

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ  
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF TELECOMMUNICATIONS

DETEKCE ČÁSTÍ OBLIČEJE V TERMOGRAFICKÉM  
SPEKTRU

DIPLOMOVÁ PRÁCE  
MASTER'S THESIS

AUTOR PRÁCE  
AUTHOR

Bc. MIROSLAV ŠUJAN



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH  
TECHNOLOGIÍ**

**ÚSTAV TELEKOMUNIKACÍ**

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF TELECOMMUNICATIONS

# DETEKCE ČÁSTÍ OBLIČEJE V TERMOGRAFICKÉM SPEKTRU

DETECTION OF FACE PARTS IN THE THERMOGRAPHIC SPECTRUM

**DIPLOMOVÁ PRÁCE**

MASTER'S THESIS

**AUTOR PRÁCE**

AUTHOR

**Bc. MIROSLAV ŠUJAN**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**Ing. JIŘÍ MEKYSKA**

BRNO 2011



VYSOKÉ UČENÍ  
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

Ústav telekomunikací

# Diplomová práce

magisterský navazující studijní obor  
Telekomunikační a informační technika

**Student:** Bc. Miroslav Šujan

**ID:** 72168

**Ročník:** 2

**Akademický rok:** 2010/2011

## NÁZEV TÉMATU:

**Detekce částí obličeje v termografickém spektru**

## POKYNY PRO VYPRACOVÁNÍ:

Cílem diplomové práce je rozbor aktuální problematiky detekce částí obličeje v termografickém spektru a následný návrh a implementace systému, který by prováděl detekci očí, úst a nosu. Systém by byl založen na algoritmu Viola-Jones, algoritmu vertikální a horizontální projekce a na modelech ASM/AAM.

## DOPORUČENÁ LITERATURA:

- [1] MADISETTI, Vijay; BENNETT WILLIAMS, Douglas. The digital signal processing handbook. USA: CRC Press, 1998. 1776 s. ISBN 0-8493-8572-5.
- [2] C. GONZALEZ, Rafael; E. WOODS, Richard; L. EDDINS, Steven. Digital Image Processing Using MATLAB. 2nd ed. Gatesmark Publishing, 2009. 827 s. ISBN 978-0982085400.
- [3] HAMZA, M. H. Visualization, Imaging, and Image Processing. Anaheim: ACTA Press, 2003. 1059 s. ISBN 0-88986-382-2.

**Termín zadání:** 7.2.2011

**Termín odevzdání:** 26.5.2011

**Vedoucí práce:** Ing. Jiří Mekyska

**prof. Ing. Kamil Vrba, CSc.**

*Předseda oborové rady*

## UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

## **ABSTRAKT**

Diplomová práce se zabývá aktuální problematikou detekce obličeje a jeho částí v termografickém spektru. Většina doposud publikované literatury se zabývá problematikou detekce ve viditelném spektru, což z detekce v termografickém spektru dělá zajímavou alternativu. Práce pojednává o zpracování obrazových signálů, obrazech a obličejích v termografickém spektru, vybraných metodách detekce obličeje a jeho částí a dále se zabývá praktickým návrhem systému pro detekci částí obličeje v tomto spektru a jeho následným testováním.

## **KLÍČOVÁ SLOVA**

AAM, AdaBoost, ASM, barevný model, detekce obličeje, histogram, integrální obraz, obličej, Otsuho práhovací metoda, segmentace, spektrum, termografie, VH projekce, Viola-Jones

## **ABSTRACT**

Master's thesis deals with current problems of face detection and its parts in the infrared thermographic spectrum. Most previously published literature deals with the detection in the visible spectrum, making the thermographic detection range an interesting alternative. The work deals with the processing of image signals, images and faces in thermographic spectrum, selected methods of face detection and its parts and also deals with practical system design for detecting facial parts in this spectrum and its subsequent testing.

## **KEYWORDS**

AAM, AdaBoost, ASM, color space, face detection, histogram, integral image, face, Otsu thresholding, segmentation, spectrum, thermography, VH projection, Viola-Jones

ŠUJAN, Miroslav *Detekce částí obličeje v termografickém spektru: diplomová práce.* Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2011. 66 s. Vedoucí práce byl Ing. Jiří Mekyska

## PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Detekce částí obličeje v termografickém spektru“ jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

Brno .....

.....

(podpis autora)

## PODĚKOVÁNÍ

Děkuji vedoucímu diplomové práce Ing. Jiřímu Mekyskovi za velmi užitečnou metodickou pomoc, cenné rady a vstřícné jednání při zpracování práce. Dále děkuji svým rodičům za to, že mě podporovali ve studiu a byli mi po celou dobu oporou. Také děkuji své přítelkyni za trpělivost a podporu při zpracování práce.

V Brně dne .....

.....

(podpis autora)

# OBSAH

Úvod	11
<b>1 Zpracování obrazových signálů</b>	<b>13</b>
1.1 Obraz a jeho reprezentace	13
1.2 Barevné modely	13
1.2.1 RGB model	14
1.2.2 CMY a CMYK modely	14
1.2.3 YUV model	14
1.2.4 YCbCr model	15
1.2.5 HSV a HSB modely	15
1.2.6 HLS model	15
1.3 Histogram	15
<b>2 Obrazy v termografickém spektru</b>	<b>17</b>
2.1 Obličej v termografickém spektru	18
<b>3 Detekce obličeje a jeho částí</b>	<b>21</b>
3.1 Detektor Viola-Jones	21
3.1.1 AdaBoost algoritmus	22
3.1.2 Integrální obraz	23
3.1.3 Příznaky podobné Haarově vlnce	23
3.1.4 Kaskádní kombinace klasifikátorů	24
3.2 VH projekce	26
3.3 Segmentace	28
3.3.1 Otsuho práhovací metoda	28
3.4 Aktivní modely tvaru (ASM) a vzhledu (AAM)	30
3.4.1 Analýza hlavních komponent (PCA)	32
<b>4 Praktická realizace metody pro detekci částí obličeje</b>	<b>33</b>
4.1 Fáze trénování	35
4.1.1 OpenCV	36
4.1.2 Trénování detektoru Viola-Jones	36
4.1.3 Active Shape Model (ASM) and Active Appearance Model (AAM) toolbox	39
4.1.4 Trénování ASM a AAM	40
4.2 Implementace detektoru	44
4.3 Detekce obličeje	44
4.3.1 Implementace detektoru Viola-Jones	44

4.3.2	Implementace detektoru VH projekce . . . . .	45
4.4	Detekce částí obličeje . . . . .	46
4.4.1	Implementace techniky ASM . . . . .	46
4.4.2	Implementace techniky AAM . . . . .	48
4.5	Popis ovládání detekčního systému . . . . .	49
4.5.1	Funkce . . . . .	49
4.5.2	GUI . . . . .	50
<b>5</b>	<b>Testování</b>	<b>52</b>
5.1	Dosažené výsledky pro modely natrénované z 25 obrázků . . . . .	54
5.2	Dosažené výsledky pro modely natrénované z 50 obrázků . . . . .	55
5.3	Dosažené výsledky pro modely natrénované ze 75 obrázků . . . . .	55
5.4	Vylepšený model . . . . .	56
<b>6</b>	<b>Závěr</b>	<b>58</b>
	<b>Literatura</b>	<b>60</b>
	<b>Seznam symbolů, veličin a zkratk</b>	<b>63</b>
<b>A</b>	<b>Obsah příloženého DVD</b>	<b>66</b>



# SEZNAM OBRÁZKŮ

1.1	Ukázka šedotónového obrazu a jeho histogramu. . . . .	16
2.1	Elektromagnetické spektrum [11]. . . . .	18
2.2	Ukázka obličejů v termografickém spektru. . . . .	20
3.1	Základní požadavky na systémy detekce. . . . .	21
3.2	$h_1$ ani $h_2$ nejsou dobré klasifikátory, ale AdaBoost je zkombinuje a tím dosáhne jednoho dobrého klasifikátoru [9]. . . . .	22
3.3	Hodnota integrálního obrazu v bodě o souřadnicích $(x, y)$ je dána součtem všech pixelů nad ním vlevo [28]. . . . .	23
3.4	Součet pixelů v obdélníku D lze vypočítat jen se 4 přístupy do paměti. Hodnota integrálního obrazu na pozici 1 je suma pixelů obdélníku A. Hodnota na pozici 2 je suma obdélníků $A+B$ , na pozici 3 suma $A+C$ a na pozici 4 suma $A+B+C+D$ . Součet pixelů v obdélníku D se spočítá jako $4+1-(2+3)$ [28]. . . . .	24
3.5	Haarovy příznaky [28]. . . . .	24
3.6	Kaskádní kombinace klasifikátorů [28]. . . . .	25
3.7	Schematické znázornění funkce detektoru VH projekce [10]. . . . .	27
3.8	Dva clustery, jejichž součet tvoří celkový průběh histogramu. . . . .	29
3.9	Ukázka rozložení bodů na významných částech obličeje. . . . .	31
4.1	Navržené schéma metody detekce významných částí obličeje. . . . .	33
4.2	Ukázka z použité databáze termografických snímků. . . . .	35
4.3	Ukázka obličejů z databáze termografických snímků, získaných metodou VH projekce. . . . .	37
4.4	Ukázka těch samých upravených obličejů pomocí utility <i>createsamples</i> na rozlišení $20 \times 20$ px. . . . .	38
4.5	Ukázka rozložení bodů pro techniky ASM i AAM. . . . .	41
4.6	Ukázka prvního vytvořeného modelu, který sestával z 33 bodů. . . . .	42
4.7	Ukázka detekcí pomocí detektoru Viola-Jones. . . . .	45
4.8	Ukázka detekcí pomocí detektoru VH projekce. . . . .	45
4.9	Ukázka detekovaných částí obličeje technikou ASM. . . . .	47
4.10	Ukázka detekovaných částí obličeje technikou AAM. . . . .	48
4.11	Ukázka grafického uživatelského prostředí. . . . .	51
5.1	Ukázka správné a špatné detekce obličeje. . . . .	52
5.2	Ukázka správně a špatně detekovaných částí obličeje. . . . .	53
5.3	Porovnání úspěšnosti metod detekce obličeje. . . . .	53
5.4	Porovnání úspěšnosti metod detekce částí obličeje pro 25 trénovacích obrázků. . . . .	54

5.5	Porovnání úspěšnosti metod detekce částí obličeje pro 50 trénovacích obrázků. . . . .	55
5.6	Porovnání úspěšnosti metod detekce částí obličeje pro 75 trénovacích obrázků. . . . .	56
5.7	Úspěšnost detekce vylepšeného modelu. . . . .	57

## SEZNAM TABULEK

5.1	Porovnání úspěšnosti detekce technik VH projekce a Viola-Jones. . .	53
5.2	Výsledky pro různé kombinace nastavení detekčního systému pro 25 trénovacích obrázků. . . . .	54
5.3	Výsledky pro různé kombinace nastavení detekčního systému 50 tré- novacích obrázků. . . . .	55
5.4	Výsledky pro různé kombinace nastavení detekčního systému 75 tré- novacích obrázků. . . . .	56
5.5	Výsledky pro vylepšený model. . . . .	57

# ÚVOD

S rozvojem nových technologií výroby termografických kamer a jejich neustále se snižující cenou začíná být velmi zajímavé a atraktivní jejich použití i pro detekci obličeje a jeho částí (jako jsou ústa, nos, oči, atd.). Detekce těchto významných bodů obrazu nachází uplatnění například při biometrické identifikaci nebo rozpoznání emočních stavů osob a to i při špatném nebo dokonce žádném osvětlení.

Cílem diplomové práce je rozbor aktuální problematiky detekce částí obličeje v termografickém spektru a následný návrh a implementace systému, který provádí detekci očí a nosu. Systém je založen na algoritmu Viola–Jones a na algoritmu vertikální a horizontální projekce, tyto algoritmy slouží k detekci obličeje. Dále je systém založen na metodách ASM/AAM, které v našem případě slouží k detekci částí obličeje. Celý systém je implementován a navržen pomocí programu Matlab, dílčí části potom pomocí programovacího prostředí C/C++ v kombinaci s knihovnou OpenCV.

Diplomová práce je přehledně rozdělena na pět kapitol, ve kterých je postupně představována problematika detekce obličeje a jeho částí v termografickém spektru a další problémy s tím související. Poslední kapitolou je závěr, kde je uvedeno shrnutí dosažených výsledků.

První kapitola je věnována zpracování obrazových signálů, zejména reprezentaci obrazu, barevným modelům (RGB, CMY, CMYK, YUV, YCbCr, HSV, HSB, HLS) a také histogramům.

Druhá kapitola se zabývá obrazy v termografickém spektru, zejména pak vlnovými délkami, které reprezentují infračervené záření, vlastnostmi tohoto záření a jeho využitím. Dále jsou v této kapitole uvedeny charakteristické vlastnosti obličejů v termografickém spektru.

Třetí kapitola pojednává o metodách detekce obličeje a jeho částí. Jsou zde uvedeny detektory obličeje Viola-Jones a VH projekce, dále metody Active Shape Models a Active Appearance Models, jež slouží k detekci kontury jakéhokoliv vzoru, v našem případě jsou ovšem použity k detekci částí obličeje. Jsou zde také uvedeny výhody a nevýhody detektorů obličeje a metod detekce částí obličeje. Tato kapitola se také věnuje segmentaci obrazu, zejména potom Otsuho práhovací metodě.

Čtvrtá kapitola je věnována praktické realizaci metody pro detekci částí obličeje. Je zde uvedeno, jakým způsobem probíhalo trénování detektoru Viola-Jones a trénování metod Active Shape Models i Active Appearance Models na použité databázi obrazů pořízených v termografickém spektru. Je zde také představeno, jakým způsobem probíhala implementace detekčního systému v programu Matlab a jeho ovládání.

Pátá kapitola se zabývá testováním navrženého detektoru a porovnáním výsledků pro různá nastavení a kombinace metod detekce obličeje a částí obličeje.

Poslední kapitolou je závěr, kde jsou zhodnoceny a porovnány dosažené výsledky. Také jsou zde uvedena možná řešení nedostatků detekčního systému, která by mohla být výchozím bodem pro další rozšiřování práce.

# 1 ZPRACOVÁNÍ OBRAZOVÝCH SIGNÁLŮ

## 1.1 Obraz a jeho reprezentace

Obrazy lze v současné době dělit do dvou základních skupin, a to na obrazy analogové a digitální. Analogový obraz  $\mathbf{A}(x, y)$  je ve 2D prostoru spojitou funkcí dvou proměnných  $x$  a  $y$ . Digitální obraz  $\mathbf{A}[m, n]$  popsáný ve 2D diskretním prostoru je odvozen z analogového obrazu vzorkovacím procesem, který je také často nazýván digitalizace. 2D spojitý obraz  $\mathbf{A}(x, y)$  je rozdělen na  $M$  řádků a  $N$  sloupců. Průsečík řádku a sloupce se nazývá pixel. Hodnota příslušející celočíselným souřadnicím  $[m, n]$ , kde  $m = 0, 1, 2, \dots, M-1$  a  $n = 0, 1, 2, \dots, N-1$  je  $\mathbf{A}[m, n]$ . Ve většině případů se ovšem jedná o vícerozměrné signály, které ještě nesou např. informace o hloubce ( $z$ ), barvě ( $\beta$ ) nebo času ( $t$ ) [19].

Jak již bylo řečeno, proces, kterým lze přejít z 2D spojitého prostoru do 2D diskretního prostoru se nazývá digitalizace. Tento proces má dva nezávislé kroky a to vzorkování a kvantování.

Vzorkování je proces, při němž se v pravidelných a přesně definovaných intervalech zaznamenávají hodnoty odpovídající analogovému ekvivalentu. Za předpokladu dodržení vzorkovacího teoremu nedochází ke zkreslení a znehodnocení takto získaných dat.

Kvantování je proces, při němž je každé hodnotě získané vzorkováním, přidělena právě jedna hodnota z určitého rozsahu, který je k dispozici. Tímto procesem dochází k určité ztrátě informace [14].

## 1.2 Barevné modely

Jestliže bychom chtěli věrně zaznamenat barvu nějakého objektu, potom bychom v každém bodě jeho povrchu museli zaznamenat spektrální křivku, proto se používají barevné modely, které slouží pro zjednodušení záznamu barevné informace. Ten používá základní barvy, jejichž mísením vzniká výsledná požadovaná barva. Při volbě vhodného barevného prostoru je nutné brát v potaz vhodnost tohoto prostoru z hlediska podání barevného dojmu a náročnosti použití, musí tedy být zajištěn jakýsi kompromis [24].

Existují dva základní způsoby míchání barev:

- Substraktivní (odečítací) míchání – pracuje na principu pohlcování bílého světla postupným přidáváním jednotlivých barevných vrstev, výsledná barva je tedy tvořena zbylými vlnovými délkami; základní barvy jsou purpurová,

azurová, žlutá (smícháním všech vznikne černá); tento způsob využívají tiskárny [24].

- Aditivní (sčítací) míchání – výsledkem sčítání barev je světlo větší intenzity; základní barvy jsou červená, zelená, modrá; tento způsob využívají displeje a monitory [24].

### 1.2.1 RGB model

Tento model vychází z faktu, že lidské oko je nejcitlivější na tři základní barvy, kterými jsou červená (vlnová délka 630 nm), zelená (530 nm) a modrá (450 nm). Jedná se o aditivní model, který lze vyjádřit pomocí jednotkové krychle, kde je v počátku  $[0,0,0]$  černá barva a v protilehlém vrcholu  $[1,1,1]$  bílá barva. Navzájem komplementární barvy leží v protilehlých vrcholech krychle a jejich mísením lze získat barvu bílou. Barevné odstíny vznikají skládáním základních barev, jejichž intenzita se udává v intervalu  $\langle 0, 1 \rangle$ . Podílovým mísením těchto barev lze získat téměř všechny složky barevného spektra.

Variantou tohoto modelu je model RGBA, kdy A označuje tzv. alfa kanál definující míru průhlednosti, což je poměr v jakém se smíchá daná barva s případnou barvou na pozadí. Zcela neprůhledná barva má hodnotu alfa kanálu rovnu jedné, zcela průhledná barva má naopak hodnotu alfa kanálu rovnu nule [8].

