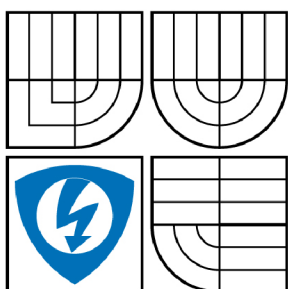




VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH  
TECHNOLOGIÍ

ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF TELECOMMUNICATIONS

# AUTOMATICKÉ DETEKCE OBLIČEJE A JEHO JEDNOTLIVÝCH ČÁSTÍ

AUTOMATIC FACE AND FACIAL FEATURE DETECTION

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

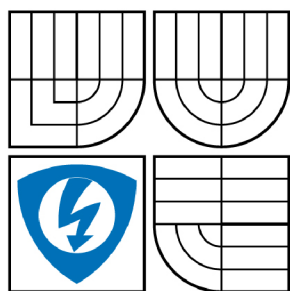
Bc. MARTIN KROLIKOWSKI

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. JIŘÍ PŘINOSIL

BRNO 2008



VYSOKÉ UČENÍ  
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

Ústav telekomunikací

## Diplomová práce

magisterský navazující studijní obor  
Telekomunikační a informační technika

**Student:** Krolikowski Martin Bc.

**ID:** 89155

**Ročník:** 2

**Akademický rok:** 2007/2008

### NÁZEV TÉMATU:

**Automatické detekce obličeje a jeho jednotlivých částí**

### POKYNY PRO VYPRACOVÁNÍ:

Prostudujte metody detekce obličeje ze statických obrazů. Nejvhodnější metodu z hlediska výpočetní náročnosti, přesnosti a robustnosti realizujte a zhodnoťte dosažené výsledky. Na základě experimentů proveďte návrh a implementaci vlastního detektoru jednotlivých částí obličeje (oči a rty). Testování míry spolehlivosti obou realizovaných detektorů proveďte na některé veřejně dostupné obrazové databázi tváří.

### DOPORUČENÁ LITERATURA:

[1] Viola, P. Jones, M. Robust Real-time Object Detection, International Journal of Computer Vision 2002, ISSN 0920-5691, 2002.

[2] Yang, M. Kriegman, D. Ahuja, N. Detecting Faces in Images: A Survey, IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), vol. 24, no. 1, pp. 34-58, 2002.

**Termín zadání:** 11.2.2008

**Termín odevzdání:** 28.5.2008

**Vedoucí práce:** Ing. Jiří Přinosil

**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 § 152 trestního zákona č. 140/1961 Sb.

# LICENČNÍ SMLOUVA

## POSKYTOVANÁ K VÝKONU PRÁVA UŽÍT ŠKOLNÍ DÍLO

uzavřená mezi smluvními stranami:

### 1. Pan/paní

Jméno a příjmení: Bc. Martin Krolikowski  
Bytem: Žižkova 997/74, 73301, Karviná - Fryštát  
Narozen/a (datum a místo): 8.7.1984, Karviná

(dále jen "autor")

a

### 2. Vysoké učení technické v Brně

Fakulta elektrotechniky a komunikačních technologií  
se sídlem Údolní 244/53, 60200 Brno 2  
jejímž jménem jedná na základě písemného pověření děkanem fakulty:  
prof. Ing. Kamil Vrba, CSc.

(dále jen "nabyvatel")

## Článek 1

### Specifikace školního díla

1. Předmětem této smlouvy je vysokoškolská kvalifikační práce (VŠKP):

- disertační práce
- diplomová práce
- bakalářská práce

jiná práce, jejíž druh je specifikován jako .....

(dále jen VŠKP nebo dílo)

Název VŠKP: Automatické detekce obličeje a jeho jednotlivých částí

Vedoucí/školicitel VŠKP: Ing. Jiří Přinosil

Ústav: Ústav telekomunikací

Datum obhajoby VŠKP: .....

VŠKP odevzdal autor nabyvateli v:

- tištěné formě - počet exemplářů 1
- elektronické formě - počet exemplářů 1

2. Autor prohlašuje, že vytvořil samostatnou vlastní tvůrčí činností dílo shora popsané a specifikované. Autor dále prohlašuje, že při zpracovávání díla se sám nedostal do rozporu s autorským zákonem a předpisy souvisejícími a že je dílo dílem původním.
3. Dílo je chráněno jako dílo dle autorského zákona v platném znění.
4. Autor potvrzuje, že listinná a elektronická verze díla je identická.

**Článek 2**  
**Udělení licenčního oprávnění**

1. Autor touto smlouvou poskytuje nabyvateli oprávnění (licenci) k výkonu práva uvedené dílo nevýdělečně užít, archivovat a zpřístupnit ke studijním, výukovým a výzkumným účelům včetně pořizování výpisů, opisů a rozmnoženin.
2. Licence je poskytována celosvětově, pro celou dobu trvání autorských a majetkových práv k dílu.
3. Autor souhlasí se zveřejněním díla v databázi přístupné v mezinárodní síti
  - ihned po uzavření této smlouvy
  - 1 rok po uzavření této smlouvy
  - 3 roky po uzavření této smlouvy
  - 5 let po uzavření této smlouvy
  - 10 let po uzavření této smlouvy(z důvodu utajení v něm obsažených informací)
4. Nevýdělečné zveřejňování díla nabyvatelem v souladu s ustanovením § 47b zákona č. 111/1998 Sb., v platném znění, nevyžaduje licenci a nabyvatel je k němu povinen a oprávněn ze zákona.

**Článek 3**  
**Závěrečná ustanovení**

1. Smlouva je sepsána ve třech vyhotoveních s platností originálu, přičemž po jednom vyhotovení obdrží autor a nabyvatel, další vyhotovení je vloženo do VŠKP.
2. Vztahy mezi smluvními stranami vzniklé a neupravené touto smlouvou se řídí autorským zákonem, občanským zákoníkem, vysokoškolským zákonem, zákonem o archivnictví, v platném znění a popř. dalšími právními předpisy.
3. Licenční smlouva byla uzavřena na základě svobodné a pravé vůle smluvních stran, s plným porozuměním jejímu textu i důsledkům, nikoliv v tísní a za nápadně nevýhodných podmínek.
4. Licenční smlouva nabývá platnosti a účinnosti dnem jejího podpisu oběma smluvními stranami.

V Brně dne: .....

.....

Nabyvatel

.....

Autor

## Anotace

Diplomová práce se zabývá problémem detekce obličejů v barevných statických obrazech. V práci jsou nastíněny základní pojmy, se kterými se lze při detekci obličeje setkat, a jejich vzájemné souvislosti. Jednotlivé přístupy k řešení problému detekce obličeje jsou rozděleny do skupin a blíže popsány. Z těchto přístupů se práce detailně zabývá algoritmem AdaBoost, jenž byl vybrán pro jeho pozitivní vlastnosti, kterými jsou zejména rychlost a dobré dosažené výsledky. V rámci práce byl implementován Viola-Jones detektor. Tento detektor byl natrénován na veřejně přístupné databázi obličejových obrazů a byla zkoumána možnost jeho kombinace s jednoduchým detektorem barvy kůže. Další oblastí, kterou se práce zabývá, je experimentální detekce určitých rysů obličeje.

## Abstract

The master thesis presents an overview of face detection task in color, static images. Face detection term is posed in the context of various branches. Main concepts of face detection and also their relationships are described. Individual approaches are divided into groups and then define in turn. In the thesis is in detail described algorithm AdaBoost, which is selected on the basis of its properties. Especially speed of computation and good detection results are key features. In the scope of this work Viola-Jones detector was implemented. This detector was trained with face pictures from public accessible database. Combination of Viola-Jones detector with simple color detector is described. In the thesis is also presented experiment approach to facial features detection.

## Prohlášení

Prohlašuji, že svůj semestrální projekt na téma Automatické detekce obličeje a jeho jednotlivých částí jsem vypracoval samostatně pod vedením vedoucího semestrálního projektu 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 uvedeného semestrálního projektu dále prohlašuji, že v souvislosti s vytvořením tohoto projektu 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.

