.htm#analog](http://classic.www.axis.com/products/video/about_networkvideo/evolution.htm#analog).
6. Jaromír, Hoško. *Bezpečnostní kamerové systémy*. Praha ČVUT 2014 : Diplomová práce, ČVUT, FEL, Katedra radioelektroniky.
7. Jiří, Kocfelda. *Kamerové systémy pro zabezpečení veřejných*. Zlín: UTB 2009 : Bakalářská práce, UTB ve Zlíně, Fakulta aplikované informatiky.
8. IP Camera. *systemofall*. [Online] DOITMAX, 2011. [Citace: 24. 5 2015.] [http://systemofall.wdsinternational.com/?page\\_id=128](http://systemofall.wdsinternational.com/?page_id=128).
9. Configure your X4S 1U NVR DVR. *ascendentgroup*. [Online] ascendentgroup, 2015. [Citace: 24. 5 2015.] <http://www.ascendentgroup.com/store/configure/X4S-NVR-1U/229>.
10. dvr. *cosmicitsolution*. [Online] Cosmic IT Solutions. [Citace: 24. 5 2015.] <http://www.cosmicitsolution.com/category/dvr/>.
11. Technologie LCD panelů. *svethardware*. [Online] oXy Online s.r.o., 8. 2 2013. [Citace: 29. 4 2015.] <http://www.svethardware.cz/technologie-lcd-panelu/14465-2>.
12. Technologie: Plazma displeje. *zcu*. [Online] paitlova. [Citace: 29. 4 2015.] <http://home.zcu.cz/~paitlova/druha.Html>.
13. Samsung's 55-inch Curved OLED TV. *dailytech*. [Online] DailyTec, 2015. [Citace: 24. 5 2015.] <http://www.dailytech.com/Samsungs+55inch+Curved+OLED+TV+Ships+for+Only+9000/article33166.htm>.
14. OLED displeje: technologie současnosti i budoucnosti. *avmania*. [Online] Mladá fronta a. s., 2014. [Citace: 29. 4 2015.] <http://avmania.e15.cz/oled-displeje-technologie-soucasnosti-i-budoucnosti/ch-4940#articleStart>.
15. prod04. *bigcity-web*. [Online] EJABER, 2003. [Citace: 24. 5 2015.] <http://www.bigcity-web.com/prod04.htm>.
16. C2N-CAMIDJ. *crestron*. [Online] Crestron Electronics, Inc., 2015. [Citace: 24. 5 2015.] [http://www.crestron.com/resources/product\\_and\\_programming\\_resources/catalogs\\_and\\_brochures/online\\_catalog/default.asp?jump=1&model=C2N-CAMIDJ](http://www.crestron.com/resources/product_and_programming_resources/catalogs_and_brochures/online_catalog/default.asp?jump=1&model=C2N-CAMIDJ).



17. Visonic MCT-220. *sourcesecurity*. [Online] Notting Hill Media, 2015. [Citace: 24. 5 2015.] <http://www.sourcesecurity.com/technical-details/intruder-alarms/control-panels-systems-and-accessories.1/control-panels-systems-and-accessories/visonic-mct-220-emergency-button.html>.
18. Prymus, Jiří. *Rozšíření knihovny pro zpracování obrazu*. Brno: VUT 2010 : Bakalářská práce, VUT Brno, Fakulta elektrotechniky, Ústav automatizace a měřicí techniky.
19. Aghajan H., Cavallaro A. *Multi-Camera Networks: Principles and Applications*. místo neznámé : Academic Press. ISBN 978-0123746337.
20. D, Song. *Sharing a Vision*. místo neznámé : Springer, 2009. ISBN 978-3-540-88064-6.
21. Samsung 2233RZ 22" LCD Monitor. *techshop*. [Online] TechShop, 2015. [Citace: 24. 5 2015.] [http://techshop.in/store/product\\_info.php?products\\_id=4287](http://techshop.in/store/product_info.php?products_id=4287).