### 1.2.2 CMY a CMYK modely

Barevný model CMY lépe odpovídá lidské zkušenosti s mícháním barev. Tento model je používán především v tiskařské technice, protože je typický pro míchání tiskařských barev. Problémem tohoto modelu je nedokonalé krytí jednotlivých barevných složek, smícháním všech barev nevznikne dokonale černá barva, ale jakási směs tmavě hnědé a černé. Proto se v praxi používá model CMYK, který definuje ještě takzvaný kontrast, ten odpovídá černé barvě a používá se také ke ztmavení ostatních barevných odstínů [8].

### 1.2.3 YUV model

Tento model odděluje jasovou složku od složek barevných, čehož se využívá hlavně v televizní technice, norma PAL. Oddělením jasové složky Y (iluminance) a barevných složek (U,V – chrominance) se docílí možnosti použití stejného signálu pro barevné i černobílé televizory zároveň [24].

### 1.2.4 YCbCr model

Tento model podobně jako model YUV také odděluje jasovou složku od složek barevných, využívá se při zpracování digitálního obrazu a videa, norma SECAM. Formát JPEG také s výhodou využívá tento model, protože při oddělení jednotlivých složek se dají snáze implementovat kompresní algoritmy. Y – jasová složka, Cb – modrá složka, Cr – červená složka [24].

### 1.2.5 HSV a HSB modely

Tyto modely nejlépe odpovídají lidskému intuitivnímu popisu barev, mají tři základní parametry – tón, sytost a jas. Pro zobrazení tohoto modelu se používá šestiboký jehlan popř. kužel. Při tvorbě barvy je nejprve zvolen odstín, následně nasycení a nakonec světlost. Tento model má jednu podstatnou nevýhodu a to, že přechod mezi bílou a černou barvou není plynulý [24].

### 1.2.6 HLS model

Nevýhodu při přechodu mezi bílou a černou barvou u modelů HSV a HSB řeší model HLS. Model je tvaru dvojitého rotačního kužele. Sytost leží na vodorovné ose, světlost na svislé ose a barevný tón představuje úhlová hodnota. Tvar modelu odpovídá skutečnosti – schopnost rozlišování barevných odstínů skutečně klesá se ztmavováním a zesvětlováním základní čisté barvy, zvyšování a snižování světlosti barvy skutečně spočívá v přidávání světlého nebo tmavého pigmentu [24].

## 1.3 Histogram

Histogram je grafická reprezentace zobrazující statistické rozložení proměnných dat. Je to odhad pravděpodobnosti výskytu těchto dat, který byl poprvé představen Karlem Pearsonem. Při tvorbě histogramu jsou vstupní data rozdělena podle své hodnoty do buněk definovaných rozsahů. Histogram je tedy potom jistý soubor čísel, která vyjadřují četnosti vstupních dat v jednotlivých buňkách. Celková plocha histogramu je rovna množství dat [24].

Histogram digitálního šedotónového obrazu (viz obr. 1.1) s jasovými úrovněmi v rozsahu  $\langle 0, L - 1 \rangle$  je diskrétní funkcí

$$h(r_k) = n_k, \quad (1.1)$$

kde  $r_k$  je  $k$ -tá úroveň šedi a  $n_k$  je počet pixelů v obraze mající úroveň šedi  $r_k$ . Histogram lze i normalizovat dělením každé jeho hodnoty celkovým počtem pixelů



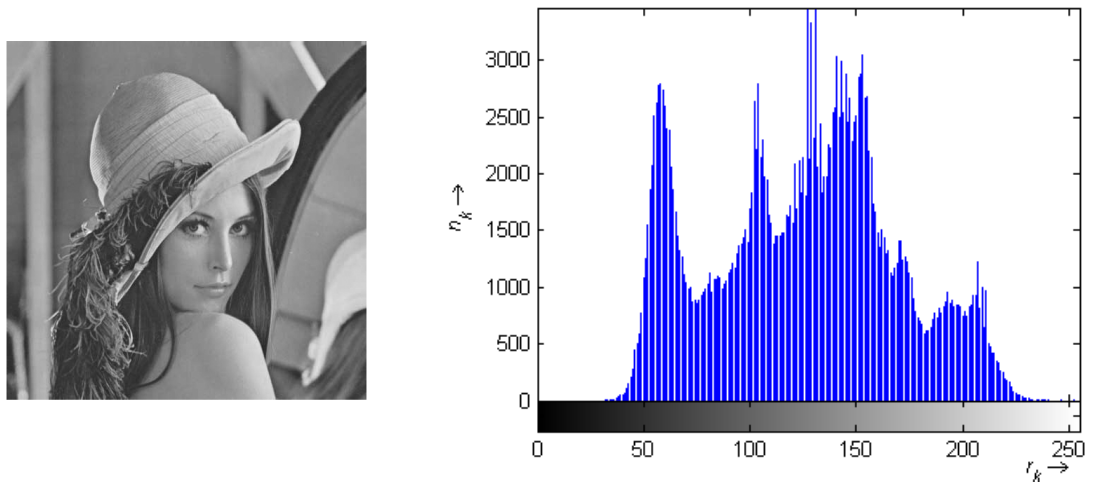
v obraze  $n$ . Pro tento normalizovaný histogram platí

$$p(r_k) = \frac{n_k}{n}, \quad (1.2)$$

pro  $k = 0, 1, \dots, L-1$ .  $p(r_k)$  pak udává pravděpodobnost výskytu úrovně šedi  $r_k$ . Součet všech prvků normalizovaného histogramu je roven 1. Histogram bývá zobrazen pomocí sloupcového diagramu, kde na vodorovné ose jsou vyneseny diskrétní hodnoty jasových úrovní a na ose svislé buď hodnoty  $h(r_k)$ , nebo  $p(r_k)$ . Jedná se tedy o závislost  $n_k$  na  $r_k$ , nebo  $\frac{n_k}{n}$  na  $r_k$ . Z důvodu přehlednosti nebývají osy histogramu popsány [25].

Histogram digitálního barevného obrazu reprezentuje rozložení barev v obraze. U digitálních obrazů reprezentuje barevný histogram počet pixelů určité barvy, které náleží dané buňce jistého barevného rozsahu. Tyto buňky rozdělují celý barevný prostor. Barevný histogram lze vytvořit pro každý barevný prostor, nejvíce se ovšem využívá pro trojdimenzionální prostory jako jsou RGB nebo HSV. Pro vícespektrální obraz, kde je každý pixel reprezentován libovolným počtem měření (například tři měření pro RGB), je histogram  $N$ -rozměrný, kde  $N$  je počet měření. Stejně jako jiné druhy histogramů je i barevný histogram jistou statistikou, která může být zobrazena jako sloupcový graf [22].

Histogramy jsou s výhodou používány pro zpracování obrazu, ať už jsou to metody pro zlepšení kvality obrazu, kompresi či segmentaci. Histogramy lze poměrně snadno vypočítat ale i hardwarově implementovat do zařízení, která v reálném čase zpracovávají obraz (fotoaparáty, kamery).



Obr. 1.1: Ukázka šedotónového obrazu a jeho histogramu.

## 2 OBRAZY V TERMOGRAFIKÉM SPEKTRU

F.W. Herschelův experiment, který vedl k objevu infračerveného záření (IR záření), měl základní význam pro vzestup a vývoj výzkumu v infračerveném spektru. Pomocí tohoto experimentu Herschel pozoroval tepelné efekty, které souvisí s různými spektrálními rozsahy slunečního záření. Umístil černé kontejnery s citlivými rtuťovými teploměry podél skleněného hranolu, kterým rozložil spektrum slunečního záření. Energie dopadajících paprsků byla absorbována kontejnery a teploměry naměřily vyšší teploty než byla teplota okolí. Při analyzování výsledků experimentu Herschel objevil, že údaje teploměrů umístěných za koncem viditelného spektra byly vyšší než ty uprostřed. Experiment prokázal, že spektrum slunečního záření je širší než je viditelný rozsah, a že přímo za červeným koncem existují paprsky okem neviditelné [21].

IR záření je tedy část elektromagnetického spektra vyzařovaného sluncem a chová se podobně jako viditelné záření, které je v rozsahu od 300 nm do 700 nm. IR záření cestuje vesmírem rychlostí světla a může se odrážet, lámat, pohlcovat a vyzařovat. Vlnová délka IR záření je v rozsahu od 700 nm do 1 mm a lze ji rozdělit na určité části. NIR (near infrared) leží hned vedle viditelné části spektra, MIR (medium infrared) a FIR (far infrared) leží daleko za viditelnou částí. Obzvláště zajímavé subpásmo TIR (thermal infrared) části FIR je v rozsahu  $3\ \mu\text{m}$  až  $14\ \mu\text{m}$ . S tímto subpásmem IR spektra se člověk setkává každý den, jedná se totiž o formu tepla neboli tepelné vyzařování. TIR je reprezentováno dvěma tzv. okny nazývanými Mid Wave IR (MWIR) ležícím mezi  $3\ \mu\text{m}$  až  $5\ \mu\text{m}$  a Long Wave IR (LWIR) ležícím mezi  $8\ \mu\text{m}$  a  $14\ \mu\text{m}$ . Mezi nimi se nachází blokované pásmo znečištěné vodními parami. Na obrázku 2.1 je vidět elektromagnetické spektrum, zvláštní pozornost je věnována infračerveným subpásmům [11][32].

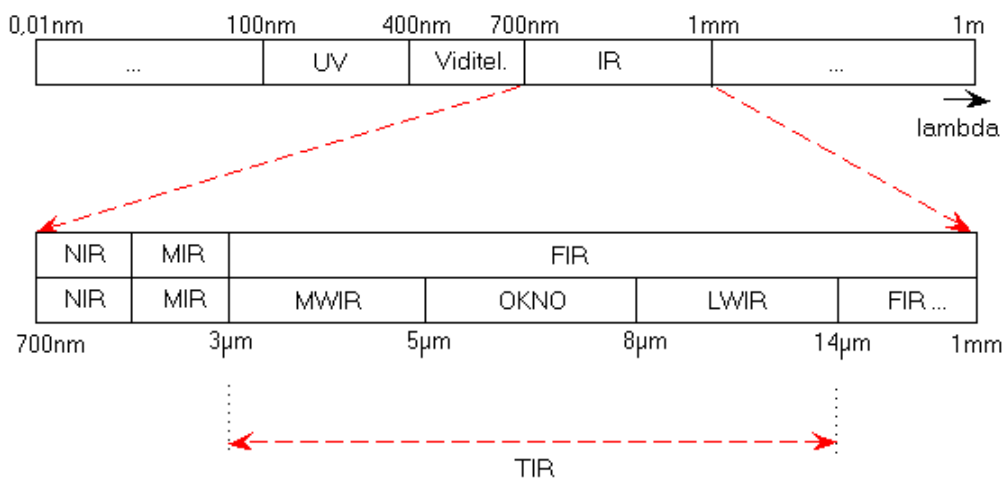
Všechny objekty emitují IR záření, které odpovídá jejich teplotám podle zákona o vyzařování absolutně černého tělesa. IR energie je generována vibrací a rotací atomů a molekul. Čím vyšší je teplota objektu, tím rychlejší je pohyb těchto částic a tím více IR energie je vyzařováno. Při absolutní nule ( $-273\ ^\circ\text{C}$ ) má materiál nejmenší energetický stav a tedy i nejnižší emisi IR záření. Skutečnosti, že všechny objekty emitují IR záření využívají termografické kamery, kterými je možno snímat obrazy v IR spektru i bez viditelného osvětlení. Tyto kamery nesnímají teploty, detekují tepelné vyzařování. Toto vyzařování lze definovat jako IR energii modulovanou atmosférou. Tato energie sestává z emitované, odražené a jiné IR energie. Nutno podotknout, že existují dva druhy termografických kamer, které se liší použitými senzory:

- Senzory bez chlazení – jsou navrženy tak, aby pracovaly jen v LWIR pásmu, ve kterém vyzařuje většina pozemských objektů.

- Sensory s chlazením – jsou chlazeny na nízké teploty, díky čemuž jsou vysoce citlivé i na malé teplotní rozdíly; obecně jsou navrženy tak, aby pracovaly jak v pásmu MWIR, tak i LWIR.

Nejobyčejnější termografické kamery, které se podobají typickým videokamerám, produkují pouze obraz tepelného vyzařování. Sofistikovanější termografické kamery dokáží změřit tepelné vyzařování povrchu jakéhokoliv objektu v obraze, tomuto objektu následně přiřadí barvu podle barevné škály, kde černá barva představuje povrch s nejnižší hodnotou tepelného vyzařování a naopak barva bílá představuje povrch s nejvyšší hodnotou tepelného vyzařování. Takovýto obraz produkovaný termografickou kamerou se nazývá termogram nebo termograf [11][32].

Obrazy v termografickém spektru mají širokou možnost využití a neustále vznikají další možnosti jejich nasazení. Využívají se například v medicíně pro odhalení a diagnostiku nemocí. Termografické kamery byly například nasazovány pro měření teploty osob na letištích, aby se zabránilo šíření viru prasečí chřipky. Další možností využití je zkoumání tepelných úniků izolacemi budov, hledání ohnisek požárů, pátrání po ztracených osobách v neprostupném terénu, zkoumání poruch elektrických vedení, hledání závad na nosných konstrukcích mostů, analýza turbín, atd.



Obr. 2.1: Elektromagnetické spektrum [11].

## 2.1 Obličej v termografickém spektru

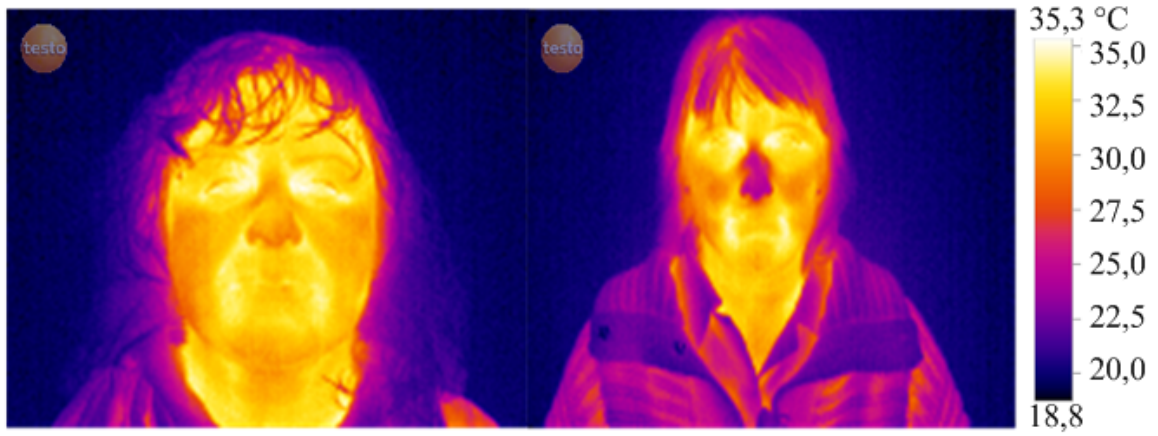
Zvláštní kapitolou termografie jsou obličeje v termografickém spektru. Literatura [10] uvádí, že nacházejí využití v biometrii a to zejména díky následujícím vlastnostem:

- Nejsou ovlivněny osvětlením, stíny, atd. Mohou být dokonce pořízeny v naprosté tmě.
- Je velmi obtížné je napodobit jako ty ve viditelném spektru, protože např. make-up ani různé artefakty nijak neovlivní distribuci krve pod kůží, která je přímo zodpovědná za tepelné vyzařování.
- Mohou rozlišit i jednovaječná dvojčata, protože tepelné vyzařování souvisí s rozložením cév pod kůží, které je u každého člověka jiné, dokonce i u jednovaječných dvojčat.
- Obsahují doplňující informace k obrazům pořízeným ve viditelném spektru, což otevírá široké možnosti využití pro rozpoznávací aplikace nejrůznějších typů.

Tyto vlastnosti platí i obecně, nejen pro obličej. Je pravdou, že tepelné vyzařování bývá blokováno např. vlasy, vousy nebo brýlemi, což je trochu v rozporu s druhým bodem.

Na obrázku 2.2 jsou vidět dva obličejové termogramy pořízené pomocí termografické kamery a následně převedené na obrázky podle stupnice teplot uvedené na pravé straně těchto obrázků. Při zkoumání databáze 41 obrázků (tato databáze byla poskytnuta Ústavem telekomunikací na Fakultě elektrotechniky a komunikačních technologií, VUT Brno) pořízených pomocí termografické kamery byly zjištěny jejich následující vlastnosti:

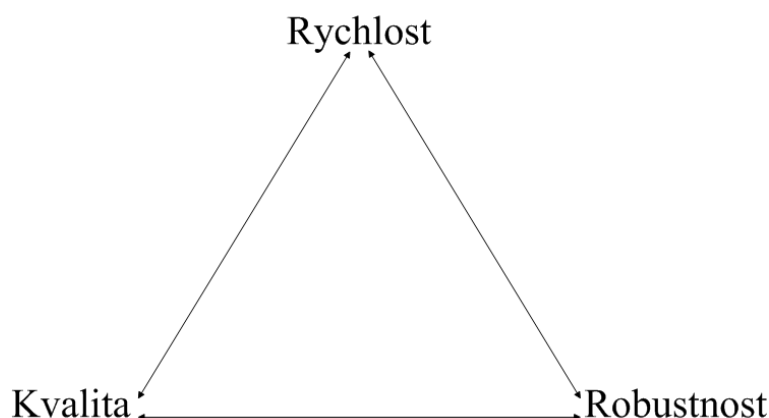
- Nejteplejším místem obličeje jsou koutky očí u kořene nosu (teplota zde dosahuje  $35\text{ }^{\circ}\text{C}$  a více). Toto tvrzení splňuje 40 subjektů.
- Dalším, ale už ne tak výrazným, nejteplejším místem jsou koutky úst, popř. celá ústa (teplota zde dosahuje  $32,5\text{ }^{\circ}\text{C} - 35\text{ }^{\circ}\text{C}$  a více). Toto tvrzení splňuje 33 subjektů.
- Čelo a brada dosahují poměrně stejných teplot ( $30\text{ }^{\circ}\text{C} - 34\text{ }^{\circ}\text{C}$ ). Toto tvrzení platí pro všechny subjekty.
- Líce patří mezi chladnější části obličeje (jejich teplota nepřekračuje  $31\text{ }^{\circ}\text{C}$ ). Toto tvrzení splňuje 33 subjektů.
- Nejchladnější oblastí celého obličeje je nos (průměrná teplota se pohybuje kolem  $25\text{ }^{\circ}\text{C}$ ). Toto tvrzení splňuje 32 subjektů.