V Brně .....

.....

podpis autora

## Poděkování

Za množství cenných rad, optimismus, trpělivost, vstřícnost a podporu děkuji vedoucímu diplomové práce ing. Jiřímu Přinosilovi.

# Obsah

Úvod .....	3
1 Pojmy a vztahy .....	5
1.1 Umělá inteligence ( <i>Artificial Inteligence, AI</i> ) .....	5
1.2 Počítačové vidění ( <i>Computer Vision</i> ).....	6
1.3 Rozpoznávání vzorů ( <i>Pattern Recognition</i> ) .....	7
1.4 Zpracování obrazu ( <i>Image Processing</i> ).....	8
1.4.1 Morfologické operace .....	9
1.5 Strojové učení ( <i>Machine Learning</i> ).....	11
1.5.1 Inference .....	12
1.5.2 Klasifikátor .....	13
1.5.3 Umělé neuronové sítě, <i>Perceptron</i> .....	13
1.5.4 <i>Boosting</i> .....	16
1.5.5 <i>PCA (Principal Component Analysis)</i> .....	16
1.5.6 <i>SVM (Support Vector Machine)</i> .....	17
1.6 Detekce tváře ( <i>Face Detection</i> ).....	17
1.7 Databáze tváří .....	20
1.8 Hodnocení detekce .....	23
2 Přístupy k detekci tváře .....	26
2.1 Metody založené na znalosti.....	26
2.2 Metody založené na invariantních rysech.....	27
2.2.1 Barevný model kůže.....	28
2.3 Metody porovnávání se šablonou .....	29
2.4 Metody založené na vzhladu .....	30
2.4.1 <i>Neuronová síť</i> .....	30

3	Praktická část.....	32
3.1	Detektor barvy kůže.....	32
3.1.1	<i>Barevné prostory</i> .....	32
3.1.2	<i>Model barvy kůže</i> .....	33
3.2	AdaBoost.....	35
3.3	Učení perceptronu .....	37
3.4	Viola-Jones detektor .....	38
3.5	Realizace detektoru .....	42
3.5.1	<i>Efektivní normalizace rozptylu v obraze</i> .....	44
3.5.2	<i>Adaptační fáze - trénování</i> .....	44
3.5.3	<i>Pracovní fáze</i> .....	46
3.5.4	<i>Fáze detekce očí</i> .....	50
3.6	Experimentální výsledky.....	52
4	Závěr.....	53
	Použitá literatura .....	54
	Seznam příloh.....	58
A	Ukázky správných detekcí .....	59
B	Ukázky chybných detekcí.....	60



# Seznam použitých obrázků

Obr. 1: Výřez obrazu <i>The Forest Has Eyes</i> (Bev Doolittle – 1984) .....	4
Obr. 1.1: Obecný systém pro rozpoznávání vzorů .....	8
Obr. 1.2: Příklady strukturních elementů .....	9
Obr. 1.3: Ukázka dilatace .....	10
Obr. 1.4: Schéma klasifikátoru .....	13
Obr. 1.5: Vrstvy neuronové sítě .....	14
Obr. 1.6: Schéma neuronu .....	14
Obr. 1.7: Příklady aktivačních funkcí .....	15
Obr. 1.8: Příklad lineárně separovatelných tříd pomocí perceptronu .....	15
Obr. 1.9: Ukázka oddělení dvou tříd pomocí převodu do vyšší dimenze .....	17
Obr. 1.10: Přehled oblastí při dělení detekce obličeje .....	18
Obr. 1.11: Blokové schéma systému pro detekci a analýzu lidských tváří .....	20
Obr. 1.12: Ukázka prvních 100 snímků z trénovací množiny .....	22
Obr. 1.13: ROC prostor .....	25
Obr. 3.1: Úspěšná detekce barvy kůže .....	34
Obr. 3.2: Ukázka selhání detektoru kůže .....	34
Obr. 3.3: Příznaky podobné Haarově vlnce (haar-like feature) .....	38
Obr. 3.4: Umístění pod-okna v původním obrázku .....	38
Obr. 3.5: Suma pixelů počítaná pomocí integrálního obrazu .....	39
Obr. 3.6: Obrázek kaskády .....	40
Obr. 3.7: Jednotlivé kroky programu .....	43
Obr. 3.8: Popis prvních tří řádků souboru <i>Classifier.txt</i> .....	46
Obr. 3.9: Znázornění struktury proměnné Cluster .....	48
Obr. 3.10: Ukázka odstranění mnohonásobné detekce .....	48
Obr. 3.11: Ukázka správně a chybně zvýrazněných očí .....	51

# Úvod

Obličej a jeho části (oči, nos, rty, atd.) jsou velice rozmanité a právě tato rozmanitost je důležitá ve vzájemném rozpoznání lidí, často na první pohled. Obličej vyjadřuje pocity a nálady a odráží se v něm věk člověka. Schopnost rozpoznávání tváře má člověk vrozenou a vyvíjí se s ním od narození. Je ukončena asi ve dvou měsících věku, kdy určitá oblast mozku již je schopna tváře rozpoznat [20]. Jedna z nejvíce podporovaných teorií vnímání tváře tvrdí, že se tento proces děje v několika krocích [1]; od základní manipulace s vnímanou informací až po získání detailů o osobě (jako je právě její věk nebo pohlaví), aby bylo možno vyvolat důležité informace vážící se k dané osobě (jméno či vzpomínky).

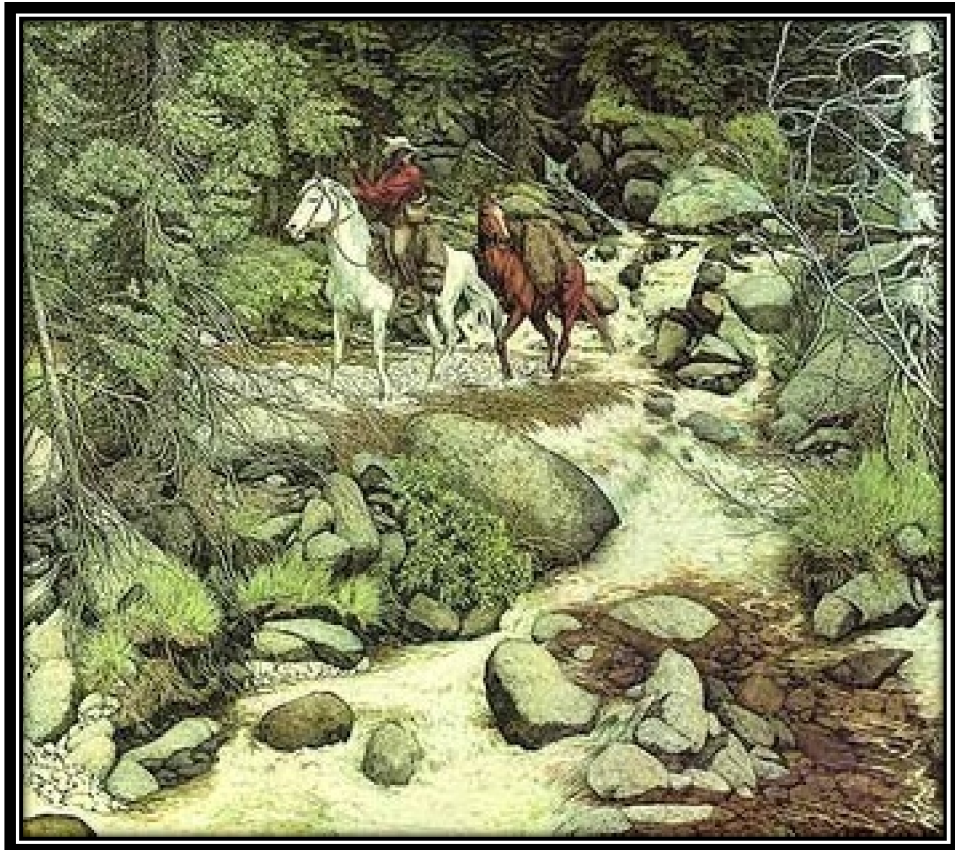
Schopnost rychlé lokalizace lidského obličeje se zdá být samozřejmá a neomezená. Přesto existují obrazy, kde to tak jednoduché být nemusí a i člověk je musí hledat (viz obr. 1). Využívá přitom tzv. hrubou sílu (*brute-force*) – prohlíží obrázek postupně a soustředí se na větší či menší detaily, což je velice často použito i u detektorů tváře.