## Seznam obrázků:

Obr. 1 Ústředna bezpečnostního kamerového systému (3) .....	2
Obr. 2 Schéma obecného kamerového systému (4) .....	3
Obr. 3 Schéma analogového kamerového systému (5) .....	3
Obr. 4 Schéma částečně analogového kamerového systému (5).....	4
Obr. 5 Příklad IP kamery s ochranným krytem (8) .....	5
Obr. 6 NVR záznamové zařízení (9) .....	6
Obr. 7 DVR záznamové zařízení (10).....	6
Obr. 8 LCD monitor (11) .....	7
Obr. 9 OLED displej (13) .....	8
Obr. 10 Ochranný kryt pro kamery (15) .....	9
Obr. 11 Ovládací panel pro kamerové systémy (16) .....	9
Obr. 12 Poplachové tlačítko (17).....	9
Obr. 13 Uspořádání knihovny OpenCV (18) .....	10
Obr. 14 Příklad využití funkce gaussianBlur().....	11
Obr. 15 Příklad využití funkce findStereoCorrespondence() a sub().....	11
Obr. 16 Příklad jednoduchého GUI v OpenCV.....	12
Obr. 17 Kamera Vivotek PT8133W (6) .....	14
Obr. 18 Tabulka technických parametrů kamery PT8133W4.2 Propojení systému s PC (6) .....	14
Obr. 19 Schéma připojení kamerového systému k počítači.....	15
Obr. 20 Installation Wizard 2 - prohlédávání sítě .....	15
Obr. 21 Hlavní menu aplikace Installation Wizard 2 .....	16
Obr. 22 Webové prostředí IP kamery.....	16
Obr. 23 Nastavení výstupního videa z kamery.....	17
Obr. 24 Hlavní menu aplikace inteligentního kamerového systému .....	20

Obr. 25 Box s jednotlivými funkcemi programu .....	21
Obr. 26 Příklad neúspěšného a úspěšného připojení kamery .....	21
Obr. 27 Příklad násilného odpojení kamery .....	22
Obr. 28 E-mail oznamující přerušení spojení s kamerou .....	22
Obr. 29 Okno Properties .....	23
Obr. 30 Okno pro definici pritických zón.....	25
Obr. 31 Chyba při špatné definici.....	25
Obr. 32 Ukázkové situace sledovacího systému .....	26
Obr. 33 E-mailová zpráva informující o narušení perimetru.....	27
Obr. 34 Příklad vyhledání aktivních bran počítače.....	28
Obr. 35 Příklad vyhledání jednotlivých IP adres.....	28
Obr. 36 Vlevo obrázek pozadí a vpravo aktuální snímek .....	29
Obr. 37 Příklad prahování a použití morfologických operací.....	30
Obr. 38 Vykreslení objektu nalezeného pomocí funkce findContours().....	31
Obr. 39 Ukázka výpočtu geometrických středů jednotlivých objektů .....	32
Obr. 40 Zleva příklad použití klasifikátoru lbpcascades a zprava haaruv kaskádní klasifikátor .....	33
Obr. 41 Příklad úspěšné detekce lidské postavy pomocí histogramu orientovaných gradientů.....	34
Obr. 42 Blokový diagram hlavních funkcí.....	35
Obr. 43 První video aplikace kamerového systému .....	36
Obr. 44 Druhé video aplikace kamerového systému .....	37
Obr. 45 Třetí video aplikace kamerového systému.....	37
Obr. 46 Čtvrté video aplikace kamerového systému .....	38

## **Seznam zkratk:**

CCTV- Closed Circuit Television  
 CRT- Cathode Ray Tube  
 LCD- Liquid Crystal Display  
 PTZ- Pan Tilt Zoom  
 IP- Internet Protocol  
 VCR- VideoCassette Recorder  
 NVR- Network Video Recorder  
 DVR- Digital Video Recorder  
 BNC- Bayonet Neill Concelman  
 OLED- Organic Light Emitting Diode  
 MLL- Machine Learning Library  
 Wi-Fi- Wide Fidelity  
 URL- Uniform Resource Locator  
 RTSP- Real Time Streaming Protocol

## **Seznam příloh:**

- a) CD obsahující: diplomovou práci (pdf)  
zdrojové kódy programu  
videa: 1.avi  
2.avi  
3.avi  
4.avi