Obr. 2.2: Ukázka obličejů v termografickém spektru.

### 3 DETEKCE OBLIČEJE A JEHO ČÁSTÍ

Detekce obličeje a jednotlivých jeho částí v obraze je v současnosti velice populární a často používaná metoda, která nachází své uplatnění například v systémech biometrického zabezpečení nebo jako doplněk digitálních fotoaparátů, které rozpoznají třeba úsměv. Možností využití je opravdu nepřehledné množství. Jedná se o technologii, která dokáže s větší či menší přesností určit, kde se v obraze nachází obličej, a jejímž výstupem jsou souřadnice, které jednoznačně určují pozici tohoto obličeje. V závislosti na typu zpracovávaných dat lze vstupní obraz rozdělit na statický a dynamický (video sekvenci). Aby byly systémy detekce obličeje spolehlivé, je nutné používat robustní, rychlé a v neposlední řadě kvalitní algoritmy. Tyto tři požadavky jsou ovšem do jisté míry protichůdné, a jestliže bude kladen důraz pouze na jeden z nich, potom ty ostatní jdou do ústraní, toto je vyjádřeno obrázkem 3.1. Je tedy potřeba volit jakýsi kompromis mezi těmito základními požadavky.



Obr. 3.1: Základní požadavky na systémy detekce.

Existuje velké množství metod pro detekci obličeje a jeho lokalizaci v obraze. Jak je uvedeno v literatuře [29], lze tyto metody rozdělit do několika skupin. Tato práce se ovšem omezuje pouze na vybrané metody – Viola-Jones a VH projekce.

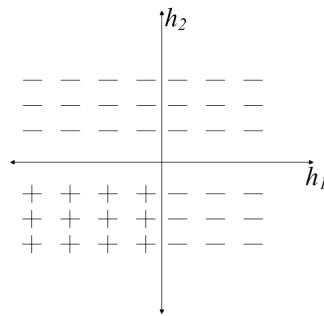
#### 3.1 Detektor Viola-Jones

Tento detektor, poprvé publikovaný v roce 2001, využívá podle literatury [28] ke své činnosti algoritmu strojového učení s učitelem, tzv. AdaBoost, který patří mezi boosting metody. Viola-Jones pracuje s šedotónovými obrazy a skládá se ze tří částí – již zmíněný AdaBoost algoritmus, výpočet integrálního obrazu a extrakce příznaků podobných Haarově vlnce [23]. Mezi výhody tohoto detektoru patří, že jím lze detekovat i více než jen jeden obličej v obraze. Nevýhodou může být složitost

tohoto detektoru a nutnost natrénování na velkém množství obličejů, aby dosahoval uspokojivých výsledků.

### 3.1.1 AdaBoost algoritmus

AdaBoost algoritmus patří mezi tzv. boosting metody, což jsou metody pro dosažení vyšší přesnosti jakéhokoliv algoritmu s učitelem. V podstatě se jedná o kaskádní kombinaci tzv. slabých klasifikátorů, které klasifikují data o něco lépe než náhodně v jediný silný klasifikátor, který klasifikuje data velmi dobře.



Obr. 3.2:  $h_1$  ani  $h_2$  nejsou dobré klasifikátory, ale AdaBoost je zkombinuje a tím dosáhne jednoho dobrého klasifikátoru [9].

Následující text a rovnice převzaty z literatury [9]. Obrázek 3.2 ilustruje, jak AdaBoost kombinuje dva klasifikátory  $h_1$  a  $h_2$ . Na začátku tento algoritmus vybírá klasifikátor, který lépe klasifikuje data. V dalším kroku je datům přiřazena nová váha, která zvyšuje důležitost neklasifikovaných vzorků. Tento proces se opakuje a v každém kroku je váha každého slabého klasifikátoru určena z ostatních slabých klasifikátorů. Algoritmus je tedy následující:

1. Nastav váhy všech vzorků na stejnou hodnotu a najdi  $h_1$ , aby se maximalizovala  $\sum_i y_i h(x_i)$ .
2. Přiřaď novou váhu, aby se zvýšila váha neklasifikovaných vzorků.
3. Najdi další  $h$ , aby se maximalizovala  $\sum_i y_i h(x_i)$ . Urči váhu tohoto klasifikátoru, označ ji  $\alpha$ .
4. Jdi na krok 2.
5. Konečný klasifikátor bude

$$H(x) = \operatorname{sgn}\left(\sum_{i=1}^T \alpha_i h_i(x)\right), \quad (3.1)$$

kde  $\operatorname{sgn}$  je funkce *signum*.

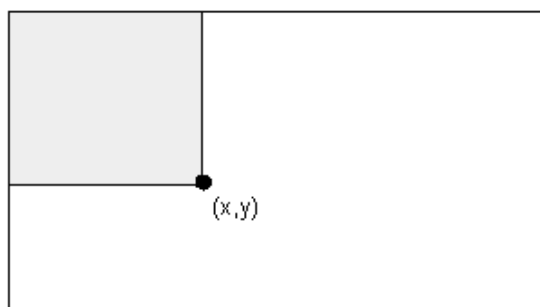
Viola a Jones ovšem zavedli vylepšení, která zvyšují rychlost detekce tohoto algoritmu. Tato vylepšení jsou popsána v následujících podkapitolách.

### 3.1.2 Integrální obraz

Ohraničení části obrazu obdélníkem lze spočítat velmi rychle při použití střední reprezentace obrazu, které nazýváme integrálním obrazem. Integrální obraz na pozici  $x, y$  je tvořen součtem pixelů nahoře vlevo od této pozice, což vyjadřuje rovnice

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y'), \quad (3.2)$$

kde  $ii(x, y)$  je integrální obraz (viz obr. 3.3) a  $i(x', y')$  je originální obraz.



Obr. 3.3: Hodnota integrálního obrazu v bodě o souřadnicích  $(x, y)$  je dána součtem všech pixelů nad ním vlevo [28].

Použitím rekurentních vztahů

$$s(x, y) = s(x, y - 1) + i(x, y), \quad (3.3)$$

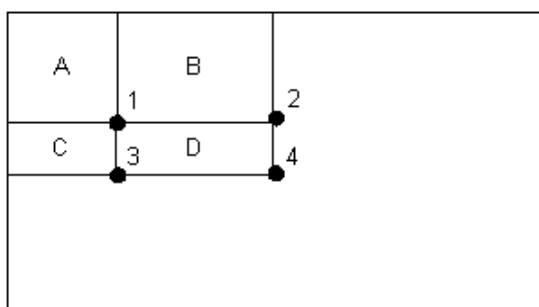
$$ii(x, y) = ii(x - 1, y) + s(x, y), \quad (3.4)$$

kde  $s(x, y)$  je kumulativní řádkový součet,  $s(x, -1) = 0$  a  $ii(-1, y) = 0$ , integrální obraz může tedy být spočítán pouze jedním průchodem originálního obrazu (viz obr. 3.4). Sumu pixelů libovolné obdélníkové oblasti tak lze spočítat v konstantním čase, čehož je využito při výpočtu Haarových příznaků [28].

### 3.1.3 Příznaky podobné Haarově vlnce

Systém Viola-Jones klasifikuje obrazy založené na hodnotě jednoduchých filtrů nazývaných Haarovy příznaky (jedná se o příznaky založené na Haarových vlnkách), které jsou efektivní pro analýzu obličeje. Díky Haarovým příznakům vzniknou primitivní dvourozměrné konvoluční filtry (viz obr. 3.5). Jakmile je vypočítán celý integrální obraz, tak jakýkoliv z těchto rysů může být určen v konstantním čase. Haarovy příznaky jsou užity namísto přímého přístupu k pixelům, protože systém založený na těchto příznacích je daleko rychlejší než ten založený na přístupu k jednotlivým





Obr. 3.4: Součet pixelů v obdélníku D lze vypočítat jen se 4 přístupy do paměti. Hodnota integrálního obrazu na pozici 1 je suma pixelů obdélníku A. Hodnota na pozici 2 je suma obdélníků  $A + B$ , na pozici 3 suma  $A + C$  a na pozici 4 suma  $A + B + C + D$ . Součet pixelů v obdélníku D se spočítá jako  $4 + 1 - (2 + 3)$  [28].

pixelům. Princip jejich použití spočívá ve vyčíslení rozdílu intenzity mezi pixely obrazu v bílé a černé oblasti filtru. Haarovy příznaky poskytují informace o obrazu na abstraktnější úrovni než hodnoty pixelů, což je pro detekční úlohy výhodné [28].



Obr. 3.5: Haarovy příznaky [28].

### 3.1.4 Kaskádní kombinace klasifikátorů

Tato podkapitola popisuje algoritmus pro vytvoření kaskády klasifikátorů (viz obr. 3.6), která zlepšuje detekční schopnosti a zároveň snižuje dobu výpočtu. Kaskáda je navržena tak, že slabší klasifikátory, které jsou ovšem rychlejší, detekují všechna okna s pozadím na základě několika málo příznaků, přičemž nezamítnou žádné okno s hledaným objektem. Takto získaná okna jsou podstoupena dalším silnějším klasifikátorům v kaskádě, které rozhodují zda se v daném okně hledaný objekt nachází či ne [28].

Jednotlivé stupně kaskády jsou vytvořeny pomocí trénování klasifikátorů výše popsaným algoritmem AdaBoost. Začíná se s dvoupříznakovým klasifikátorem, efektivní filtr obličeje je vytvořen regulováním silného klasifikátoru pomocí změny rozhodovací úrovně, aby se minimalizoval počet chybných detekcí. Inicializační rozho-

dovací úroveň algoritmu AdaBoost

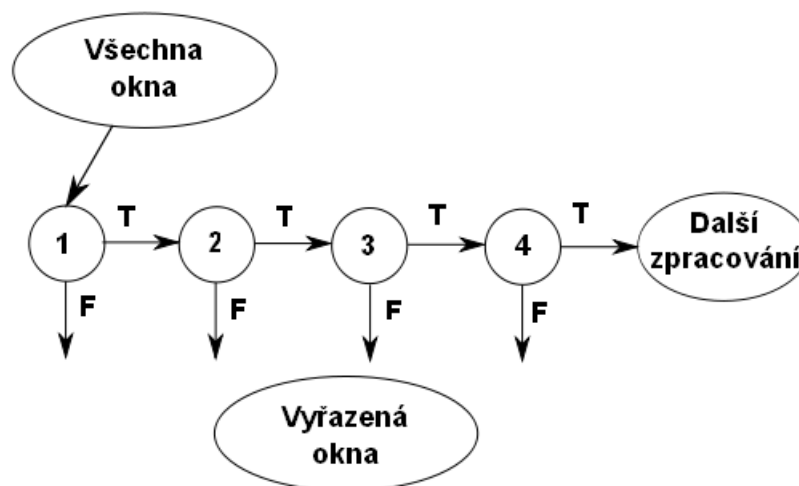
$$\frac{1}{2} \sum_{t=1}^T \alpha_t \quad (3.5)$$

je nastavena tak, aby se vnášela co nejmenší chyba do trénovacích dat. Nižší rozhodovací úroveň znamená vyšší počet detekcí, ale zároveň vyšší počet chybných detekcí. Jak je uvedeno v literatuře [28], dvoupříznakový klasifikátor je schopen detekovat 100 % obličejů v obraze, ale zároveň je zatížen chybou 40 % chybných detekcí [28].

Schopnosti detekce dvoupříznakového klasifikátoru jsou daleko za možnostmi použití pro systémy detekce. Nicméně takovýto klasifikátor může významně snížit množství oken, která potřebují další zpracování užitím jednoduchých operací:

1. Ohodnocení obdélníkových příznaků (vyžadováno 6 až 9 odkazů na obdélník).
2. Výpočet slabého klasifikátoru pro každý příznak (nutná jedna rozhodovací operace na příznak).
3. Kombinace slabých klasifikátorů.

Dvoupříznakový klasifikátor vyžaduje kolem 60 operací mikroprocesoru. Celkový rozhodovací proces je tedy upraven na degradovaný rozhodovací strom, který nazýváme kaskádou. Pozitivní výsledek prvního filtru je postoupen zpracování druhému filtru, který je nastaven tak, aby také dosahoval vysokých detekčních schopností. Pozitivní výsledek druhého filtru je následně zpracován třetím filtrem a tak dále. Negativní výsledek jakéhokoliv filtru kaskády vede k okamžitému zahození daného okna, každý klasifikátor v kaskádě má možnost dané okno z procesu prohledání vyřadit [28].



Obr. 3.6: Kaskádní kombinace klasifikátorů [28].

## 3.2 VH projekce

VH projekce je poměrně nový detektor obličejů, který byl poprvé publikován v roce 2009. Narozdíl od detektoru Viola-Jones, který je určen pro detekci obličejů ve viditelném spektru, se detektor VH projekce používá pro detekci obličejů v termografickém spektru. Autoři tohoto detektoru v literatuře [10] uvádějí, že je pro detekci obličejů v termografickém spektru asi desetkrát rychlejší než klasický detektor Viola-Jones. Další výhodou je, že není nutné tento detektor trénovat na zkušební databázi obličejů. Velkým omezením tohoto detektoru ovšem je, že při větším množství obličejů v obraze tento detektor selhává nebo podává velmi zkreslené výsledky.

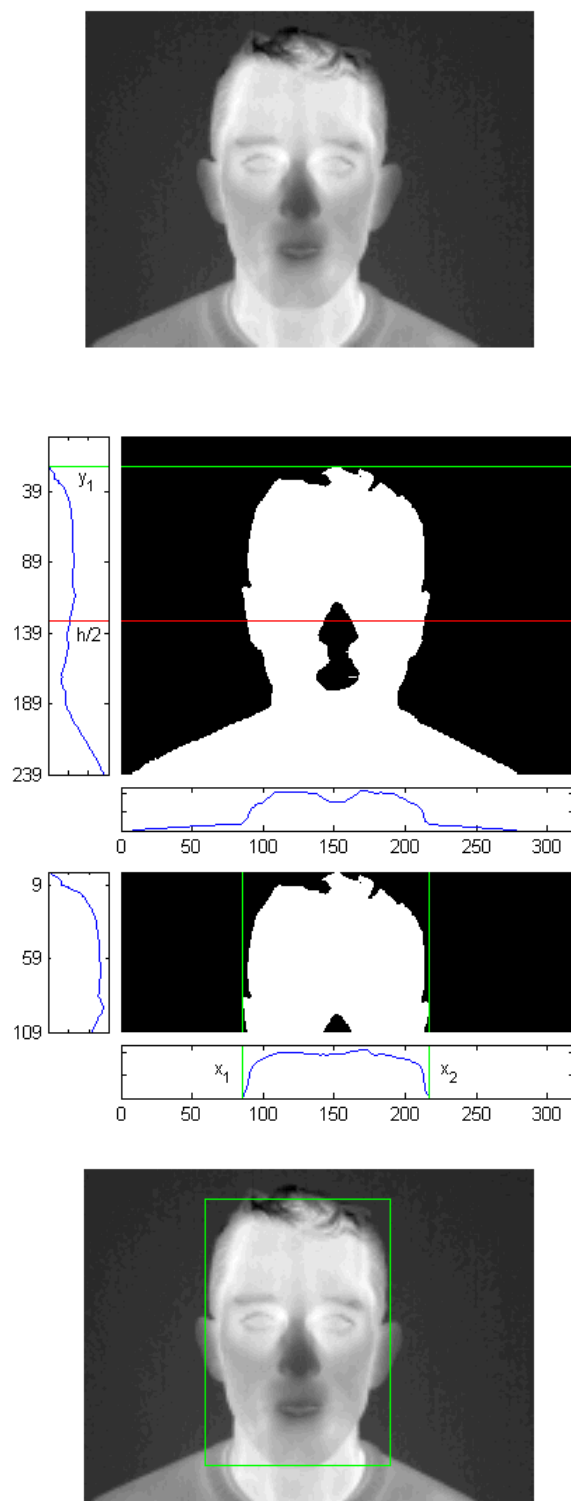
Dále bude popsán algoritmus detekce pro obrazy v termografickém spektru. Cílem této metody je nalezení souřadnic vrcholů obdélníku  $(x_1, y_1, x_2, y_2)$ , který jednoznačně určuje, kde se v obraze nachází hledaný obličej.

Algoritmus detekce:

1. Obraz je nejprve binarizován, práh  $T$  je volen Otsuho metodou (viz kapitola 3.3.1). Jestliže je  $f(x, y) < T$ , potom  $g(x, y) = 0$ , jinak  $g(x, y) = 1$ .
2. Je spočtena vertikální a horizontální projekce.
3. První hranice vertikálního profilu je označena jako  $y_1$ .
4.  $h$  je vzdálenost od  $y_1$  až ke spodní části obrazu.
5. Je vybrána část obličejů od  $y_1$  do  $h/2$  a následně jsou určeny pozice  $x_1$  a  $x_2$  jako levá a pravá hranice tohoto výřezu.
6. Spodní část obličejů je určena ze vztahu

$$y_2 = y_1 + (x_2 - x_1) \cdot (13/9). \quad (3.6)$$

Jednotlivé kroky VH projekce jsou schematicky naznačeny na obr. 3.7. První je obraz zaznamenaný termografickou kamerou, poslední obraz znázorňuje výstup detekce [10].



Obr. 3.7: Schematické znázornění funkce detektoru VH projekce [10].

## 3.3 Segmentace

Segmentace obrazu je důležitou technologií pro zpracování obrazu. Existuje mnoho aplikací, ať už na syntézu objektů nebo počítačové grafiky, které vyžadují přesnou segmentaci. Segmentaci lze popsat jako proces, při kterém je celý obraz dělen do určitých regionů, které lze označit jako  $\mathbf{R}_1$  až  $\mathbf{R}_n$ . V ideálním případě je každý objekt v obraze označen jako jistý region  $\mathbf{R}_i$ , celý obraz je potom složen z  $n$  regionů. Segmentace je řazena mezi náročné techniky zpracování obrazu a je ovlivňována mnoha faktory, jako jsou osvětlení, šum nebo rozmazání daného obrazu. Objekty se v obraze mohou překrývat nebo mohou být velmi složité. Z tohoto důvodu neexistuje žádná komplexní metoda segmentace, která by podávala dostatečné výsledky za všech podmínek a zároveň byla aplikovatelná v každé oblasti zpracování obrazu [15].