Cílem této práce je seznámit se problematikou a popsat z komplexního náhledu možnosti detekce obličeje, což je technologie schopná ve vstupním obraze lokalizovat jednotlivé obličeje bez ohledu na jejich orientaci a výraz, velikost nebo světelné podmínky, za kterých je obraz pořízen. Detekuje charakteristické znaky tváře a ignoruje cokoli jiného (stromy, budovy, automobily). Detekce obličeje může být chápána jako zvláštní případ detekce objektů, která lokalizuje jakýkoliv objekt, jenž spadá do dané skupiny.

Protože se jedná o velmi rozsáhlou problematiku, je první kapitola jen stručným přehledem souvislostí a popisem oblastí, se kterými se lze ve vztahu s detekcí obličeje setkat. Seznamuje s pojmy, jako je umělá inteligence, počítačové vidění, rozpoznávání vzorů, zpracování obrazu. V souvislosti s oborem strojového učení jsou popsány

některé ze základních metod, které se při detekci tváře s úspěchem využívají. Druhá kapitola popisuje možné metody přístupu, které dělí do čtyř skupin. Vzhledem k množství různých realizací, vybírá vždy jen jednu, kterou blíže popisuje. Větší pozornost je věnována detektoru barvy kůže a metodě založené na neuronové síti.

Třetí kapitola se věnuje řešení vlastního problému detekce tváře a obličejových částí. Je popsán detektor barvy kůže a Viola-Jones detektor, který klasifikuje obraz na základě Haarových příznaků. Jsou popsány základy těchto metod. Současně je popsán samotný program, který je implementován v prostředí MATLAB a C/C++, a uvedeny výstupy z jednotlivých jeho částí.



Obr. 1: Výřez obrazu *The Forest Has Eyes* (Bev Doolittle – 1984)

# 1 Pojmy a vztahy

## 1.1 Umělá inteligence (*Artificial Intelligence, AI*)

Moderní definice umělé inteligence je „studium a návrh inteligentních agentů“, kde spojení inteligentní agent představuje systém, který chápe a vnímá své okolí a podniká kroky, které maximalizují jeho šance na úspěch. Pojem umělé inteligence slouží také k popisu vlastnosti strojů nebo programů – projevují se inteligencí, např. dokáží reagovat na různé změny a přizpůsobit se jim.

Mezi vlastnosti, jakými se tato inteligence může projevovat, patří úsudek, znalost, plánování, učení, komunikace, vnímání a schopnost pohybu a manipulace s předměty. V případě, že se mluví o všeobecné inteligenci (označována jako „*strong AI*“), pak je to taková, která odpovídá lidské. Zatím jí ovšem nebylo dosaženo, přesto je ale dlouhodobým cílem AI výzkumu [28].

Podstatná část umělé inteligence se zabývá autonomním plánováním či uvažováním systémů, které konají mechanické akce jako například pohyb robota určitým prostředím. Takový typ zpracování obvykle vyžaduje vstupní data od systému počítačového vidění, jenž zde představuje zrakový senzor a poskytuje tak velmi podrobná data nejen o okolním prostředí, ale i o robotu vzhledem k němu. Je tedy zřejmé, že velmi úzce souvisí s oborem počítačového vidění, které je tak někdy chápáno jako součást umělé inteligence.

## 1.2 Počítačové vidění (*Computer Vision*)

Internet je místem, které se z velké části využívá pro přenos grafických informací. Příkladem mohou být lidé, kteří na něm prezentují své fotografie. Grafický formát má ovšem oproti textovému jednu důležitou nevýhodu. Zatímco v textu lidé mohou libovolně vyhledávat slova, která je zajímají. V grafické informaci to tak jednoduché není. Pro vyhledávání takového typu musí být obraz indexován, případně popsán, díky čemuž víme, že například daná fotografie je rodinnou fotografií nebo fotografií krajiny, že obsahuje takové a takové předměty, atp. Pro velký objem obrazových dat je ovšem tento popis pracovně náročný, nákladný a pomalý, protože jej musí provádět člověk. Snaha nahrazení člověka v této situaci je jedním z mnoha důvodů vzniku vědního oboru nazvaného počítačové vidění.

Tato věda a technologie s ní související umožňuje počítačům „vidět“. Jde o stále se vyvíjející obor, přičemž vznikají nástroje, které umožňují počítačům „pochopení“ vizuální informace v obrazech. Tento vývoj je dán jednak rozvojem poznatků a jednak vývojem elektroniky a počítačů, které umožňují tyto poznatky realizovat. Jedná se spíše o spojení více různých metod, které jsou zaměřeny na specifický cíl, či zastřešení podobných problémů a jejich řešení; a jen zřídka je lze zobecnit na nějakou širší oblast aplikací. Úloha počítačového vidění tedy není nijak striktně definována. Různorodost aplikací dokládají následující příklady:

- Řídicí procesy (průmysloví roboti)
- Organizace informací (databáze obrazů)
- Detekce událostí (čítač lidí – jejich průchod určitou oblastí)
- Modelování objektů nebo prostředí (analýza lékařských snímků)
- Inspekční úkoly (kontrola povrchů)
- Rozpoznávání vzorů (geografické informační systémy)
- 3D rekonstrukce, interpretace scény

Setkat se lze rovněž s pojmem **strojové vidění** (*machine vision*), což je aplikace počítačového vidění v průmyslové praxi a výrobě. Na rozdíl od počítačového vidění

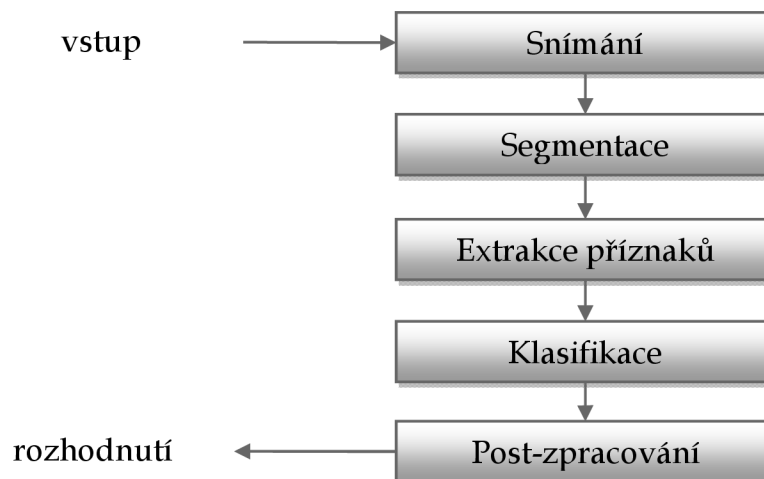
je v něm zapotřebí digitálních vstup/výstupních zařízení a počítačové sítě, aby bylo možné ovládat jiné výrobní stroje.

Klasickým problémem v počítačovém vidění je rozhodování, zda obraz obsahuje nějaký specifický objekt, charakteristickou vlastnost nebo aktivitu (příkladem je **rozpoznávání vzorů** – *pattern recognition*, blíže další kapitola). Přestože zde neexistuje žádný obecný přístup, existují různé metody, které dokáží řešit tento problém pro specifické objekty, např. pro jednoduché geometrické objekty, lidské tváře, tištěný či psaný text, dopravní prostředky; a specifické podmínky, které mohou být dány dostatečným osvětlením, strukturou pozadí a relativní pozicí objektu vzhledem ke kameře.

Cílem výzkumu v oblasti počítačového vidění je poskytnout počítačům vnímání podobné člověku tak, že budou schopny snímat své okolí, rozumět snímaným datům, podnikat odpovídající kroky a učit se ze zkušenosti za účelem zlepšení svého výkonu. Toto zdokonalování na základě vstupních dat spadá do rozsáhlého oboru nazvaného **strojové učení**. Protože snímaná data vždy nemusí být ideálně formována, je zapotřebí je často ještě upravit. V případě snímků takové procesy zahrnuje obor zpracování obrazu.

### 1.3 Rozpoznávání vzorů (*Pattern Recognition*)

Rozpoznávání vzorů je odvětví strojového učení, jehož cílem je třídění (klasifikace) objektů (vzorů) do tříd na základě předchozí znalosti nebo statistických vlastností. Toto třídění je založeno na metodách učení s učitelem i bez učitele. Má v dnešní době široký záběr využití, neboť objekty mohou představovat písmena strojového textu, ručně psaného textu nebo např. jednotlivá slova lidské řeči, čemuž odpovídají aplikace rozpoznávání znaků OCR, rozpoznávání ručně psaných textů nebo rozpoznávání řeči. Systém na rozpoznávání vzorů se skládá z několika částí, které je možné vidět na obr. 1.1. První krok představuje snímání obrazu (signálu) kamerou (mikrofonem).



Obr. 1.1: Obecný systém pro rozpoznávání vzorů

Dalším krokem je segmentace, kdy dochází k vydělení a izolaci objektů, které jsou předmětem rozpoznávání, od ostatních objektů. Jedná se o jeden z nejtěžších kroků, výrazně ovlivňuje výkon celého systému a usnadňuje provádění dalších kroků. V následujícím kroku probíhá extrakce příznaků, což představuje měření vlastností objektů (např. tvar, barva, textura), jež jsou významné pro klasifikaci. Tyto příznaky by ideálně měly být invariantní vůči transformacím a škálování. Typ příznaků je speciálně závislý na typu a předmětu rozpoznávání. Na základě vyextrahovaných příznaků se provádí klasifikace, kdy se příznaky vyhodnocují a podle výsledků tohoto vyhodnocení se objekty zařazují do určité kategorie. Problémem je, že ne vždy lze nalézt příznaky, které by zcela popisovali danou kategorii.

## 1.4 Zpracování obrazu (*Image Processing*)

Zpracování obrazu je jakékoliv zpracování informace, jehož vstupní data tvoří obraz a to nejen statický, ale i v čase se měnící, příkladem pak může být fotografie nebo snímky videa; výstup nemusí být nutně obraz, ale může být například soubor křivek (detekce hran). Většina technik zahrnuje zacházení s obrazem jako dvou-dimenzionálním signálem a používá na něm standardní techniky zpracování signálů

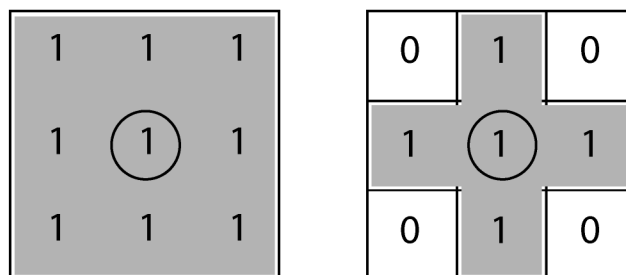
- příkladem může být rozostření nebo doostření, které je na obraz aplikováno pomocí konvoluce se speciálním filtrem. Mezi typické operace zpracování obrazu patří:

- Geometrické transformace (zvětšení/zvětšení, rotace)
- Korekce barev (nastavení jasu a kontrastu, převod do jiného barevného prostoru)
- Kombinace dvou nebo více obrazů (průměr, míchání, rozdíl)
- Segmentace do regionů
- Editace a digitální retušování
- Zvětšení kvality (dekonvoluce k omezení rozostření, odstranění trhlin)

Některé techniky, jako morfologické zpracování obrazu (morfologie je nauka o tvarech, jedná se o extrakci požadovaných součástí obrazu - dilatace, eroze), jsou specifické pro binární nebo šedotónové obrazy.

### 1.4.1 Morfologické operace

Morfologie je nauka o tvarech. Matematická morfologie poskytuje nástroje pro extrakci požadovaných částí obrazu a je založena na nelineárních operacích v obrazu. Každá operace je vnímána jako transformace mezi obrazem a tzv. **strukturním elementem**, viz obr. 1.2. Strukturní element představuje, stejně jako obraz, množinu bodů. Od vstupního obrazu se však liší svou velikostí - je totiž mnohem menší. Typické aplikace těchto metod jsou předzpracování obrazu (odstranění šumu), finální úpravy obrazu, detekce hran [12] .



Obr. 1.2: Příklady strukturních elementů

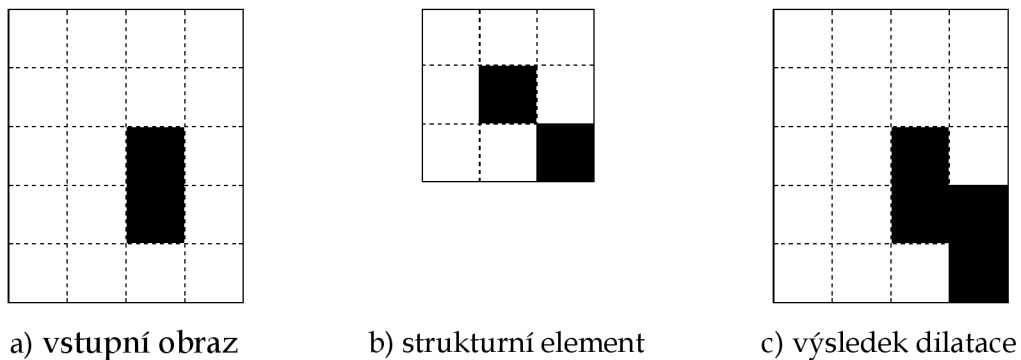


## Dilatace

Dilatace je jednou ze dvou základních operací používaných v matematické morfologii, sčítá dvě bodové množiny – strukturní element  $SE$  a vstupní obraz  $A$ , viz rovnice (1.1). Typicky je používána spolu s binárními obrazy, existuje ovšem i její šedotónová varianta. Efektem binární dilatace je zvětšení objektu, zatímco díry v objektu se zaplňují. Vstup této operace tvoří obraz, který má být transformován, a strukturní element.

$$D(A, SE) = A \oplus SE = \bigcup_{se \in SE} A_{se} \quad (1.1)$$

Zjednodušeně lze dilataci chápat jako porovnávání vstupního obrazu se strukturním elementem, který je v obraze posouván. Pokud je hodnota středu strukturního elementu rovna odpovídajícímu bodu obrazu, obraz je vyplněn (sečten) se strukturním elementem, viz obr. 1.3.



Obr. 1.3: Ukázka dilatace

## Eroze

Eroze je definována jako průnik všech posunů v obraze  $A$ , viz rovnice (1.2). Její základní funkcí je odstranění/ořezání hranic objektu. Objekt se touto operací zmenšuje, díry v objektech se naopak zvětšují a objekty menší než strukturní element vymizí. Eroze se často využívá k rozložení objektu na jednodušší části a tím k zjednodušení struktury.

$$E(A, SE) = A \ominus SE = \bigcap_{se \in SE} A_{-se} \quad (1.2)$$

## 1.5 Strojové učení (*Machine Learning*)

Obor strojového učení je založen na myšlence, že algoritmy a systémy mohou zdokonalovat svůj výkon v závislosti na čase. Mohou se „učit“. Toto učení je velice často založeno na různých pravděpodobnostech, proto se v dalším textu pojmy z oboru pravděpodobnosti často vyskytnou.