Jak uvádí literatura [27] segmentace obrazu  $\mathbf{A}[m, n]$  je definována jako jeho dělení na podobrazy  $\mathbf{R}_1, \mathbf{R}_2, \dots, \mathbf{R}_n$  tak, že tyto podobrazy splňují následující podmínky:

1.  $\bigcup_{i=1}^n \mathbf{R}_i = \mathbf{A}[m, n]$ ,
2.  $\mathbf{R}_i \cap \mathbf{R}_j = \emptyset, i \neq j$ ,
3. Každý podobraz splňuje nějaké tvrzení, popř. množinu tvrzení:
  - všechny pixely v podobraze  $\mathbf{R}_i$  mají stejnou úroveň šedi,
  - všechny pixely v podobraze  $\mathbf{R}_i$  se neliší v úrovni šedi více než o předepsanou hodnotu,
  - standardní odchylka úrovně šedi všech pixelů podobrazu  $\mathbf{R}_i$  je dostatečně malá [27],
  - všechny pixely v podobraze  $\mathbf{R}_i$  reprezentují jeden druh objektu, např. oblohu, lidskou postavu či obličej.

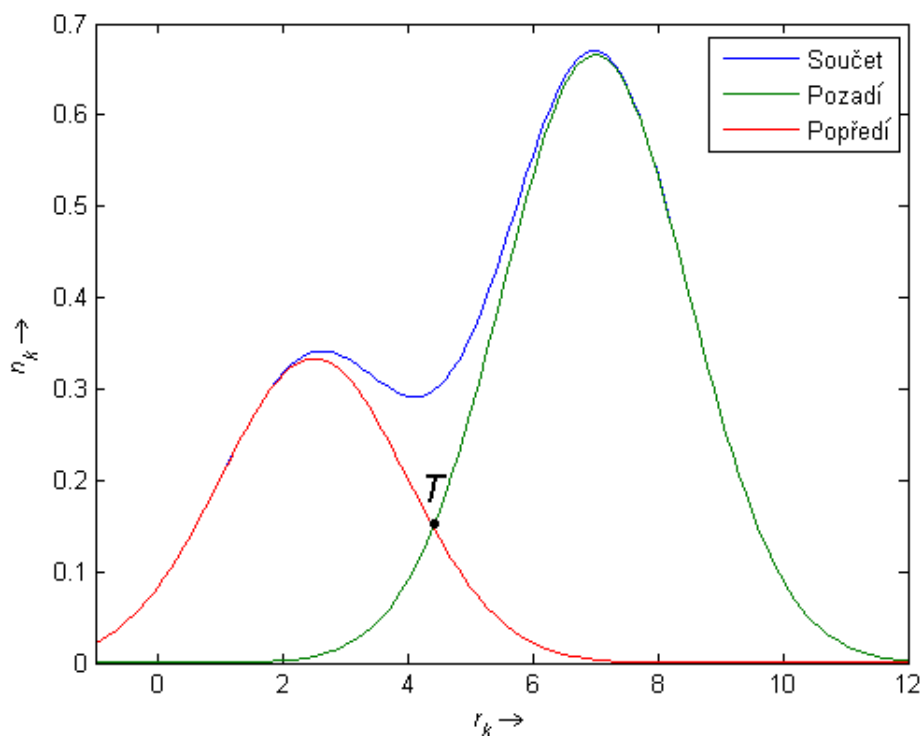
Segmentačních metod tedy existuje velké množství, ovšem dále v textu bude pojednáno jen o těch, které jsou využívány u algoritmů detekce obličeje pomocí VH projekce.

### 3.3.1 Otsuho práhová metoda

Tato metoda se používá pro převod šedotónového obrazu na obraz monochromatický, jinými slovy se jedná o efektivní nástroj pro separaci objektů a pozadí. Otsuho metoda, pojmenovaná po svém tvůrci Nubuyukimu Otsuovi je jednou z mnoha známých binarizačních metod, která vyniká svou jednoduchostí a nízkou výpočetní náročností. Tuto transformaci lze definovat jako pozičně nezávislou bodovou transformaci, což je vyjádřeno vztahem

$$g(x, y) = \begin{cases} 1 & \text{jestli } f(x, y) \geq T, \\ 0 & \text{jinak,} \end{cases} \quad (3.7)$$

kde  $f(x, y)$  je vstupní šedotónový obraz a  $g(x, y)$  je výsledný binární obraz,  $T$  je práh intenzity. Jedna reprezentuje popředí, nula pozadí. Prah  $T$  se volí buď ručně, nicméně snahou je tento krok zautomatizovat, což se provádí analýzou histogramu obrazu. V něm se formují tzv. clustery obrazu, z nichž jeden reprezentuje popředí a druhý pozadí, jejich složením vznikne celkový průběh histogramu. Ideální práh  $T$  lze zvolit nalezením lokálního minima v průběhu histogramu, což ovšem většinou není možné, protože se oba clustery překrývají. V takovém případě se práh  $T$  volí tam, kde se průběhy obou clusterů v histogramu protínají (viz obr. 3.8) [13] [18].



Obr. 3.8: Dva clustery, jejichž součet tvoří celkový průběh histogramu.

Otsuho metoda hledá optimální práh  $T$  na základě výpočtu rozptylu. Jsou definovány dva rozptyly, tzv. mezi-rozptyl (značený  $m_r$ ) a vnitřní rozptyl (značený  $v_r$ ), které reprezentují rovnice

$$\sigma_{m_r}^2(T) = \omega_B(T)\omega_0(T)[\mu_B(T) - \mu_0(T)]^2, \quad (3.8)$$

$$\sigma_{v_r}^2(T) = \omega_B(T)\sigma_B^2 + \omega_0(T)\sigma_0^2(T), \quad (3.9)$$

kde

$$\omega_B(T) = \sum_{i=0}^{T-1} p(i), \omega_0(T) = \sum_{i=T}^{L-1} p(i), \quad (3.10)$$

jsou váhy.  $\sigma_B^2(T)$  je rozptyl pixelů pozadí (pod práhem),  $\sigma_0^2(T)$  je rozptyl pixelů popředí (nad práhem).  $\mu_B(T)$  a  $\mu_0(T)$  jsou průměrné hodnoty intenzit pozadí a popředí,  $\langle 0, L - 1 \rangle$  představuje počet úrovní intenzit v obraze. Celkový rozptyl je potom dán vztahem

$$\sigma^2(T) = \sigma_{\text{mr}}^2(T) + \sigma_{\text{vr}}^2(T). \quad (3.11)$$

Cílem je najít takový práh  $T$ , který maximalizuje mezi-rozptyl, popř. minimalizuje vnitřní rozptyl. Výhodnější z hlediska nižší výpočetní náročnosti je počítat mezi-rozptyl [18].

### 3.4 Aktivní modely tvaru (ASM) a vzhledu (AAM)

Tyto metody pracují s normalizovaným statistickým modelem tvaru, který lze vytvořit pomocí manuálně labelovaných trénovacích dat. Tyto data představují množinu obrazů, typických příkladů toho, co se má následně v obraze hledat. Nutným vstupem těchto metod je tedy jistá skupina bodů (tzv. landmarks), které popisují tvar vzoru, a které mohou být spolehlivě nalezeny na každém trénovacím obraze. Tyto body (viz obr. 3.9) by měly ležet na významných místech (např. hrany), která se dají jednoznačně popsat pomocí tzv. deskriptorů [1][4][23].

Pro 2D obraz lze definovat  $n$  významných bodů  $(x, y)$ , pro jediný příklad jako dvourozměrný vektor  $\mathbf{x}$ , kde

$$\mathbf{x} = (x_1, \dots, x_n, y_1, \dots, y_n)^T. \quad (3.12)$$

Jestliže je k dispozici  $m$  trénovacích vzorků, potom lze generovat  $m$  jako vektor  $\mathbf{x}_j$  [1].

Nyní lze definovat funkci transformace podobnosti (za předpokladu, že významné body jsou k sobě vztaženy invariátním způsobem) následující rovnicí:

$$T \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x_p \\ y_p \end{pmatrix} + \begin{pmatrix} s \cdot \cos \theta & s \cdot \sin \theta \\ -s \cdot \sin \theta & s \cdot \cos \theta \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix}, \quad (3.13)$$

kde  $(x_p, y_p)$  je posun,  $s$  je změna měřítka a úhel  $\theta$  je rotace modelu tvaru. Rovnice 3.13 vyjadřuje průměrný model tvaru z trénovacích dat zarovnáním všech modelů tvaru navzájem. Průměrný statistický model tvaru lze jednoduše modifikovat v rozsahu trénovacích modelů pomocí vektoru parametrů deformačního modelu  $\mathbf{b}$ , což vyjadřuje rovnice 3.17 [23].

Z trénovacích dat nejsou určeny jen důležité body (jejich umístění a vzájemná poloha), ale také tzv. deskriptory. Deskriptory slouží k jednoznačnému určení daného významného bodu v jakémkoliv obraze. Pro výpočet deskriptorů se používají různé

přístupy, jako např. profily intenzity jasu na spojnici mezi dvěma významnými body, 2D diskrétní kosinová transformace, waveletová transformace nebo lze použít nějaký klasifikační algoritmus (v literatuře [23] je použit GentleBoost). Při využití intenzity jasu vznikne tzv. model profilů intenzity jasu (ten se také využívá dále v této práci).

Metoda ASM sestává ze dvou neustále se opakujících iterativních kroků, které se provádí dokud nedojde k požadované míře konvergence. Po umístění průměrného modelu tvaru do obrazu se v prvním kroku pomocí deskriptorů (model profilů intenzity jasu) určí nejpravděpodobnější pozice významných bodů a v kroku druhém jsou modifikovány tak, aby byl dodržen tvar definovaný statistickým modelem tvaru [3] [23].



Obr. 3.9: Ukázka rozložení bodů na významných částech obličeje.

AAM rozšiřuje techniku ASM o statistický popis intenzity pixelů v obrazu, fáze trénování a tvorby statistického modelu tvaru jsou shodné s ASM. Pro vytvoření popisu distribuce intenzity obrazu je nezbytné zdeformování (warping) všech trénovacích modelů tak, aby body (landmarks) ležely na pozicích průměrného modelu tvaru  $\bar{x}$ . Zdeformované trénovací modely se navzorkují a normalizují, aby se odstranil vliv různého jasu. Nyní lze opět použít funkci transformace podobnosti, čímž dojde k vytvoření dvou modelů, z nichž jeden popisuje tvar a druhý distribuci intenzity. Jejich sjednocením vznikne jeden tzv. vzhledový model, který popisuje vztahy mezi intenzitou a tvarem v celé trénovací množině. Tato metoda je narozdíl od metody ASM velice náchylná na velmi přesné umístění počátečního modelu tvaru v daném obrazu a při jeho nepřesném umístění potom selhává. Podrobnosti lze nalézt v literatuře [1] a [4].



### 3.4.1 Analýza hlavních komponent (PCA)

Obě předchozí metody zmíněné v kapitole 3.4 používají ke své činnosti algoritmu PCA, což je statistická metoda redukce vstupních dat, která zachovává maximální možnou variabilitu těchto dat [16] [20].

Předpokládejme, že máme  $m$  trénovacích vzorků, které jsou reprezentovány vektorem  $\mathbf{x}_j$ , kde  $j = 1, \dots, m$ . Algoritmus PCA je potom definován následujícími kroky:

1. Výpočet průměrného modelu tvaru z množiny  $m$  trénovacích vzorků pomocí rovnice

$$\bar{\mathbf{x}} = \frac{1}{m} \sum_{j=1}^m \mathbf{x}_j. \quad (3.14)$$

2. Mezivýpočet pro určení kovariační matice z trénovacích dat jako

$$\mathbf{S} = \frac{1}{m} \sum_{j=1}^m (\mathbf{x}_j - \bar{\mathbf{x}})(\mathbf{x}_j - \bar{\mathbf{x}})^T. \quad (3.15)$$

3. Sestavení kovariační matice jako

$$\Phi = [\phi_1 | \phi_2 | \dots | \phi_q], \quad (3.16)$$

kde  $\phi_j$ ,  $j = 1, \dots, q$  reprezentuje vlastní vektory matice  $\mathbf{S}$  korespondující s nejvyšší  $q$  hodnotou vektoru vlastních hodnot.

4. Ze znalosti kovariační matice  $\Phi$  a průměrného modelu tvaru  $\bar{\mathbf{x}}$  lze aproximovat každý tvar podle rovnice

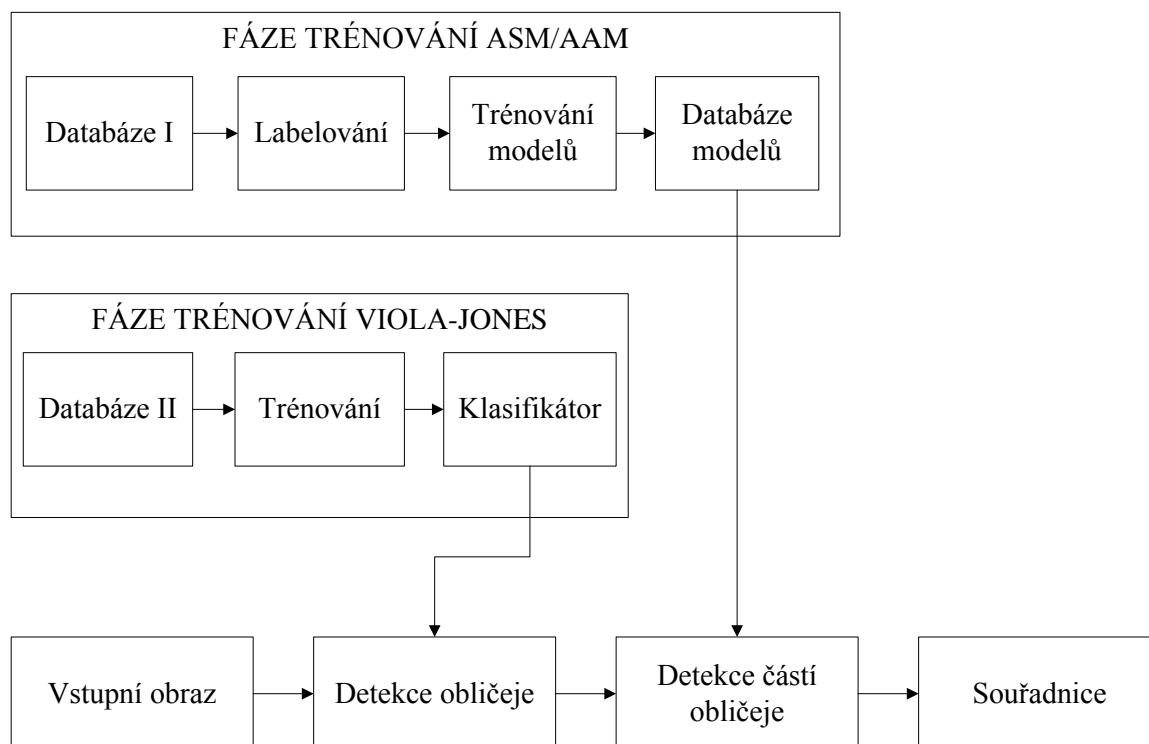
$$\mathbf{x}_j \approx \bar{\mathbf{x}} + \Phi \cdot \mathbf{b}, \quad (3.17)$$

kde

$$\mathbf{b}_j = \Phi^T (\mathbf{x}_j - \bar{\mathbf{x}}). \quad (3.18)$$

Metoda PCA je známá v problematice zabývající se zpracováním řeči, kde se používá k identifikaci mluvčího. V tomto případě se používá pro vytvoření matice vlastních vektorů, které se používají k určení nejlepšího obrazu ze sady, jenž je následně porovnáván s aktuálním objektem detekce. Tato podkapitola čerpá převážně z literatury [16].

## 4 PRAKTICKÁ REALIZACE METODY PRO DETEKCI ČÁSTÍ OBLIČEJE



Obr. 4.1: Navržené schéma metody detekce významných částí obličeje.

Obecné blokové schéma navržené metody pro detekci významných částí obličeje je znázorněno na obr. 4.1. Tato metoda vychází z již publikovaných metod detekce, které byly ovšem doposud použity jen pro detekci obličeje a jeho částí ve viditelném spektru. Praktická realizace je provedena s pomocí programu Matlab a dalších open-source nástrojů napsaných v jazyce C/C++ (OpenCV, kap. 4.1.1). Schéma sestává z jedenácti bloků, z nichž každý má svou specifickou funkci, která bude nyní stručně popsána (podrobnější informace k jednotlivým fázím budou popsány dále v textu).

*Výpis jednotlivých bloků a jejich funkce:*

### **Vstupní obraz**

Tento blok představuje vstup celého detekčního systému, jedná se o šedotónový obraz v termografickém spektru, který zachycuje jednu osobu, dívající se směrem k objektivu, takže je vidět její obličej z předního pohledu.

## **Detekce obličeje**

Tento blok představuje detekci obličeje ve vstupním obrazu. Detekce obličeje je realizována pomocí detektoru Viola-Jones (viz kap. 3.1) nebo pomocí detektoru VH projekce (viz kap. 3.2).

## **Detekce částí obličeje**

Vstupem tohoto bloku jsou čtyři souřadnice, které definují polohu obličeje v obrazu. Pouze na plochu definovanou zmíněnými čtyřmi souřadnicemi je aplikována technika ASM popř. AAM (viz kap. 3.4), které zde slouží k detekci významných bodů obličeje.

## **Souřadnice**

Výstupem celého detekčního systému je soubor souřadnic, které definují významné části obličeje (obě oči, nos). Detekce úst byla z detekčního systému vyloučena z důvodů, které jsou popsány v kapitole 4.1.4.

### *FÁZE TRÉNOVÁNÍ ASM/AAM*

#### **Databáze I**

Blok Databáze I představuje určitý soubor trénovacích šedotónových obrazů v termografickém spektru, na kterých je zachycena jedna osoba dívající se směrem do objektivu.

#### **Labelování**

Blok Labelování představuje ruční umístění bodů na významné části obličeje, které bude systém detekovat.

#### **Trénování modelů**

Tento blok představuje natrénování modelů, které jsou nezbytné pro funkci algoritmů ASM i AAM.