Obrazy obsahující obličej jsou podstatným zdrojem pro inteligentní systémy využívající strojové učení. Výzkumné cíle se soustředí na zpracování tváří jako je rozpoznávání tváří (*face recognition*), sledování tváří (*face tracking*), odhad pozice (*pose estimation*) a rozpoznání výrazu tváře (*expression recognition*). Nicméně všechny z těchto úkonů předpokládají už práci se samotnou tváří. Je tedy nutné tvář v obraze nebo sekvenci obrazů identifikovat a lokalizovat. Pro vytvoření plně automatického systému, který analyzuje informace o tvářích v obrazech, je zajisté nutný robustní a efektivní algoritmus detekce tváře. Robustní z hlediska přesnosti (měl by dokázat detekovat všechny tváře s co nejmenší chybou) a efektivní z hlediska výpočetní rychlosti (hledání tváře by mělo být v reálném čase).

Na základě přístupu k učení lze algoritmy strojového učení rozdělit do dvou základních skupin [7], přičemž při detekci obličejů bývá nejvyužívanější právě první zmíněný.

- **Učení s učitelem** (*Supervised learning*). Na základě **trénovacích dat**, což je množina vstupních hodnot a jim odpovídajících výstupních hodnot, vytváří klasifikátor (viz kap. 1.5.2) funkci, která mapuje vstupní hodnoty na výstupní. Příkladem trénovacích dat mohou být obrazy a jim přiřazena hodnota výstupu -1 nebo 1, pokud se jedná či nejedná o obličej. Pokud je po natrénování na vstup přiveden obraz, výstupem funkce je -1 nebo 1. Protože toto natrénování není schopno obsáhnout celou množinu tváří, bude klasifikace vždy více či méně chybová (viz kap. 1.8). Snahou je tuto chybu co nejvíce snížit.

- **Učení bez učitele** (*Unsupervised learning*). V tomto případě chybí trénovací data a algoritmus tak nezná požadované výstupní hodnoty. Vstupní data jsou shromažďována a jsou považována za množinu náhodných proměnných. K této množině pak existují dva přístupy. Jednak je možné vytvořit statistický model (např. Bayesova síť) pomocí odhadu hustoty pravděpodobnosti nebo pomocí techniky extrakce charakteristických vlastností získat ze vstupu statistické zákonitosti [6] .

### 1.5.1 Inference

Inference je postup k dosažení výsledku, odvození. Existují tři typy, které mohou nastat v závislosti na rovnici (1.3), kde  $F$  značí vstupní proměnnou,  $C$  výstupní proměnnou a  $T$  zobrazení, které přiřazuje vstupní proměnné výstupní proměnnou.

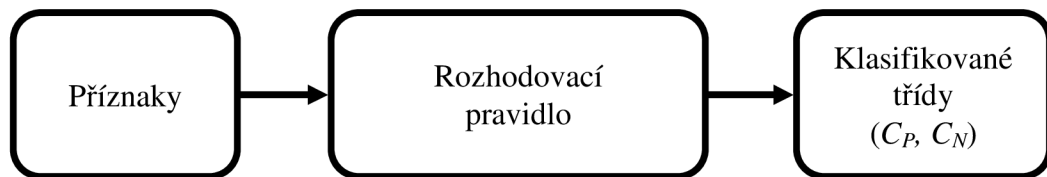
$$F \xrightarrow{T} C \quad (1.3)$$

- **Dedukce** spočívá ve znalosti proměnné  $F$ , zobrazení  $T$  a snahou je určit  $C$ . Pokud bude správné  $T$ , pak bude vždy správný výsledek. Lehce si to lze ukázat na příkladu  $F: x = 2$ ;  $T: y = x + 2$ ; pak  $C: y = ?$ )
- **Abdukce** je proces, kdy jsou známé výstupní proměnná  $C$ , zobrazení  $T$  a hledaná je vstupní proměnná  $F$ . Správný výsledek v tomto případě není zaručen, protože např.  $T: y = x^2$ ;  $C: y = 9$ ; pak  $F: x = -3$  nebo  $3$  ?
- **Indukce** hledá zobrazení  $T$  ze známých hodnot  $F$  a  $C$ . Není zaručen správný výsledek (př.  $F: x = 3$ ;  $C: y = 9$ ;  $T: y = 3x$  nebo  $y = x^2$  ?).

Detekci obličejů v kontextu strojového učení pak lze ve většině případů chápat jako proces indukce, kdy jsou známa vstupní data (charakteristické vlastnosti obrazů) a jim odpovídající výstupní data (označení, zda se v daném obraze tvář nachází či nikoli); a snahou je z nich extrahovat funkci, která bude co možná nejpřesněji mapovat tyto vstupy na výstup.

## 1.5.2 Klasifikátor

Klasifikátor je algoritmus, který při vhodné množině znalostí je schopen úspěšně rozdělovat vstupní data s hodnotami atributů (**příznaků**) do výstupních předem zvolených skupin (**tříd**), jak je vidět na obr. 1.4. Vhodná volba klasifikačního algoritmu představuje nutnou podmínku k úspěšné klasifikaci. V případě, že třídy jsou pouze dvě, bývá označována jako tzv. **dichotomická klasifikace**. Pokud tyto třídy spolu souvisí ve smyslu, že v první třídě je množina, která má určitou vlastnost (např. obsahuje obličej) a druhá třída pokrývá množinu, která tuto vlastnost nemá (např. obsahuje cokoli jiného než obličej) používá se označení **binární klasifikátor**. Detekci obličeje lze tedy chápat jako binární klasifikaci. Pro přehlednost dalšího textu je vhodné tyto třídy označit. Obrázky tváře tedy budou označovány jako třída  $C_P$  (znamená pozitivní případy) a obrázky pozadí budou označovány jako třída  $C_N$  (negativní, nežádoucí případy).

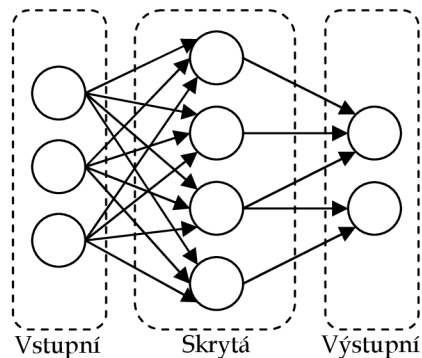


Obr. 1.4: Schéma klasifikátoru

## 1.5.3 Umělé neuronové sítě, Perceptron

Jedná se o matematický nebo výpočetní model založený na pozorování biologické neuronové sítě. Ta se skládá z neuronů, což jsou elektricky excitované buňky nervového systému, které dokáží zpracovávat a přenášet informaci. Stejně tak umělá neuronová síť se skládá z mezi sebou vzájemně spojených skupin (vytváří síť) umělých neuronů a zpracovává informace pomocí takového spojení. Neuronovou síť lze rozdělit na vrstvy, jak je zobrazeno na obr. 1.5. Každá vrstva obsahuje určitý

počet neuronů. Vstupní data přicházejí na vstupní vrstvu, jejíž výstup pokračuje do skryté vrstvy. Výstup sítě je tvořen výstupní vrstvou.

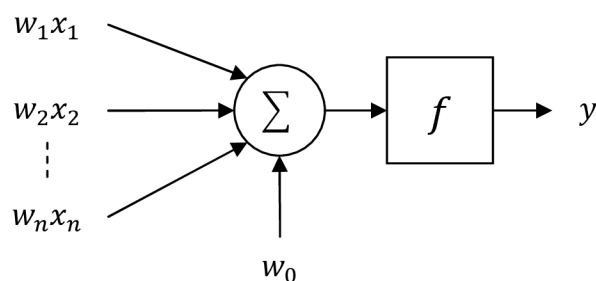


Obr. 1.5: Vrstvy neuronové sítě

**Neuron**, jehož schéma je na obr. 1.6, lze popsat pomocí rovnice (1.4), kde  $x_i \in \mathbf{X}, w_i \in \mathbf{W}, pro i = 0, \dots, n$ .

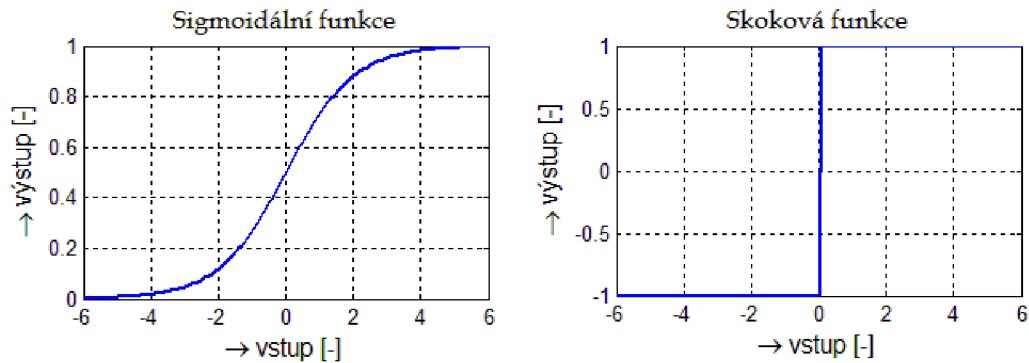
$$y = f\left(\sum_{i=0}^n x_i \cdot w_i\right) \quad (1.4)$$

Vstupy neuronu tvoří vektor  $\mathbf{x}$ . Každý z  $n$  vstupů má přiřazenou určitou váhu  $w$ . Speciální váhu zde představuje  $w_0$ . Jedná se o tzv. **práh**, který tvoří bariéru na vstupu neuronu a tím ovlivňuje práh rozhodování. Funkce  $f$  se označuje jako **aktivační (přenosová) funkce** a může mít různé průběhy (obr. 1.7) v závislosti na konkrétním typu řešené úlohy. Ze sumy vážených vstupních hodnot je počítána hodnota aktivační funkce, která odpovídá výstupní hodnotě neuronu.



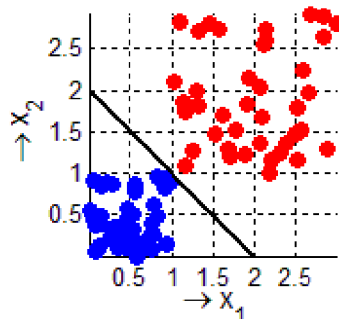
Obr. 1.6: Schéma neuronu

Práci neuronové sítě lze rozdělit na dvě fáze: adaptační a pracovní. Při adaptační fázi dochází k tzv. **trénování/učení** - nastavování vah jednotlivých vstupů na základě trénovací množiny  $\{x^j, y^j\}, j = 1, \dots, K$ , kde  $x^j$  označuje j-tý vektor příznaků,  $y^j = \{0; 1\}$  jemu odpovídající třídu a  $K$  je počet vzorků trénovací množiny. V pracovní fázi jsou již váhy nastaveny a neuronová síť zpracovává vstupní vektory a generuje příslušný výstup.



Obr. 1.7: Příklady aktivačních funkcí

Modelů neuronové sítě existuje celá řada, např. dopředná neuronová síť, Kohonenova mapa. Nejjednodušším typem neuronové sítě je **perceptron** [23]. Je tvořen pouze jedním neuronem a určen pro dichotomickou klasifikaci, proto odpovídá jeho aktivační funkce skokové funkci. Perceptron je schopen pouze lineární separace. Tato lineární separace probíhá výběrem nadroviny, která odděluje obě třídy. Tuto nadrovinu si lze představit jako přímku v dvojrozměrném prostoru (obr. 1.8).



Obr. 1.8: Příklad lineárně separovatelných tříd pomocí perceptronu

## 1.5.4 Boosting

Cílem metody boosting je zlepšení klasifikační přesnosti libovolného algoritmu strojového učení. Základem je vytvoření více klasifikátorů, označovaných jako slabí žáci, **slabé klasifikátory** (*weak learners*). Tyto klasifikátory vznikají pomocí výběru vzorků ze základní trénovací množiny (bez vracení). První klasifikátor má přesnost jen o něco málo lepší než 50%. Dále jsou přidávány další klasifikátory mající stejnou přesnost a jsou trénovány na předchozích chybně klasifikovaných vzorcích. Tím je vygenerován soubor klasifikátorů označovaných jako silný žák, **silný klasifikátor** (*strong learner*), jehož celková klasifikační přesnost je libovolně vysoká vzhledem ke vzorkům v trénovací množině – klasifikace byla zesílena (*boosted*) [30]. Nevýhodou této metody je požadavek velmi rozsáhlé trénovací množiny. Nejpopulárnějším příkladem této metody je AdaBoost (viz kap. 3.2).

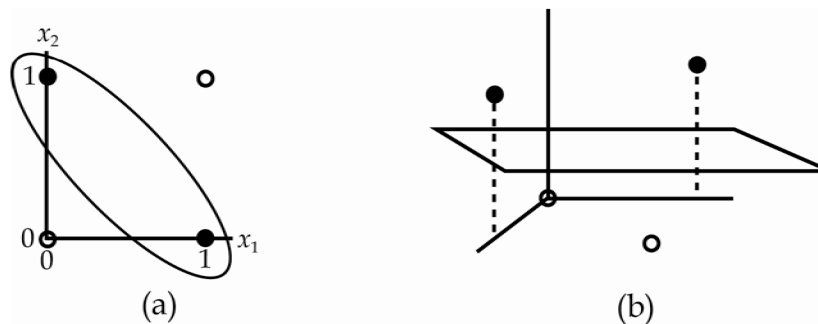
## 1.5.5 PCA (Principal Component Analysis)

Jedná se o techniku používanou ke snížení dimenze dat s co nejmenší ztrátou informace. PCA bývá označována také jako Karhunen-Loèveho transformace nebo Hotellingova transformace. Jejím cílem je redukce původního počtu popisovaných proměnných novými veličinami (umělými), označenými jako **komponenty**, které shrnují informaci o původních proměnných za cenu minimální ztráty informace. Tyto komponenty jsou vzájemně nezávislé a jsou seřazeny podle svého příspěvku k vysvětlení celkového rozptylu pozorovaných proměnných.

Princip spočívá v určení kovarianční matice a jejích vlastních čísel a vektorů – komponenty. Vlastní číslo s největší hodnotou je pak nazýváno hlavní komponenta (*principal component*) souboru dat. Velikost vlastních čísel odráží jejich důležitost. Vlastní čísla s nejnižší hodnotou jsou nejméně důležitá, mají nejmenší vliv na soubor dat a jejich zanedbáním se dosahuje omezení dimenze. Seřazením vlastních vektorů podle velikosti vlastních čísel do matice je získán charakteristický vektor (*feature vector*).

## 1.5.6 SVM (Support Vector Machine)

Support vector machine (SVM) je metoda klasifikace lineárních dat [32]. Její geometrickou interpretaci lze chápat jako hledání optimální **nadroviny** (*hyperplane*), která odděluje hodnoty různých tříd tak, že je stejně vzdálena od souboru hodnot obou tříd. Důležitou součástí této metody je **jádrová transformace** (*kernel method*), která - v případě, že třídy nejde lineárně oddělit nadplochou - převádí hodnoty do prostoru vyšší dimenze, kde toto oddělení již je možné (viz obr. 1.9).