#### **Databáze modelů**

Výstupem bloku Trénování modelů je soubor již natrénované databáze, která následně slouží jako vstup pro blok Detekce částí obličeje.

### *FÁZE TRÉNOVÁNÍ VIOLA-JONES*

#### **Databáze II**

Blok Databáze II v této fázi trénování představuje již normalizovaný soubor trénovacích šedotónových obrazů v termografickém spektru, na kterých je zachycen pouze obličej osob, způsob normalizace je uveden v kapitole 4.1.2.

## Trénování

Blok Trénování schematicky naznačuje nutnost natrénovat detektor obličeje Viola-Jones před prvním použitím.

## Klasifikátor

Tento blok představuje již natrénované klasifikátory pro detektor Viola-Jones.

### 4.1 Fáze trénování

Tato kapitola se zabývá trénováním jak detektoru obličeje Viola-Jones, tak i trénováním metod ASM a AAM, které jsou použity k detekci částí obličeje. K tomu, aby bylo možno nějaký detektor natrénovat, je potřebná databáze vzorů, v našem případě se jedná o databázi obličejů v termografickém spektru. Použitá databáze byla poskytnuta Ústavem telekomunikací při Fakultě elektrotechniky a komunikačních technologií na VUT v Brně a obsahuje 2460 šedotónových obrázků zachycených termální kamerou TESTO 880–3. Na těchto obrázcích je zachyceno 41 osob na čtyřech různých sezeních, přičemž ke každé osobě je k dispozici 60 snímků, jejichž rozlišení je  $320 \times 240$  px, viz obr. 4.2. Tato databáze byla dále rozdělena na tři části, které sloužily k různým účelům (jen u technik ASM a AAM). Maximálně tři snímky vybraných 25 osob byly použity pro trénování, všechny snímky jiných 14 osob (celkem 840 snímků) byly využity pro testování a vybrané snímky posledních 2 osob byly použity pro ověřování funkčnosti algoritmů.



Obr. 4.2: Ukázka z použité databáze termografických snímků.

### 4.1.1 OpenCV

OpenCV (Open Source Computer Vision) je knihovna počítačového vidění, která je dostupná z [30]. Projekt OpenCV vznikl v roce 1999 ve společnosti Intel, která jej poskytuje pod licenci BSD. Ve stručnosti uveďme, že tato licence umožňuje spojení s jiným počítačovým programem, navíc není nabyvatel licence povinen poskytovat zdrojové kódy tohoto programu při jeho následné distribuci. Každý program vytvořený pod touto licenci ovšem musí obsahovat její znění. Knihovna je napsána v C/C++ a je použitelná na platformách Linux, Windows i Mac OS X. Je také dostupná na rozhraních pro Python, Ruby, Matlab a některé funkce jsou dostupné i pod GPU [2].

Knihovna OpenCV byla navržena s ohledem na výpočetní náročnost se silným zaměřením na aplikace pracující v reálném čase, optimalizaci jejího kódu byla tudíž věnována velká pozornost. První verze 1.0 vyšla v roce 2006, v říjnu roku 2009 následně vyšla další verze 2.0, která se dočkala na konci roku 2010 vylepšení a vznikla tak zatím poslední verze 2.2. Od verze 2.0 je implementována podpora pro vícejádrové procesory [2].

Jedním z cílů OpenCV je poskytovat jednoduchou infrastrukturu počítačového vidění, která tvůrcům pomáhá budovat velmi rychle poměrně sofistikované aplikace. Knihovna OpenCV obsahuje více než 500 funkcí, které zahrnují mnoho oblastí počítačového vidění, jako jsou inspekce výrobků, zpracování lékařských obrazů, bezpečnost, uživatelské rozhraní, kalibrace kamery, stereo vidění a robotika. Vzhledem k tomu, že počítačové vidění a strojové učení bývají úzce spjatá, OpenCV také poskytuje Machine Learning Library (MLL), která slouží k již zmíněnému strojovému učení. Funkce strojového učení bude využita v následující kapitole 4.1.2. Tato knihovna je zaměřena na statistické rozpoznávání a clustering. Knihovna MLL je velmi užitečná pro úkoly vidění, které jsou jádrem OpenCV, zároveň je tato knihovna dostatečně obecná pro aplikaci na jakýkoliv problém strojového učení [2].

Komerční alternativou nebo doplňkem knihovny OpenCV může být knihovna IPP (Intel Performance Primitives), která je rovněž od společnosti Intel. Jestliže je nainstalována na systém, kde je již OpenCV v provozu, potom dokáže podstatně urychlit práci této knihovny díky optimalizovaným nízkourovňovým rutinám [2].

### 4.1.2 Trénování detektoru Viola-Jones

K natrénování tohoto detektoru byla s výhodou využita zmíněná knihovna OpenCV, která ve své základní instalaci obsahuje utilitu *haartraining* [31]. Ta slouží k vytvoření souboru natrénovaných kaskád klasifikátorů, které lze využít k detekci jakéhokoliv vzoru (nemusí se nutně jednat o obličej), v literatuře [12] je tato kaskáda například natrénována na míse. Fáze trénování je velice časově náročný a na výkonu počítače

závislý proces, který vyžaduje značné množství vstupních dat. Manuál (html soubor dostupný po nainstalování OpenCV) k utilitě *haartraining* uvádí, že je zapotřebí asi 7000 pozitivních a 3000 negativních vzorků, lze ovšem použít i jiné množství. Zde je ovšem zapotřebí zmínit, že čím více vzorků, jak negativních, tak pozitivních je použito, tím lépe bude výsledný klasifikátor fungovat.

K trénování je tedy zapotřebí dvou různých trénovacích sad, z nichž jedna obsahuje negativní vzorky, což jsou vzorky, kde se nevyskytuje žádný obličej (jedná se o pozadí). V našem případě byly jako negativní vzorky použity výřezy pozadí z použité databáze o různých velikostech. Jako pozitivní vzorky byly použity obličeje detekované pomocí metody VH projekce, tato metoda nepotřebuje ke své činnosti trénování. Ukázky použitých obličejů na trénování jsou na obrázku 4.3. Z důvodu zrychlení trénování se doporučuje pozitivní trénovací vzorky upravit na nižší rozlišení, v našem případě  $20 \times 20$  px, jak doporučuje literatura [12], upravené vzorky jsou zobrazeny na obr. 4.4. Celkový počet negativních vzorků byl 1400 a celkový počet pozitivních vzorků 4200.



Obr. 4.3: Ukázka obličejů z databáze termografických snímků, získaných metodou VH projekce.

Vlastní spouštění programu pro trénování se provádí z příkazové řádky. Rychlost a hlavně kvalitu výsledného klasifikátoru lze ovlivnit několika parametry, jejichž výpis následuje.

- **data** <název\_složky>  
Určuje složku, do které se ukládají natrénované klasifikátory.
- **vec** <název\_souboru>  
Soubor s pozitivními vzorky vytvořený pomocí (*trainingsamples*) utility.
- **bg** <název\_souboru>  
Soubor s negativními vzorky.



Obr. 4.4: Ukázka těch samých upravených obličejů pomocí utility *createsamples* na rozlišení  $20 \times 20$  px.

- **npos** <počet\_pozitivních\_vzorků>  
Počet pozitivních vzorků.
- **nneg** <počet\_negativních\_vzorků>  
Počet negativních vzorků.
- **nstages** <počet\_stupňů\_kaskády>  
Počet stupňů kaskády.
- **nsplits** <počet\_dělení>  
Určení použitého slabého klasifikátoru, 1 (obyčejný klasifikátor) nebo 2 a víc (vylepšený klasifikátor).
- **mem** <paměť\_v\_MB>  
Použitelná paměť pro trénink, čím víc, tím je trénování rychlejší.
- **sym** <defaultně>
- **nonsym**  
Určují, jestli jsou trénovací objekty symetrické podle vertikální osy.
- **minhitrate** <počet>  
Minimální stupeň správných detekcí pro jednotlivé stupně kaskády.
- **maxfalsealarm** <počet>  
Maximální stupeň chybných detekcí pro každý stupeň kaskády.
- **weighttrimming** <poměr>  
Poměr pro zmenšování okna při vytváření příznaků.
- **mode** <BASIC (default) | CORE | ALL>  
Výběr typu Haarových příznaků, BASIC označuje pravoúhlé a ALL všechny příznaky (i jejich  $45^\circ$  rotaci).
- **w** <šířka>
- **h** <výška>  
Určují velikost trénovacích obrázků.

Na následujících řádcích je výpis příkazů použitých k trénování klasifikátoru:  
`createsamples.exe -info positive.txt -vec positive.vec -num 4200 -w 20  
-h 20`

`haartraining.exe -data data/cascade -vec positive.vec -bg negative.txt  
-npos 4200 -nneg 1400 -nstages 30 -mem 1000 -mode ALL -w 20 -h 20`

`convert_cascade --size="20x20" data/cascade haarcascade_th_cativite.xml`

Pozn. `positive.txt` a `negative.txt` jsou textové soubory, ve kterých jsou definovány cesty k pozitivním a negativním vzorkům určeným pro trénování. Poslední příkaz slouží k převodu natrénované kaskády do `.xml` souboru.

Výstupem celého trénovacího procesu je tedy `.xml` soubor, který obsahuje informace o natrénovaných stupních kaskády, které obsahují počet definovaných příznaků. Tento `.xml` soubor bude použit pro detekci obličeje pomocí programu Matlab, což bude objasněno dále v textu.

### 4.1.3 Active Shape Model (ASM) and Active Appearance Model (AAM) toolbox

Tento toolbox, který je dostupný na serveru MatlabCentral (viz [17]) se, jak již název napovídá, věnuje zpracování obrazových dat pomocí technik ASM a AAM tak, jak byly představeny Cootesem a Taylorem (detailnější informace lze nalézt v literatuře [5]). Toolbox byl naprogramován prof. Dirk-Jan Kroonem a jeho týmem na University of Twente, Netherlands. Obsahuje jak funkce pro manuální labelování obrázků, tak funkce pro následnou detekci a je distribuován s licencí BSD, takže je možné jej použít pro naše účely. Pro jeho činnost je zapotřebí mít nainstalován Matlab s Image Processing Toolbox. Původní využití bylo pro detekci kontury ruky s bílým pozadím, základní funkčnost tohoto toolboxu byla ovšem změněna tak, aby byl použitelný pro účely detekce částí obličeje v termografickém spektru. Tyto úpravy jsou popsány v kapitole 4.4.

#### ***Struktura toolboxu***

– AAM Functions

*AAM\_align\_data.m*

*AAM\_align\_data\_inverse.m*

*AAM\_Appearance2Vector.m*

*AAM\_ApplyModel.m*

*AAM\_CombineShapeAppearance.m*



- AAM\_MakeAppearanceModel.m*
- AAM\_MakeSearchModel.m*
- AAM\_MakeShapeModel.m*
- AAM\_NormalizeAppearance.m*
- AAM\_Vector2Appearance.m*
- AAM\_Weights.m*
- ASM Functions
  - ASM\_align\_data.m*
  - ASM\_align\_data\_inverse.m*
  - ASM\_ApplyModel.m*
  - ASM\_GetContourNormals.m*
  - ASM\_getProfileAndDerivatives.m*
  - ASM\_MakeAppearanceModel.m*
  - ASM\_MakeShapeModel.m*
- Functions
  - DrawContourGui.fig*
  - DrawContourGui.m*
  - drawObject.m*
  - linspace\_multi.m*
  - LoadDataSetNiceContour.m*
  - PCA.m*
  - PreProcessCp2tform.m*
  - SelectPosition.m*

Pozn. Popisy a požadavky jednotlivých funkcí lze nalézt přímo v *.m* souborech jako nápovědu, z toho důvodu zde nebudou jednotlivě rozebírány.

#### 4.1.4 Trénování ASM a AAM

Trénování modelů těchto dvou metod bylo prováděno s využitím programu Matlab ve verzi 7.7.0.471 (R2008b) a toolboxu představeného v kapitole 4.1.3. K vlastnímu manuálnímu labelování obrázků slouží v toolboxu *DrawContourGui*, které definuje jednoduché funkce pro načtení obrázku, následné umístování bodů pomocí myši a také funkce pro uložení informací o obrázku, počtu umístěných bodů a jejich pozicích do *.mat* souboru.

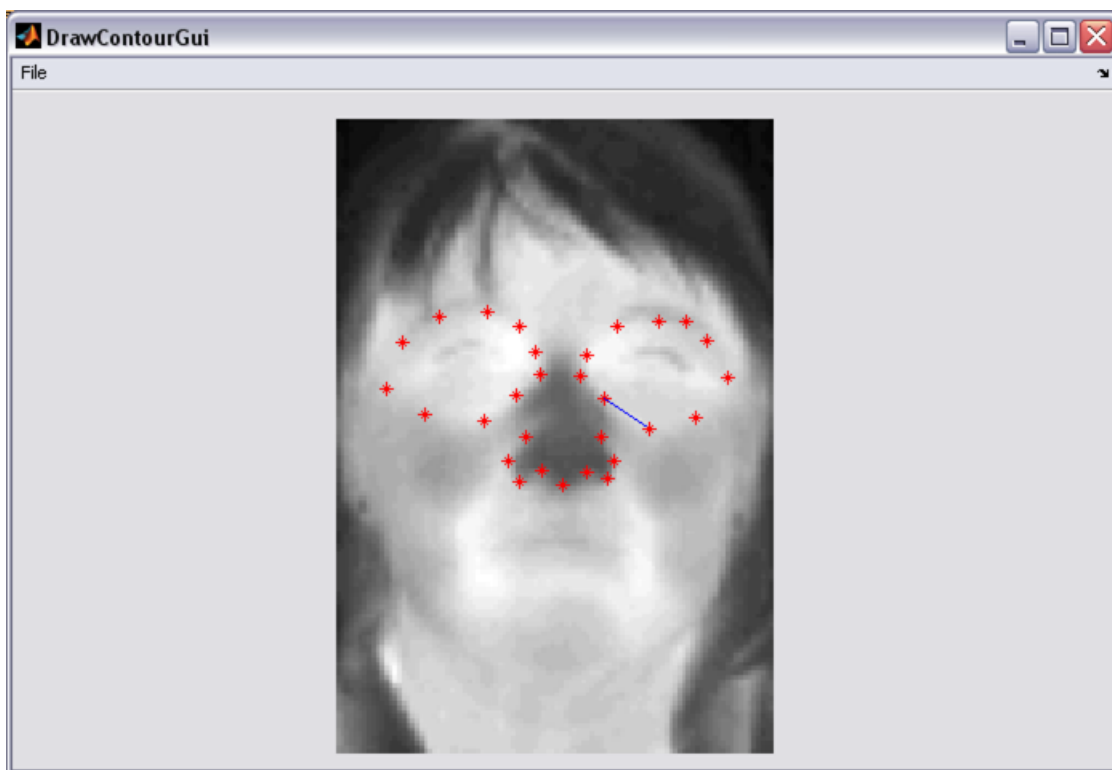
Ukázka rozložení bodů na obličejí je vidět na obr. 4.5. Je možné si povšimnout, že na tomto obrázku chybí označení kontury úst, protože detekce této části obličeje byla z detekčního systému vyloučena, protože nevykazovala dobré výsledky. Důvodů k vyloučení detekce úst je několik:

1. Největším problémem je fakt, že na většině snímcích z databáze nelze přesně určit konturu úst, což je způsobeno tím, že ústa mají podobnou (ne-li stejnou) teplotu jako tváře. Z tohoto faktu je zřejmé, že ústa splývají se svým okolím, takže není možné je manuálně labelovat.
2. Detekce úst není z pohledu dalšího využití celého systému až tak podstatná.
3. Tím, že se vyloučí detekce úst, dojde ke snížení celkového počtu bodů, které definují model, takže dojde ke zrychlení celého detekčního systému [20].

Počty bodů, které jsou určeny pro vytvoření modelu, jsou následující:

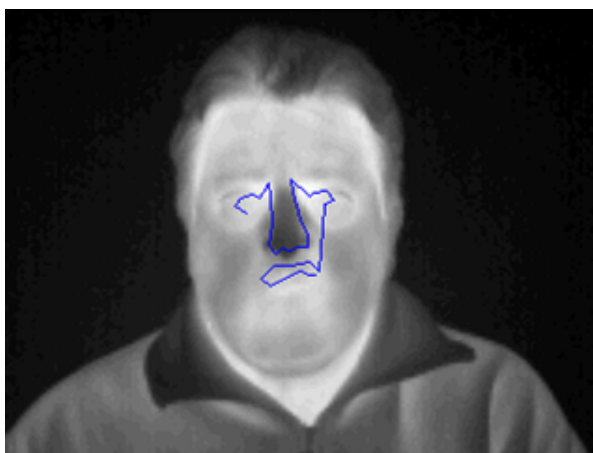
- LEVÉ OKO  $\Rightarrow$  9 bodů
- PRAVÉ OKO  $\Rightarrow$  9 bodů
- NOS  $\Rightarrow$  11 bodů

Celkem je tedy model definován 29 body pro techniku ASM i AAM. Na obrázku 4.5 je vidět, že kontura nosu je snadno rozeznatelná, takže s umístováním bodů nebyl žádný problém, s konturou očí je to ovšem horší. Při labelování tedy nebyly označovány přímo oči, ale jakési očnice, které jsou pozorovatelné na většině subjektech z trénovací množiny. Očnice jsou shora definovány konturou spodní části obočí a zdola přechodem tmavší části líce a světlejší části vlastního prostoru oka tak, jak je to vidět na obr. 4.5.



Obr. 4.5: Ukázka rozložení bodů pro techniky ASM i AAM.

Na obr. 4.6 je znázorněn první návrh modelu pro detekci, který sloužil jak pro přesnou detekci očí, tak i úst. Oči byly shora definovány horními víčky a zdola spodními víčky (byla-li viditelná). Tento model byl natrénován na 400 manuálně labelovaných obrázcích. Z obr. 4.6 je také patrné, proč byla detekce úst z detekčního systému vyloučena. Nakonec byl ovšem tento model zamítnut, protože nepodával přesvědčivé výsledky.



Obr. 4.6: Ukázka prvního vytvořeného modelu, který sestával z 33 bodů.