Obr. 1.9: Ukázka oddělení dvou tříd pomocí převodu do vyšší dimenze

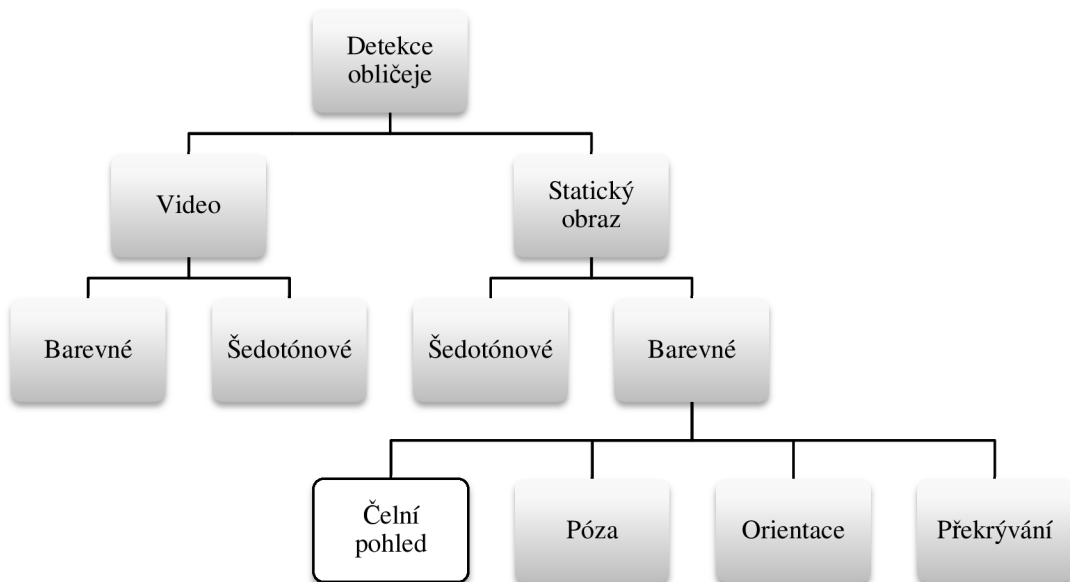
## 1.6 Detekce tváře (*Face Detection*)

V předchozích kapitolách byly uvedeny vybrané základní poznatky používané k rozpoznávání jednotlivých objektů. Jejich vzájemnou kombinaci jsme schopni využít při řešení problematiky detekce obličeje. Na detekci obličeje lze nahlížet jako na úlohu rozpoznávání vzorů, kdy dochází k rozpoznávání dvou tříd - obličeje a pozadí (cokoli jiného než tvář). Naším úkolem bude identifikovat a lokalizovat (popsat souřadnicemi nebo označit) lidské tváře bez ohledu na:

- pozici obličeje v obraze
- velikost obličeje
- orientaci (natočení v ose optického snímače)
- pózu - natočení obličeje (v ose páteře - profil, 45°)
- osvětlení



Na detektor tak lze nahlížet jako na binární klasifikátor, který rozhoduje, zda daná část obrazu obsahuje obličej nebo ne. Detekování obličeje je obtížné, neboť obličej každého člověka je jiný. A samozřejmě také proto, že závisí na výše uvedených podmínkách, které se mohou různě měnit. Obraz tváře člověka se mění v závislosti na natočení jeho hlavy, je jiná při čelním pohledu a jinak vypadá při pohledu z profilu. Pokud je tvář natočena pouze v rovině snímku, bývá označován jako **frontální obraz**. Obličej se na snímcích jeví různě v závislosti na osvětlení – jinak vypadá při osvětlení zepředu a jinak, když polovinu obličeje halí stín, atd. Velikost tváře se také jeví jinak v závislosti na vzdálenosti člověka od objektivu. Úlohu detekce dále znesnadňují např. vousy, brýle či jiné předměty, které tvář překrývají. Na vzhled tváře mají i vliv emoce člověka (výraz tváře). Je jasné, že tvář člověka, který se mračí, vypadá jinak, než když se směje. S tímto vším se musí detektor vyrovnat. V průběhu let se ukázalo, že nejefektivnějším přístupem k řešení takovéto detekce je právě strojové učení, proto je mu v poslední době věnována zvýšená pozornost. Příklad dělení detekce obličeje podle typu vstupních dat je na obr. 1.10. Tato práce se zabývá detekcí tváří orientovaných čelně v barevných, statických snímcích.



Obr. 1.10: Přehled oblastí při dělení detekce obličeje

Existuje mnoho systémů, které souvisejí se zpracováním tváří:

- **Lokalizace tváře** (*face localization*) slouží k určení pozice jediné tváře.
- **Detekce obličejových rysů** (*facial features detection*) hledá a lokalizuje např. oči, nos, ústa, apod.
- **Rozpoznávání obličeje** (*face recognition*) a **identifikace obličeje** (*face identification*) porovnává vstupní obraz s databází a určí, zda se tvář nachází v databázi.
- **Autentizace obličeje** (*face authentication*) slouží k ověření, že obličej v obraze je opravdu ten, za který se vydává.
- **Sledování obličeje** (*face tracking*) nepřetržitě určuje polohu i orientaci obličeje v posloupnosti obrazů v reálném čase.
- **Rozpoznávání výrazu tváře** (*facial expression recognition*) slouží k určení nálady a citového stavu člověka (radost, smutek, atd.).

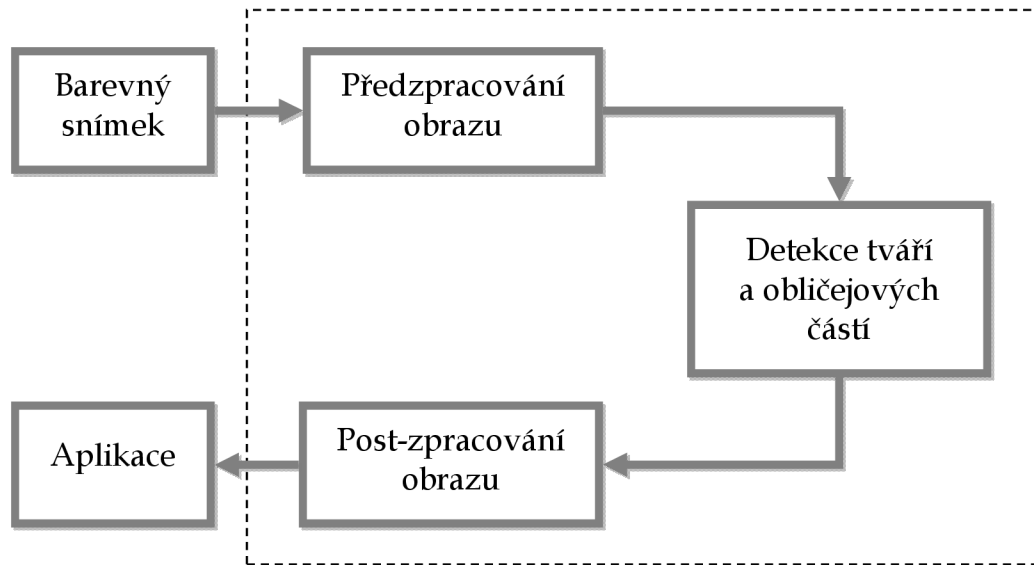
Je tedy zřejmé, že detekce obličeje je prvním krokem k realizaci výše zmíněných systémů. Detekce tváře může být provedena na základě různých úvah:

- barva pleti (pro obličeje na barevných obrazech)
- pohyb (tváře ve video sekvenci)
- tvar hlavy/obličeje
- výskyt obličejových rysů
- kombinace předchozích

Vstup obecného systému pro detekci tváře (viz obr. 1.11) představuje šedotónový, barevný snímek nebo videosekvence (posloupnost snímků). Ten je zpracován v prvním bloku – předzpracování obrazu, kde dochází k zlepšení jeho vlastností (kontrast, jas, ostrost a jiné) a extrakci nízko-úrovňových příznaků (detekce barvy pokožky, detekce hran). Výsledný upravený snímek spolu s příznaky vstupuje do bloku detekce tváří.