Po nalabelování trénovacích obrázků se může přikročit k vlastnímu trénování. Zde je ovšem ještě potřeba zdůraznit, že pro trénování technik ASM a AAM jsou doporučována nekorelovaná data, což v podstatě znamená, že je možno použít pouze jeden snímek od každého z 25 lidí trénovací množiny. Byly proto natrénovány modely z 25 snímků (od každé osoby jeden), 50 snímků (od každé osoby dva) a 75 snímků (od každé osoby tři). Aby byla zajištěna alespoň částečná nekorelovanost dat, byly snímky každé osoby vybírány z různých sezení. Vlastní trénování probíhalo pomocí dvou skriptů vytvořených v Matlabu a jsou k dispozici na přiloženém DVD. Tyto skripty obsahují funkce z toolboxu popsaného v kapitole 4.1.3, jeden slouží k trénování modelů metody ASM a druhý k trénování modelu metody AAM.

Na následujících řádcích je uveden výpis důležitých parametrů a funkcí použitých k trénování modelů metody ASM:

```
% Počet interpolovaných bodů mezi body definujícími konturu.  
options.ni = 2;  
% Délka profilu intenzit každého bodu definujícího model.  
options.k = 2; % (ve skutečnosti je to ovšem 12, viz kód)  
% Počet rozlišení obrázku.  
options.nscale = 1;  
% Vytvoření statistického modelu tvaru, obsahuje informace o pozici  
% bodů.
```

```
[ShapeData TrainingData] = ASM_MakeShapeModel(TrainingData);
% Vytvoření modelu profilů intenzity jasu, obsahuje informace o
% jasových hodnotách kolem umístěných bodů.
AppearanceData = ASM_MakeAppearanceModel(TrainingData,options);
Tyto parametry musí být nastaveny stejně, jak v procesu trénování, tak v procesu
detekce. Výsledkem trénování metody ASM jsou dva .mat soubory, ASM_Shape
Data_X.mat a ASM_AppearanceData_X.mat, kde X označuje počet obrázků, které
byly použity pro trénování.
```

Výpis důležitých parametrů a funkcí použitých k trénování modelu metody AAM následuje:

```
% Počet interpolovaných bodů mezi body definujícími konturu.
options.ni = 2;
% Nastavení normální vzhledové kontury.
options.m = 1;
% Velikost textury definující vzhledový model.
options.texturesize = [100 100];
% Počet rozlišení obrázku.
options.nscales = 1;
% Vytvoření statistického modelu tvaru, obsahuje informace o pozici
% bodů.
[ShapeData TrainingData] = ASM_MakeShapeModel(TrainingData);
% Vytvoření modelu profilů intenzity jasu, obsahuje informace o
% jasových hodnotách kolem umístěných bodů.
AppearanceData = ASM_MakeAppearanceModel(TrainingData,options);
% Vytvoření jednoho celkového vzhledového modelu.
ShapeAppearanceData = AAM_CombineShapeAppearance(TrainingData,Shape
Data,AppearanceData,options);
Parametry i zde musí být stejné, jak v procesu trénování, tak v procesu detekce.
Výsledkem trénování metody AAM je jeden .mat soubor, AAM_Data_X.mat, kde X
označuje počet obrázků, které byly použity pro trénování.
```

Rychlost trénování je velmi závislá na použitém HW. Modely pro techniku ASM lze bez problémů natrénovat na dnes již zastaralé sestavě s procesorem Mobile AMD Sempron (3400+ MHz), 1GB RAM. Trénování modelů pro 75 obrázků trvá na této sestavě asi 10 s. Model pro techniku AAM už ovšem na této sestavě natrénovat nelze, protože limitujícím faktorem je zde operační paměť, která je pouze 1GB. Trénování modelu pro tuto techniku bylo tedy provedeno na jiné sestavě s procesorem Intel Core i5 (3,6 GHz), 4GB RAM. Trénování modelu ze 75 obrázků trvá na této sestavě kolem 6 minut a celková použitá paměť je něco málo přes 3GB RAM.

## 4.2 Implementace detektoru

Detekční systém byl vytvořen v 32-bitové verzi programu Matlab 7.7.0.471 (R2008b) na sestavě s následující konfigurací: Mobile AMD Sempron (3400+ MHz), 1GB RAM, Windows XP Professional se Service Packem 3. K činnosti využívá funkce představené v následujících kapitolách, které jsou upraveny pro potřeby detekčního systému způsobem, jaký bude popsán, nemohou proto být použity přímo z webových stránek MatlabCentral. Při tvorbě detektoru bylo čerpáno z příruček [6] [7] o programu Matlab.

## 4.3 Detekce obličeje

Detekční systém používá k určení polohy obličeje dvou metod představených v kapitolách 3.1 a 3.2. Každá má svá specifika a z nich plynoucí výhody a nevýhody. Metoda detekce Viola-Jones je použitelná pro detekci více obličejů ve vstupním obrázku. Je ovšem potřebné ji natrénovat (viz kapitola 4.1.2) a je také pomalejší oproti metodě VH [10], kterou není před prvním použitím třeba trénovat. Tato metoda je ovšem na druhou stranu použitelná k detekci jen jednoho obličeje ve vstupním obrázku. Výhoda možnosti detekce více obličejů u metody VJ je ovšem v našem případě nevyužita, protože detekční systém je navržen tak, aby zpracovával pouze obrázky, na kterých je zachycena pouze jedna osoba v termografickém spektru. Detekce obličeje je nutná pro co nejpřesnější umístění modelů technik ASM a AAM.

### 4.3.1 Implementace detektoru Viola-Jones

K implementaci tohoto detektoru byl využit toolbox Open CV Viola-Jones Face Detection in Matlab [26], který zprovozňuje funkce pro detekci obličeje (vytvořené v OpenCV) v programu Matlab. V tomto toolboxu je k dispozici zdrojový kód napsaný v C, který využívá funkce OpenCV pro detekci obličeje. Tento zdrojový kód je nutné přeložit pomocí Matlabu na *.mexw32* soubor (v našem případě se jedná o *FaceDetect.mexw32*, jejímž vstupem je kaskáda *haarcascade\_th\_cattivite.xml* natrénovaná v kapitole 4.1.2), se kterým umí pracovat. Vlastní detekce je zajištěna funkcí psanou v Matlabu, která je k dispozici na přiloženém DVD, a která je vypsána na dalším řádku:

```
function [face rect face_pos] = extract_face_from_vii(src, ratio);
```

Tato funkce má dva vstupy a tři výstupy, je ovšem využíván jen jeden vstup a jeden výstup. Ze vstupů je využíván jen *src*, který definuje vstupní obrázek, ve kterém má probíhat detekce. Z výstupů je využíván jen *face\_pos*, který představuje pozici, na

které se nachází obličej v daném obrázku. Ukázky detekce pomocí tohoto detektoru jsou zobrazeny na obr. 4.7.



Obr. 4.7: Ukázka detekcí pomocí detektoru Viola-Jones.

### 4.3.2 Implementace detektoru VH projekce

Implementace tohoto detektoru je provedena jako funkce v programu Matlab podle teorie uvedené v kapitole 3.2. Funkce je vypsána na následujícím řádku:

```
function [face, x, y] = extract_face_from_thi(src);
```

Jak je vidět, funkce má jeden vstup a tři výstupy. Vstupem je `src`, který definuje vstupní obrázek, ve kterém má probíhat detekce. Výstupem této funkce jsou body `x,y` definující obdélník, v němž se nachází obličej, i zde tedy nejsou využívány všechny vstupy a výstupy. Funkce je opět dostupná na příloženém DVD. Ukázky detekce pomocí tohoto detektoru jsou zobrazeny na obr. 4.8.



Obr. 4.8: Ukázka detekcí pomocí detektoru VH projekce.

## 4.4 Detekce částí obličeje

Pro detekci částí obličeje bylo opět využito toolboxu představeného v kapitole 4.1.3. Jak již bylo zmíněno, tento toolbox ve své základní verzi slouží pro detekci kontury ruky, bylo tedy nutné jej upravit pro účely detekce obličeje. Vlastní natrénování na částech obličeje v termografickém spektru bylo představeno v kapitole 4.1.4, výstupy trénování jsou využity v této fázi pro vlastní detekci.

Změny provedené oproti původní verzi se týkají především zautomatizování celého systému, kdy není nutná interakce uživatele během provádění programu. V původní verzi je potřeba umístit natrénovaný model pomocí myši, což přináší dobré výsledky, ovšem za cenu nutného zásahu ze strany uživatele. Další velkou nevýhodou je nemožnost úpravy prostorové velikosti modelu pro detekci, jeho rozměry totiž odpovídají rozměrům trénovacích vzorů. Další změny lze nalézt ve zdrojovém kódu, který je k dispozici na přiloženém DVD.

Nedostatky uvedené v předchozím odstavci jsou odstraněny následujícím způsobem:

1. K umisťování modelů technik ASM i AAM se využívá souřadnic, které jsou výstupem detekce obličeje. Tyto souřadnice definují polohu obličeje v daném vstupním obrázku a s výhodou jich můžeme použít pro umístění modelů. Model obou technik je umístěn přesně doprostřed detekované oblasti (ve vertikálním i horizontálním směru), kde se předpokládá výskyt nosu a očí. Střed modelu tedy teoreticky splývá s geometrickým středem obličeje, kde se u většiny lidí nachází kořen nosu. Důležité je zde slovo teoreticky, protože samozřejmě záleží na přesné detekci obličeje. V kapitole 5 je uvedeno v kolika správně detekovaných obličejích z databáze jsou také správně detekovány jeho části.
2. Velikost modelu je dynamicky řízena podle velikosti detekovaného obličeje ve vstupním obrázku. Testováním byla zvolena proměnná *master\_scale* odpovídající hodnotě *sirka\_obliceje/110*, pomocí níž se následně upravuje vertikální i horizontální velikost modelu.

Zde je ještě potřeba zdůraznit, že detekce částí obličeje byla podřízena co nejvyšší rychlosti při zachování dobrého detekčního poměru z důvodu možnosti budoucího rozšíření celého detekčního systému o možnost detekce v reálném čase, což je ovšem nad rámec této práce.

### 4.4.1 Implementace techniky ASM

Implementace této techniky je provedena jako funkce v programu Matlab a je možné ji nalézt v toolboxu z kapitoly 4.1.3. Byla ovšem doplněna o některé vstupní parametry, které jsou popsány výše. Výpis funkce je na následujících řádcích:

```
[posx_final, posy_final] = ASM_ApplyModel(obr, tform, ShapeData,  
AppearanceData, options, master_scale, obdelnik);
```

Jak je vidět, funkce má sedm vstupů a dva výstupy.

### Vstupy funkce

`obr` – definuje vstupní obrázek, ve které má probíhat detekce

`tform` – definuje počáteční pozici modelu

`ShapeData` – natrénovaný statistický model tvaru

`AppearanceData` – natrénovaný model profilů intenzity jasu

`options` – viz dále

`master_scale` – definuje proměnnou pro úpravu velikosti modelu tvaru

`obdelnik` – definuje polohu obličeje ve vstupním obrázku

### Výstupy funkce

`posx_final` a `posy_final` – definují polohu nalezených částí obličeje

Výpis důležitých parametrů detekce techniky ASM (parametr `options`):

```
% Délka profilu intenzit každého bodu definujícího model.
```

```
options.k = 2; % (ve skutečnosti je to ovšem 12, viz kód)
```

```
% Prohledávací délka (v pixelech).
```

```
options.ns = floor(4*master_scale);
```

```
% Počet rozlišení obrázku.
```

```
options.nscale = 1;
```

```
% Počet iterací prohledávání.
```

```
options.nsearch = 10;
```

Ukázka detekce částí obličeje touto metodou je zobrazena na obrázku 4.9.



Obr. 4.9: Ukázka detekovaných částí obličeje technikou ASM.



## 4.4.2 Implementace techniky AAM

Implementace této techniky je opět provedena jako funkce v programu Matlab a je možné ji nalézt v toolboxu z kapitoly 4.1.3, byla ovšem doplněna o některé vstupní parametry, které jsou popsány výše. Výpis funkce je na následujících řádcích:

```
[posx_final, posy_final] = AAM_ApplyModel(obr, tform, Data, options,  
master_scale, obdelnik);
```

Jak je vidět, funkce má šest vstupů a dva výstupy.

### Vstupy funkce

`obr` – definuje vstupní obrázek, ve které má probíhat detekce

`tform` – definuje počáteční pozici modelu

`Data` – natrénovaný vzhledový model

`options` – viz dále

`master_scale` – definuje proměnnou pro úpravu velikosti modelu tvaru

`obdelnik` – definuje polohu obličeje ve vstupním obrázku

### Výstupy funkce

`posx_final` a `posy_final` – body definující polohu nalezených částí obličeje.

Výpis důležitých parametrů detekce techniky AAM (parametr *options*):

```
% Velikost textury modelu vzhledu.
```

```
options.texturesize = [100 100];
```

```
% Počet rozlišení obrázku.
```

```
options.nscales = 1;
```

```
% Počet iterací prohledávání.
```

```
options.nsearch = 10;
```

Ukázka detekce částí obličeje touto metodou je zobrazena na obrázku 4.10.



Obr. 4.10: Ukázka detekovaných částí obličeje technikou AAM.

## 4.5 Popis ovládání detekčního systému

Detekční systém je koncipován jako funkce, která byla vytvořena v programu Matlab a je popsána v kapitole 4.5.1. Jako nadstavba nad touto funkcí bylo dále vytvořeno GUI, které slouží pro snadnější ovládání celého detekčního systému, viz kapitola 4.5.2. Na přiloženém DVD jsou k dispozici obě tato řešení ve svých vlastních složkách, protože pro potřeby GUI byla funkce mírně upravena. V následujících podkapitolách bude popsáno, jak tato dvě řešení správně používat. Jedná se především o popsání vstupů funkce, protože ovládání GUI je vesměs intuitivní, pro úplnost je ovšem také popsáno.

### 4.5.1 Funkce

Funkce vypsaná na následujících dvou řádcích:

```
function [obdelnik body] = detekovat_body(obr,detekce_obliceje,  
detekce_casti, teplota1,teplota2),
```

představuje vlastní detekční systém a sdružuje v sobě funkce uvedené v předchozích kapitolách. Z jejího výpisu je patrné, že má pět vstupů a dva výstupy, které budou následně popsány.

#### Vstupy funkce

**obr** – vstupní šedotónový obrázek, ve kterém chceme detekci provádět (zadat např. 'obrazek.jpg')

**detekce\_obliceje** – vstupem je textový řetězec 'vj' pro detekci obličeje metodou Viola-Jones nebo textový řetězec 'vh' pro detekci obličeje metodou vertikální a horizontální projekce

**detekce\_casti** – vstupem je textový řetězec 'asm' pro detekci částí obličeje technikou Active Shape Models nebo textový řetězec 'aam' pro detekci částí obličeje technikou Active Appearance Models

**teplota1** – číselná hodnota definující dolní hranici teplot, ve kterých byl obrázek v termografickém spektru pořízen

**teplota2** – číselná hodnota definující horní hranici teplot, ve kterých byl obrázek v termografickém spektru pořízen

#### Výstupy funkce

**obdelnik** – vektor souřadnic definující polohu obličeje v obrázku

**body** – matice souřadnic definující polohu očí a nosu, dále jasovou hodnotu pixelu na daných souřadnicích a jim odpovídající teplotu

Detailnější informace lze nalézt po zadání příkazu:  
`help detekovat_body`

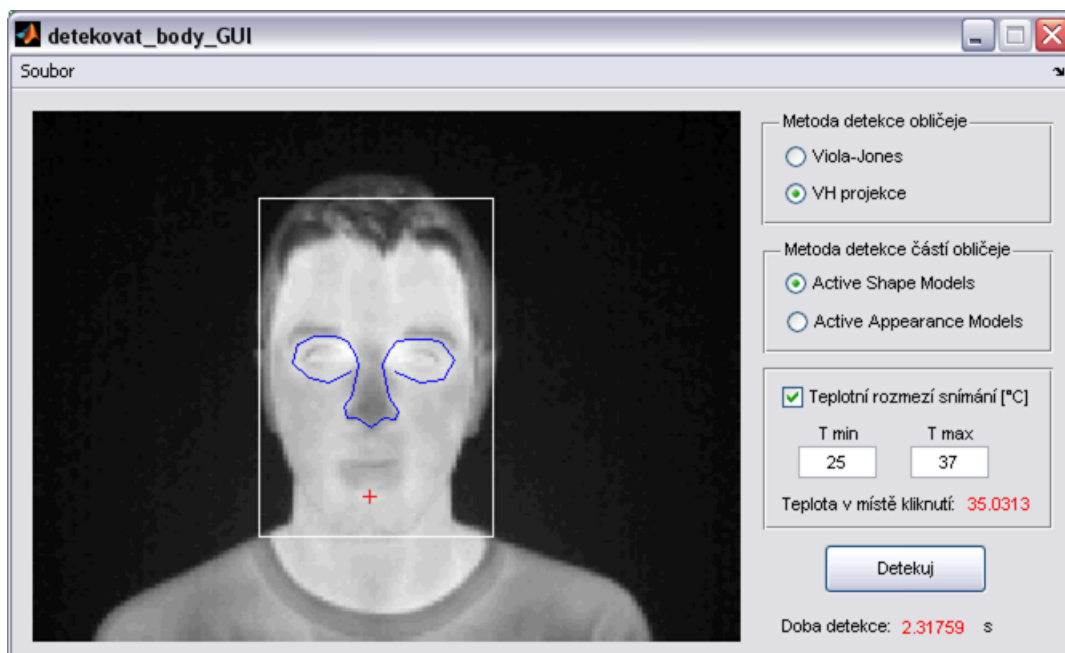
## 4.5.2 GUI

GUI bylo vytvořeno za účelem příznivější možnosti ovládání pro uživatele, využívá služby funkce přiblížené v kapitole 4.5.1, kterou ovšem ještě doplňuje o možnosti získání teploty jakéhokoliv bodu v obrázku a nejen teploty bodů definujících nalezenou konturu. GUI je možné spouštět pomocí skriptu *detekovat\_body\_GUI.m* a je znázorněno na obr. 4.11. Hlavní předností tohoto řešení je jednoduché a snadné ovládání narozdíl od používání funkce představené v předchozí kapitole.

Načtení vstupního obrázku se provede volbou Otevřít obrázek z nabídky Soubor, popř. klávesovou zkratkou Ctrl+O. Ve stejné nabídce je ještě možnost Uložit obrázek (klávesová zkratka Ctrl+S) a Konec, pro ukončení programu (klávesová zkratka Ctrl+X). Podporované formáty obrázků pro načítání a ukládání jsou BMP, JPG nebo PNG.

V pravé části okna jsou dále možnosti výběru detekce obličeje (Viola-Jones nebo VH projekce) a možnosti výběru detekce částí obličeje (Active Shape Models nebo Active Appearance Models). Dále je zde možnost měření teploty ve vstupním obrázku pomocí ukazatele myši, ta se ovšem aktivuje až po načtení obrázku a aktivování této možnosti pomocí zatržení příslušného políčka Teplotní rozmezí snímání [°C]. Do příslušných políček je potřeba zadat rozmezí teplot, ve kterých byl vstupní obrázek pořízen. Po načtení vstupního obrázku se rovněž aktivuje možnost spuštění detekce. V pravé části okna se po skončení detekce, popř. po změření teploty dále zobrazují červeně zvýrazněné naměřené hodnoty.

Zdrojové kódy GUI je možné si opět prohlédnout na přiloženém DVD.



Obr. 4.11: Ukázka grafického uživatelského prostředí.

## 5 TESTOVÁNÍ

Detekční systém byl testován v 32-bitové verzi programu Matlab 7.7.0.471 (R2008b) na sestavě s následující konfigurací: Intel Core2 Duo E7200(2,53 GHz), 2GB RAM, Windows XP Professional ve verzi 2002 se Service Packem 3.

Než se přistoupí k samotnému testování, je nutné specifikovat některá označení a definovat, co je to správná/špatná detekce obličeje a co je to správná/špatná detekce částí obličeje. Při hodnocení detekčních schopností systému zavedeme dva pojmy a těmi jsou  $SDR_F$  (Successfull Detection Rate Face) a  $SDR_{PF}$  (Successfull Detection Rate Parts of Face), kde

$$SDR_F = 100 \left( \frac{N_{CF}}{N_A} \right) [\%] \quad (5.1)$$

a

$$SDR_{PF} = 100 \left( \frac{N_{CPF}}{N_A} \right) [\%]. \quad (5.2)$$

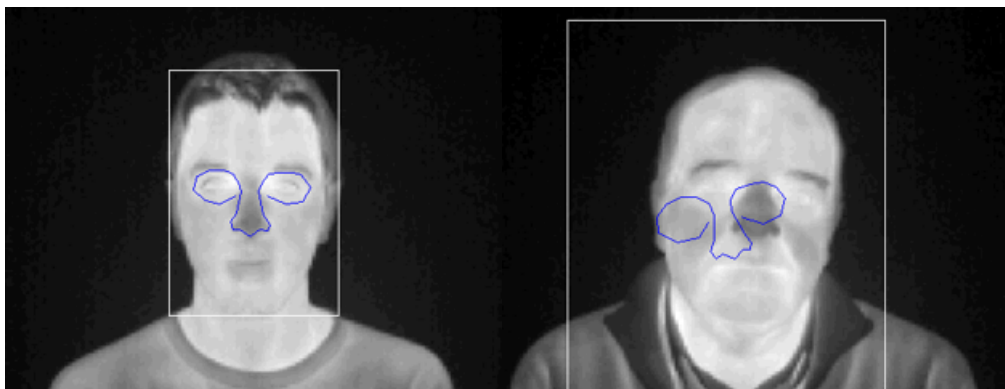
$N_{CF}$  je počet správně detekovaných obličejů,  $N_{CPF}$  je počet správně detekovaných částí obličejů a  $N_A$  je celkový počet všech obličejů z testovací databáze, která obsahuje celkem 840 obrázků, a proto  $N_A = 840$ .

Kritéria správné detekce obličeje byla stanovena subjektivně a to tak, že správně detekovaný obličej musí obsahovat minimálně oboje obočí, obě oči a rty, maximálně pak může obsahovat nahoře vlasy, dole část krku a na bocích jsou hranicí ušní boltce, popř. konečky vlasů, nepřesahují-li o mnoho obličej. Ukázky správné a špatné detekce obličeje jsou znázorněny na obrázku 5.1.



Obr. 5.1: Ukázka správné a špatné detekce obličeje.

Kritéria správné detekce částí obličeje byla stanovena také subjektivně a to tak, že za správnou detekci je považován stav, kdy model označuje obě očníce alespoň tak, že se uvnitř modelu nachází minimálně obě oči a špička nosu. Maximálně je potom za správnou detekci považován stav, kdy se uvnitř modelu nachází i obočí a spodní hranice nosu s malou částí horního rtu. Ukázky správné a špatné detekce částí obličeje jsou znázorněny na obrázku 5.2.

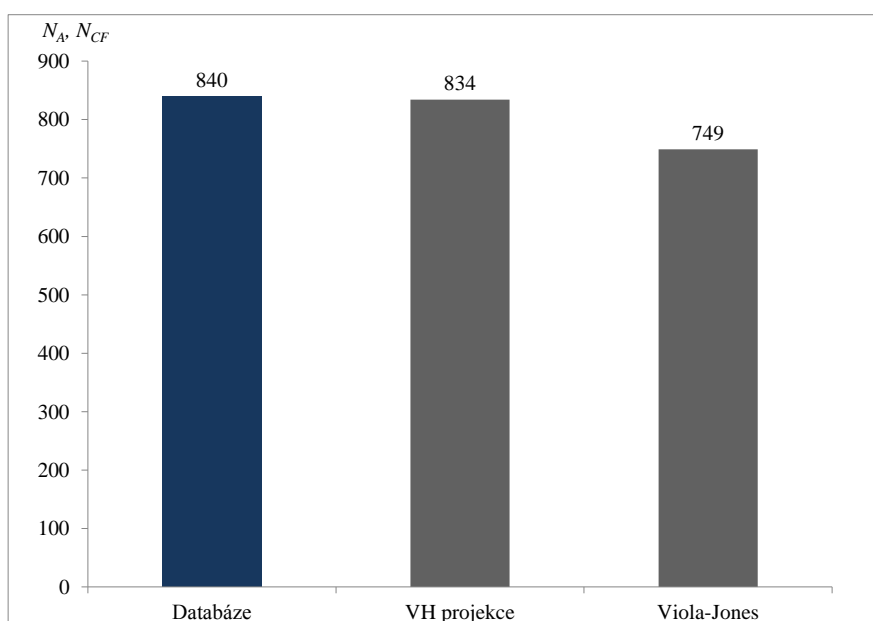


Obr. 5.2: Ukázka správně a špatně detekovaných částí obličeje.

Počty správně detekovaných obličejů pro obě techniky detekce obličeje jsou uvedeny v tabulce 5.1. Tyto hodnoty jsou následně vyneseny do grafu na obrázku 5.3 pro jejich porovnání.

Tab. 5.1: Porovnání úspěšnosti detekce technik VH projekce a Viola-Jones.

–	$N_{CF}$	$SDR_F$
VH projekce	834	99%
Viola-Jones	749	89%



Obr. 5.3: Porovnání úspěšnosti metod detekce obličeje.

Nyní již lze přistoupit k vlastnímu testování metod detekce částí obličeje, které byly provedeny pro různé počty trénovacích obrázků, testování bylo provedeno pro

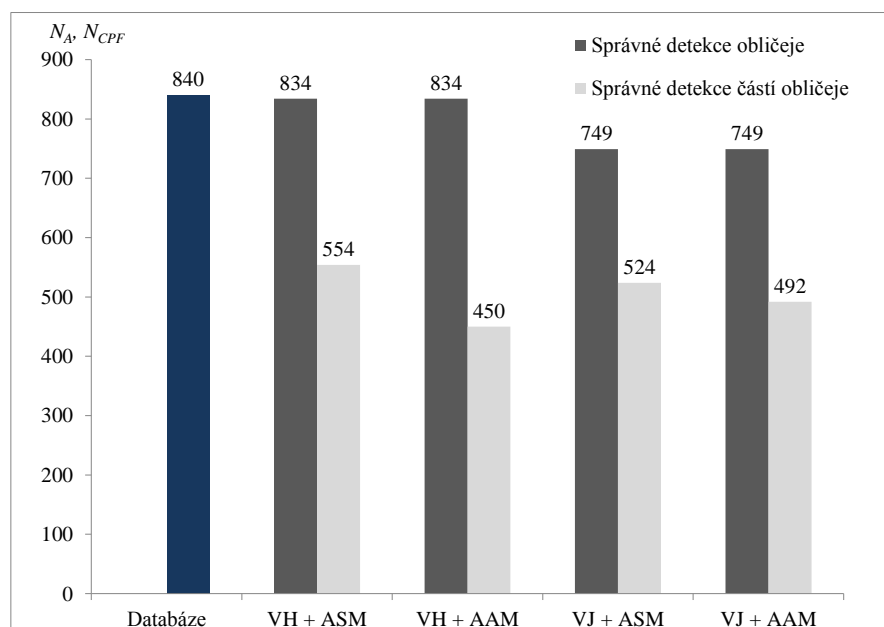
modely natrénované z 25, 50 a 75 obrázků pro obě techniky (ASM i AAM), nastavení metod viz kapitoly 4.4.1 a 4.4.2. Pro všechny modely byly provedeny série testů v různých kombinacích detektoru obličeje a metody detekce částí obličeje.

## 5.1 Dosažené výsledky pro modely natrénované z 25 obrázků

Dosažené výsledky pro různá nastavení detekčního systému jsou přehledně uvedeny v tabulce 5.2 a následně vyneseny do grafu 5.4.

Tab. 5.2: Výsledky pro různé kombinace nastavení detekčního systému pro 25 trénovacích obrázků.

–	$N_{CPF}$	$SDR_{PF}$	Prům. čas detekce
VH projekce + Active Shape Models	554	66 %	0,5941 s
VH projekce + Active Appearance Models	450	54 %	0,9774 s
Viola-Jones + Active Shape Models	524	63 %	0,7096 s
Viola-Jones + Active Appearance Models	492	59 %	1,0592 s



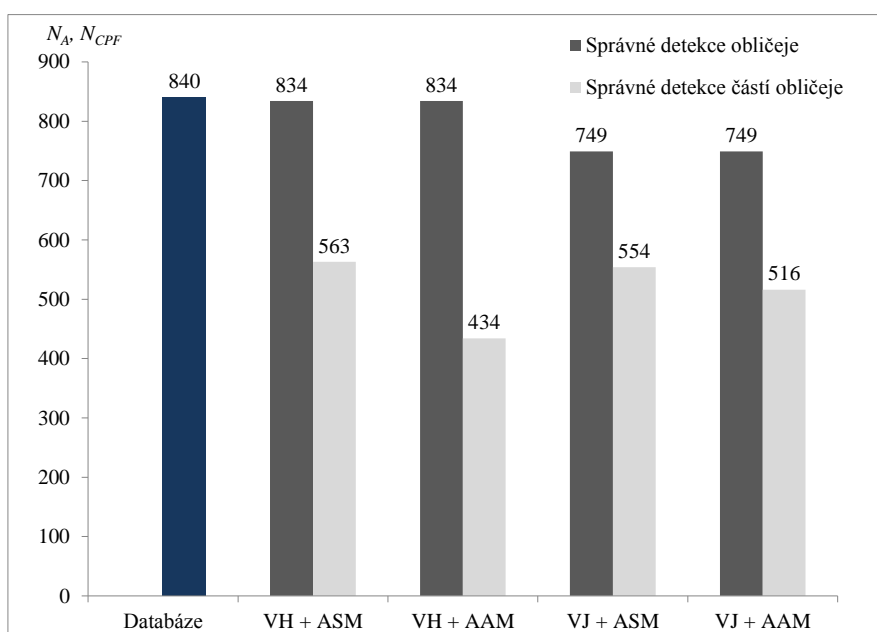
Obr. 5.4: Porovnání úspěšnosti metod detekce částí obličeje pro 25 trénovacích obrázků.

## 5.2 Dosažené výsledky pro modely natrénované z 50 obrázků

Dosažené výsledky pro různá nastavení detekčního systému jsou přehledně uvedeny v tabulce 5.3 a následně vyneseny do grafu na obrázku 5.5.

Tab. 5.3: Výsledky pro různé kombinace nastavení detekčního systému 50 trénovacích obrázků.

–	$N_{CPF}$	$SDR_{PF}$	Prům. čas detekce
VH projekce + Active Shape Models	563	67 %	0,5777 s
VH projekce + Active Appearance Models	434	52 %	0,9954 s
Viola-Jones + Active Shape Models	554	66 %	0,6607 s
Viola-Jones + Active Appearance Models	516	62 %	1,0664 s



Obr. 5.5: Porovnání úspěšnosti metod detekce částí obličeje pro 50 trénovacích obrázků.

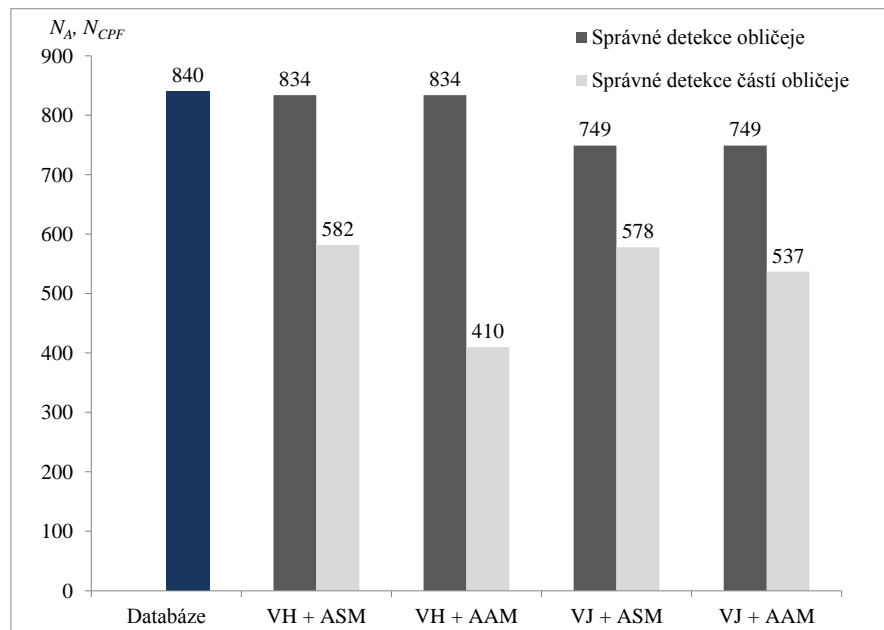
## 5.3 Dosažené výsledky pro modely natrénované ze 75 obrázků

Dosažené výsledky pro různá nastavení detekčního systému jsou přehledně uvedeny v tabulce 5.4 a následně vyneseny do grafu na obrázku 5.6.



Tab. 5.4: Výsledky pro různé kombinace nastavení detekčního systému 75 trénovacích obrázků.

–	$N_{CPF}$	$SDR_{PF}$	Prům. čas detekce
VH projekce + Active Shape Models	582	69 %	0,5669 s
VH projekce + Active Appearance Models	410	49 %	1,0525 s
Viola-Jones + Active Shape Models	578	69 %	0,6575 s
Viola-Jones + Active Appearance Models	537	64 %	1,1111 s



Obr. 5.6: Porovnání úspěšnosti metod detekce částí obličeje pro 75 trénovacích obrázků.

## 5.4 Vylepšený model

Jak vyplývá z předchozích grafů, účinnost detekce není příliš vysoká, což je způsobeno snahou dosáhnout co nejnižší detekční doby. Byla tedy vybrána nejlepší kombinace metody detekce obličeje a metody detekce částí obličeje, která při zachování nejnižší detekční doby podávala nejlepší výsledky (tato kombinace odpovídá metodě VH projekce v kombinaci s modely ASM, viz kapitola 5.3). Byly natrénovány další modely pro techniku ASM, nyní už ovšem nešlo o dosažení co nejnižší detekční doby, ale o dosažení lepších výsledků detekce, přičemž pro výběr parametrů pro trénování bylo vycházeno z literatury [5]. Na následujících řádcích je uveden výpis důležitých parametrů a funkcí použitých k trénování vylepšených modelů metody ASM:

% Počet interpolovaných bodů mezi body definujícími konturu.

```
options.ni = 2;
```

```

% Délka profilu intenzit každého bodu definujícího model.
options.k = 3; % (ve skutečnosti je to ovšem 13, viz kód)
% Počet rozlišení obrázku.
options.nsamples = 3;

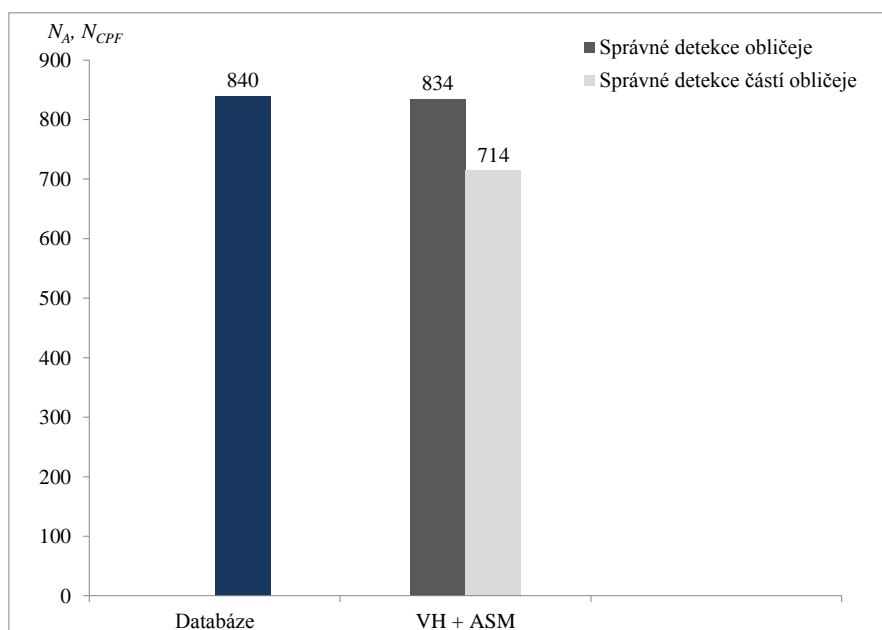
```

Tyto parametry musí být opět nastaveny stejně, jak v procesu trénování, tak v procesu detekce. Výsledkem trénování metody ASM jsou dva *.mat* soubory, *ASM\_ShapeData\_75\_improved.mat* a *ASM\_AppearanceData\_75\_improved.mat*, kde číslo 75 označuje počet obrázků, které byly použity pro trénování.

Dosažené výsledky pro vylepšený model jsou přehledně uvedeny v tabulce 5.5 a následně vyneseny do grafu na obrázku 5.7.

Tab. 5.5: Výsledky pro vylepšený model.

–	$N_{CPF}$	$SDR_{PF}$	Prům. čas detekce
VH projekce + Active Shape Models	714	85 %	0,9951 s



Obr. 5.7: Úspěšnost detekce vylepšeného modelu.

## 6 ZÁVĚR

Cílem diplomové práce byl rozbor aktuální problematiky detekce částí obličeje v termografickém spektru a následný návrh systému, který prováděl detekci očí a nosu, detekce úst byla z detekčního systému vyloučena z důvodů popsaných v kapitole 4.1.4. Navržený a realizovaný systém byl založen na algoritmu Viola-Jones, algoritmu vertikální a horizontální projekce a na modelech ASM/AAM.

V první kapitole je pojednáno o zpracování obrazových signálů – obraz a jeho reprezentace, barevné modely a histogram. Druhá kapitola je věnována obrazům a obličeji v termografickém spektru. Třetí kapitola se věnuje detekci obličeje a jeho částí – detektor Viola-Jones, VH projekce, ASM a AAM. Čtvrtá kapitola je věnována praktické realizaci detekčního systému – trénování metod, implementace v programu Matlab a popis ovládání. Pátá kapitola se zabývá testováním navrženého detektoru a porovnáním výsledků pro různá nastavení a kombinace metod detekce obličeje a částí obličeje.

Při testování detekčního systému bylo potřeba manuálně projít přes 10 000 obrázků a zhodnotit, jestli jsou dané části obličeje detekovány podle požadavků stanovených v kapitole 5. Bylo zjištěno, že lepších výsledků dosahuje metoda ASM oproti metodě AAM, která pro dobrou konvergenci vyžaduje velmi přesné umístění počátečního modelu do obrazu. Dalším zjištěním bylo, že s rostoucím počtem trénovacích dat roste i úspěšnost detekce. Nejlepších výsledků potom dosahovala metoda ASM v kombinaci s algoritmem VH projekce, tato kombinace byla zároveň i nejrychlejší. Dobrých výsledků dosahovala i kombinace algoritmu Viola-Jones s metodou ASM, na procentuální počet správně detekovaných obličejů byla tato kombinace dokonce lepší než předchozí kombinace, ovšem zároveň byla i o něco pomalejší. Je potřeba zdůraznit, že celková účinnost detekčního systému, který byl koncipován pro co nejvyšší rychlost, byla celkem nízká, průměr všech kombinací metod pro všechny počty natrénovaných modelů je něco málo přes 60 %. Byla proto vybrána nejlepší kombinace metody detekce obličeje a metody detekce částí obličeje (VH projekce + ASM), která byla upravena tak, aby podávala lepší detekční výsledky na úkor rychlosti, celková úspěšnost detekce se tím zvýšila na 85 %.

Nízká detekční schopnost systému je zřejmě zapříčiněna nízkým počtem trénovacích dat pro techniky ASM i AAM, v některých případech také nevýraznou konturou částí obličeje, kdy nejvyšší vliv na špatnou detekci má teplý (tedy v termografickém spektru světlý) nos, další příčinou neúspěšné detekce jsou také vousy, kdy model konverguje například ke kontuře kníru namísto spodní hranici nosu. Negativní vliv na detekci částí obličeje má také špatná detekce vlastního obličeje, kdy je model umístěn příliš nízko (ústa), popř. příliš vysoko (čelo) a již nestihne konvergovat ke správné pozici.

Budoucím vylepšením systému by mohlo být rozšíření metody VH projekce o detekci více obličejů a tedy i možná vícenásobná detekce částí obličeje. Další rozšíření detekčního systému by mohlo být o možnost detekce v reálném čase, k tomu by ovšem bylo potřeba zvýšit účinnost detekce při zachování nebo lépe snížení detekční doby (jedná se o kombinaci VH projekce a modelů ASM). Pro detekci v reálném čase by bylo vhodné zkombinovat nejrychlejší variantu detekce částí obličeje s metodou sledování optického toku nebo odhadem vektoru pohybu ve videu.

## LITERATURA

- [1] BALDOCK, ED., R.; COOTES, T., F.; GRAHAM, J.; *Image Processing and Analysis*. Oxford: Oxford University Press, Model-Based Methods in Analysis of Biomedical Images – An Introduction to Active Shape Models, 2000. s. 223–248.
- [2] BRADSKI, G.; KAEHLER, A.; *Learning OpenCV: Computer Vision with the OpenCV Library*. 1st edition. United States of America: O’Reilly, 2008. 555 s. ISBN 978-0-596-51613-0.
- [3] COOTES, T., F.; EDWARDS, G.; TAYLOR, C., J.; *Active appearance models*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, pp. 681–685, 2001.
- [4] COOTES, T., F.; TAYLOR, C., J.; *Proc. SPIE Medical Imaging*. UK: University of Manchester, Imaging Science and Biomedical Engineering, 2001. 14 s.
- [5] COOTES, T., F.; TAYLOR, C., J.; *Statistical Model of Appearance for Computer Vision*. Technical Report, The University of Manchester School of Medicine, 2004. Dostupné z WWW: <[http://www.isbe.man.ac.uk/bim/Models/app\\_models.pdf](http://www.isbe.man.ac.uk/bim/Models/app_models.pdf)>.
- [6] C. GONZALEZ, R.; E. WOODS, R. *Digital Image Processing*. Second Edition. Upper Saddle River, New Jersey 07458, USA: Prentice-Hall, 2002. 813 s. ISBN 0-201-18075-8.
- [7] C. GONZALEZ, R.; E. WOODS, R.; L. EDDINS, S. *Digital Image Processing Using MATLAB. 2nd ed.* Gatesmark Publishing, 2009. 827 s. ISBN 978-0982085400.
- [8] ČERBA, O. *Http://www.gis.zcu.cz: Barevné modely* [online]. Pracovní verze. Západočeská univerzita : Vydala Fakulta aplikovaných věd, katedra matematiky v Plzni. , 2010, 6.10.2006 [cit. 17.10.2010]. Barvy v počítači a kartografii. Dostupné z WWW: <<http://www.gis.zcu.cz/studium/pok/Materialy/Book/ar03s01.html>>.
- [9] FALAKI, H. *AdaBoost Algorithm*. Los Angeles: University of California, Computer science Department, 2009, 4 s.
- [10] FAUNDEZ-ZANUY, M.; MEKYSKA, J.; ESPINOSA-DURÓ, V. Face segmentation: A comparison between visible and thermal images. In *IEEE International Carnahan Conference on Security Technology (ICCST) 2010*, San Jose, 5–8 Oct. 2010. s. 185–189. ISBN 978-1-4244-7403-5.

- [11] FAUNDEZ-ZANUY, M.; MEKYSKA, J.; ESPINOSA-DURÓ, V. On the focus of thermal images. In *Pattern Recognition Letters* [online]. 23.8.2010 [cit. 25.11.2010]. Dostupné z WWW: <<http://www.sciencedirect.com/science/article/B6V15-52RXV6F-4/2/6daa7894a438d4456ae226e2b13be38d>> ISSN 0167-8655
- [12] FLORIAN A. *OpenCV's Rapid Object Detection: How-To build a cascade of boosted classifiers based on Haar-like features*. 2003. 6 s. Dostupné z WWW: <<http://lab.cntl.kyutech.ac.jp/kobalab/nishida/opencv/OpenCV-ObjectDetection-HowTo.pdf>>.
- [13] GREENSTED, A.; *The Lab Book Pages: Otsu Thresholding* [online]. 2010. 2010, 17th June 2010 [cit. 2010-11-17]. Otsu Thresholding. Dostupné z WWW: <<http://www.labbookpages.co.uk/software/imgProc/otsuThreshold.html>>.
- [14] HAMZA, M. H. *Visualization, Imaging, and Image Processing*. Anaheim: ACTA Press, 2003. 1059 s. ISBN 0-88986-382-2.
- [15] CHIN-YA, H.; MON-JU, W. *Image Segmentation*. Madison: University of Wisconsin, ECE 533 Final Project, 2006. 43 s.
- [16] KOSCHAN. A.; KANG S.; PAIK J.; ABIDI B.; ABIDI M. *Color active shape models for tracking non-rigid objects* in *Pattern Recognition Letters, Special Issue on Color Image Processing*, Vol. 24, pp. 1751–1765, July 2003. Dostupné z WWW: <<http://imaging.utk.edu/publications/papers/2003/koschan-prlip03.pdf>>.
- [17] KROON, D. J. *Active Shape Model (ASM) and Active Appearance Model (AAM)* [online]. Poslední úprava 16.2.2010 [cit. 25.2.2011] Dostupné z WWW: <<http://www.mathworks.com/matlabcentral/fileexchange/26706-active-shape-model-asm-and-active-appearance-model-aam>>.
- [18] LIANG, X. *Image binarization using Otsu method*. CASIA, NLPR-PAL Group, 2009. 22 s.
- [19] MADISETTI, V.; BENNETT WILLIAMS, D. *The digital signal processing handbook*. USA: CRC Press, 1998. 1776 s. ISBN 0-8493-8572-5.
- [20] MILBORROW, S. *Locating Facial Features with Active Shape Models*: Master's thesis. University of Cape Town, Faculty of Engineering, 2007, 103 s. Dostupné z WWW: <<http://www.milbo.users.sonic.net>>.

- [21] MINKINA, W.; DUDZIK, S. *Infrared Thermography: Errors and Uncertainties*. Great Britain: John Wiley and Sons, Ltd, 2009. 222 s. ISBN 978-0-470-74718-6(Hbk).
- [22] NOVÁK, C.L.; SHAFER, S.A. Anatomy of a color histogram. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, USA: Champaign, 15.6.1992–18.6.1992. s.599–605. ISBN 0-8186-2855-3.
- [23] PŘINOSIL, J.; MÍČA, I. *Efektivní detekce významných bodů částí obličeje*. 2010, vyd. 57, 5 s., ISSN 1213-1539
- [24] RAJMIC, P. *Základy počítačové sazby a grafiky: skripta*. Brno: FEI VUT v Brně, 2010. 121 s.
- [25] ŘÍHA, K. *Pokročilé techniky zpracování obrazu: skripta*. Brno: FEI VUT v Brně, 2007. 109 s.
- [26] SREEKAR, K. *Open CV Viola-Jones Face Detection in Matlab* [online]. Poslední úprava 21.5.2008 [cit. 20.2.2011] Dostupné z WWW: <<http://www.mathworks.com/matlabcentral/fileexchange/19912-open-cv-viola-jones-face-detection-in-matlab>>.
- [27] ŠPANĚL, M.; BERAN, V. [www.fit.vutbr.cz](http://www.fit.vutbr.cz) [online]. FIT VUT v Brně: 12.10.2005, 19.1.2006 [cit. 26.11.2010]. *Obrazové segmentační techniky*. Dostupné z WWW: <<http://www.fit.vutbr.cz/spanel/segmentace/>>.
- [28] VIOLA, P.; JONES, M. *Robust Real-time Object Detection*. Second international workshop on statistical and computational theories of vision — modeling, learning, computing, and sampling, 2001, Vancouver, Canada 25 s.
- [29] YANG, M. H.; KRIEGMAN, D. J.; AHUJA, N. *Detecting Faces in Images: A Survey*. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), 2002. vol. 24. 25 s.
- [30] *Open Source Computer Vision Library* [online]. Poslední úprava 4.1.2011 [cit. 1.5.2011]. Dostupné z WWW: <<http://opencv.willowgarage.com/wiki/>>.
- [31] *Open Source Computer Vision Library: Reference Manual* [online]. U.S.A: Intel Corporation, 2010 [cit. 1.5.2011]. Dostupné z WWW: <[cs.unc.edu/Research/stc/FAQs/OpenCV/OpenCVReferenceManual.pdf](http://cs.unc.edu/Research/stc/FAQs/OpenCV/OpenCVReferenceManual.pdf)>.
- [32] [Http://www.infraredtraining.com](http://www.infraredtraining.com) [online]. Upravené vydání. [U.S.]: FLIR Systems, 2010 [cit. 29.10.2010]. IR Thermography Primer. Dostupné z WWW: <[http://www.infraredtraining.com/ir\\_primer.asp](http://www.infraredtraining.com/ir_primer.asp)>.

## SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

2D	Two Dimensional – dvojdimenzionální prostor
AAM	Active Appearance Models – aktivní modely vzhledu
AMD	Advanced Micro Devices – společnost vyrábějící počítačové komponenty
ASM	Active Shape Models – aktivní modely tvaru
BSD	Berkeley Software Distribution – licence pro svobodný software
CMY	Cyan Magenta Yellow – barevný model (purpurová, azurová, žlutá)
CMYK	Cyan Magenta Yellow black – barevný model (purpurová, azurová, žlutá, černá)
FIR	Far Infrared – vzdálené infračervené pásmo
GB	Gigabyte – označení množství dat v informatice
GPU	Graphics Processing Unit – grafický procesor
GUI	Graphical User Interface – grafické uživatelské rozhraní
HLS	Hue Lightness Saturation – barevný model
HSB	Hue Saturation Brightness – barevný model
HSV	Hue Saturation Value – barevný model
HW	Hardware – pevné části počítače
IPP	Integrated Performance Primitives – knihovna počítačového vidění od společnosti Intel, komerční alternativa k OpenCV
IR	Infrared – infračervené
JPEG	Joint Photographic Experts Group – standard komprese obrazu
LWIR	Long Wave Infrared – infračervené subpásmo pásma TIR
MIR	Medium Infrared – střední infračervené pásmo
MLL	Machine Learning Library – knihovna pro strojové učení OpenCV
MWIR	Mid Wave Infrared – infračervené subpásmo pásma TIR



NIR	Near Infrared – blízké infračervené pásmo
PAL	norma pro televizní vysílání
PCA	Principal Component Analysis – analýza hlavních komponent
RAM	Random Access Memory – paměť s náhodným přístupem
RGB	Red Green Blue – barevný model (červená, zelená, modrá)
RGBA	Red Green Blue Alpha – barevný model (červená, zelená, modrá, alfa)
SECAM	norma pro televizní vysílání
<i>sgn</i>	funkce signum
TIR	Thermal Infrared – tepelné infračervené pásmo
UV	Ultra Violet – ultrafialové pásmo viditelného spektra
XML	Extensible Markup Language – značkovací jazyk
YCbCr	barevný model (Y – jasová složka; Cb, Cr – barvonosné)
YUV	barevný model (Y – jasová složka; U, V – barvonosné)
VH	vertikální horizontální
$\alpha$	váha klasifikátoru algoritmu AdaBoost
$\beta$	barva vícerozměrného signálu
$\mu_0$	průměrná hodnota intenzity popředí daného obrazu
$\mu_B$	průměrná hodnota intenzity pozadí daného obrazu
$\Phi$	kovariační matice všech trénovacích modelů PCA
$\sigma^2$	celkový rozptyl pixelů daného obrazu
$\sigma_0^2$	rozptyl pixelů popředí daného obrazu
$\sigma_\beta^2$	rozptyl pixelů pozadí daného obrazu
$\sigma_{mr}^2$	mezi-rozptyl Otsuho metody
$\sigma_{vr}^2$	vnitřní rozptyl Otsuho metody

$\theta$	úhel, který řídí rotaci modelu tvaru
$\mathbf{A}$	obecná matice definující spojitý nebo digitální obraz
$\mathbf{b}$	deformační model metody ASM
$H$	klasifikátor algoritmu AdaBoost
$ii$	integrální obraz
$i$	originální obraz
$m$	počet trénovacích vzorků
$n_k$	počet pixelů šedotónového obrazu mající úroveň $r_k$
$p(r_k)$	pravděpodobnost výskytu úrovně šedi $r_k$
$\mathbf{R}$	podobrazy vzniklé segmentací
$r_k$	$k$ -tá úroveň šedi šedotónového obrazu
$t$	čas
$T$	rozhodovací práh VH projekce
$\bar{\mathbf{x}}$	průměrný model tvaru metody ASM
$\mathbf{x}_j$	vektor definující $m$ trénovacích vzorků metody PCA
$x_1, x_2$	hranice horizontálního profilu VH projekce
$y_1, y_2$	hranice vertikálního profilu VH projekce
$z$	hloubka vícerozměrného signálu

## A OBSAH PŘILOŽENÉHO DVD

Přiložené DVD obsahuje zdrojové kódy GUI, které bylo představeno v kapitole 4.5.2 a zdrojové kódy funkce představené v kapitole 4.5.1. Dále obsahuje testovací databázi, natrénované modely, výsledky a elektronickou verzi práce ve formátu *.pdf*.

GUI lze spustit po namapování adresáře `\GUI\` v programu Matlab a zadání příkazu `detekovat_body_GUI`. Ovládání je popsáno v kapitole 4.5.2.

Funkci lze spustit po namapování adresáře `\Funkce\` v programu Matlab a zadání příkazu `[obdelnik body] = detekovat_body(obr,detekce_obliceje,detekce_casti,teplota1,teplota2)` s příslušnými parametry popsány v kapitole 4.5.1, nápovědu k funkci lze vyvolat příkazem `help detekovat_body`.

Jestliže dojde k jakékoliv chybě při vykonávání, bude to zřejmě způsobeno verzí programu Matlab, systém byl programován a testován na verzi popsané v kapitole 4.2. Další možnou příčinou může být fakt, že *.merw32* soubor pro detektor Viola-Jones byl kompilován pro 32-bitový operační systém.

Dále následuje výpis důležitých adresářů a jejich obsah:

- *Funkce* – adresář obsahující zdrojové kódy k funkci představující detekční systém, několik testovacích obrázků a natrénované modely
- *GUI* – adresář obsahující zdrojové kódy GUI, několik testovacích obrázků a natrénované modely
- *Natrénované modely* – adresář obsahující všechny natrénované modely používané v práci
- *Testovací databáze* – adresář obsahující pouze testovací část celé databáze
- *Text* – adresář obsahující vlastní text práce ve formátu *.pdf*
- *Trénování* – adresář obsahující použité trénovací obrázky pro techniky ASM i AAM a skripty použité k trénování
- *Výsledky* – adresář obsahující výsledky testování a *.xls* soubor s dobami detekce