Zde se v obraze vyberou oblasti odpovídající lidským tvářím. Dále se zde detekují jednotlivé části obličeje, zejména oči, nos a ústa. Požadavkem na tento blok, aby se uplatnil v reálných aplikacích, je zejména rychlost. Dalšími pak jsou nezávislost na počtu tváří, na jejich velikosti, natočení, atd. Informace získané v tomto bloku jsou

podrobeny dalšímu zpracování, kdy se např. může z několikanásobné detekce vybrat jediná nebo se na základě vzhledu tváře určí emoce, tvář se pak vůči nim může normalizovat. Informace o obličejí a jeho částech pak vystupují ze systému a jsou poskytnuty aplikaci, která na jejich základě dokáže řešit různé problémy, jakými jsou rozpoznávání tváří, rozpoznávání pohlaví, atd. Ve většině případů mezi aplikací a systémem je pevný vztah. Systém je pak navržen pro specifické požadavky aplikace.



Obr. 1.11: Blokové schéma systému pro detekci a analýzu lidských tváří

## 1.7 Databáze tváří

Nedílnou součástí úlohy detekce tváří je bezpochyby sada snímků vhodných pro trénování a testování detekčního algoritmu. Existují různé zdroje, které umožňují stažení takovýchto sad prostřednictvím internetu. Některé bývají placené, s možností získání dat na DVD médiu, jistá část je ovšem zpřístupněna bez poplatků. Některé z nich však vyžadují registraci a některé citaci autorů databáze. Neplacené databáze jsou uvedeny v tab. 1.1, seřazeny podle počtu snímků sestupně. Databáze se liší v počtech jednotlivých snímků, v jejich rozlišeních, v počtech osob, které jsou na

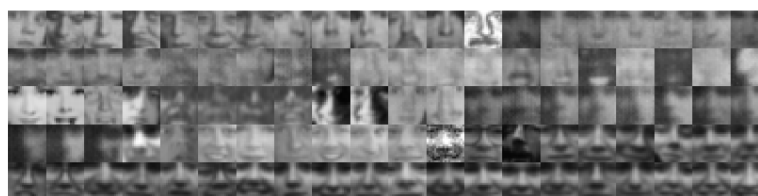
snímcích, světelnými podmínkami, různým natočením tváří vzhledem k snímacímu zařízení atp. Ukázka vzorků z použité databáze je na obr. 1.12.

Tab. 1.1: Přehled vybraných databází tváří

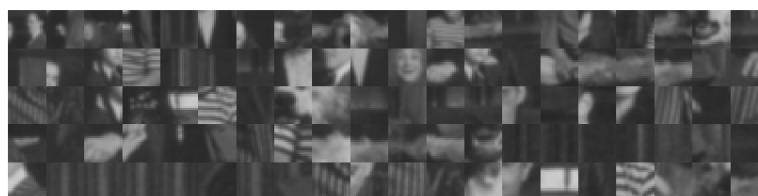
Název databáze	Počet obrázků	Typ obrázků	Rozlišení [pixel]	Popis
<b>CBCL Face Database</b> (Massachusetts Institute of Technology) <a href="http://cbcl.mit.edu/cbcl/software-datasets/FaceData2.html">http://cbcl.mit.edu/cbcl/software-datasets/FaceData2.html</a>	Více než 30000	SŠ	19x19	- Testovací i trénovací množina
<b>Labeled Faces in the Wild</b> (University Of Massachusetts) <a href="http://vis-www.cs.umass.edu/lfw/">http://vis-www.cs.umass.edu/lfw/</a>	Více než 13000	B	150x150	- Shromážděno na internetu a popsáno
<b>Vision Group of Essex University Face Database</b> <a href="http://cswww.essex.ac.uk/mv/allfaces/index.html">http://cswww.essex.ac.uk/mv/allfaces/index.html</a>	7900	B	různá	- Tváře 395 osob - Různé podmínky
<b>The Yale Face Database B</b> <a href="http://cvc.yale.edu/projects/yalefacesB/yalefacesB.html">http://cvc.yale.edu/projects/yalefacesB/yalefacesB.html</a>	5760	SŠ	640x480	- 9 póz - 64 různých světelných podmínek - Popis umístění rysů tváře
<b>The AR Face Database</b> <a href="http://cobweb.ecn.purdue.edu/~aleix/aleix_face_DB.html">http://cobweb.ecn.purdue.edu/~aleix/aleix_face_DB.html</a>	4000	B	768x576	- Tváře 126 lidí - Různé výrazy, světelné podmínky, různé předměty zakrývající tvář - Nutná registrace a citace
<b>FGnet – Head Pose Image Database</b> <a href="http://www-prima.inrialpes.fr/perso/Gourier/Faces/HPDatabase.html">http://www-prima.inrialpes.fr/perso/Gourier/Faces/HPDatabase.html</a>	2790	B	384x288	- Různá natočení hlavy - Tváře 15 osob - Plné pozadí
<b>The HumanScan (BioID) Face Database</b> <a href="http://www.bioid.com/downloads/facedb/ind">http://www.bioid.com/downloads/facedb/ind</a>	1521	SŠ	384x286	- Čelní pohled - Tváře 23 osob - Rozmanitá pozadí,

ex.php				světelné podmínky a velikost - Soubor popisující umístění očí
<b>VALID Database University College Dublin</b> <a href="http://ee.ucd.ie/validdb/datasets.html">http://ee.ucd.ie/validdb/datasets.html</a>	1060	B	576x720	- Tváře 106 lidí - Snímky pořízeny v průběhu měsíce ve 4 kancelářích a ateliéru - Popis umístění očí a úst - Nutná registrace
<b>The Yale Face Database A</b> <a href="http://cvc.yale.edu/projects/yalefaces/yalefaces.html">http://cvc.yale.edu/projects/yalefaces/yalefaces.html</a>	165	SŠ	-	- Tváře 15 lidí - Různé výrazy a světelné podmínky
<b>CMU Frontal Face Images</b> <a href="http://vasc.ri.cmu.edu/idb/html/face/frontal_images/index.html">http://vasc.ri.cmu.edu/idb/html/face/frontal_images/index.html</a>	130	SŠ	různá	- Testovací množina - Popis umístění částí obličeje

**Pozn.: B – barevné snímky, SŠ – snímky ve stupních šedi**



tváře



pozadí

Obr. 1.12: Ukázka prvních 100 snímků z trénovací množiny

## 1.8 Hodnocení detekce

Aby bylo možno ohodnotit kvalitu detektoru, je nutné zavést pár pojmů. Velikost množiny  $C_P$ , tedy celkový počet obrázků tváře ve zkoumané množině, bude označován jako  $P$ . Naproti tomu, velikost množiny  $C_N$  bude označován  $N$ .

Případy, kdy klasifikátoru správně určí snímek z množiny  $C_P$ , se označují  $TP$  (*true positive*), naopak případy, kdy je správně určeno pozadí, jsou označeny jako  $TN$  (*true negative*). Na druhou stranu, aby bylo možno si udělat představu o tom, jak detektor správně klasifikuje, je nutné znát i počet chyb klasifikace. Mohou nastat dvě chyby:

- **chyba I. typu** – označuje se  $FP$  (*false positive*) a nastává v případech, kdy detektor chybně zvolí třídu  $C_P$  (místo tváře vybere pozadí).
- **chyba II. typu** – označuje se  $FN$  (*false negative*) a dochází k ní tehdy, pokud klasifikátor určí třídu  $C_N$  (tvář označí za pozadí).

Tyto základní ukazatele lze zapsat do tzv. čtyřpolní tabulky, viz tab. 1.2. Čísla na hlavní diagonále reprezentují správnou klasifikaci, čísla na vedlejší diagonále vyjadřují chybu mezi dvěma třídami.

Tab. 1.2: Kontingenční tabulka – čtyřpolní tabulka

		Skutečná třída	
		pozitivní	negativní
Klasifikovaná třída	pozitivní	TP	FP
	negativní	FN	TN

Důležitou veličinou je **senzitivita** nebo také **detekční poměr** (*detection rate, true positive rate, TPR*). Senzitivita je procentuálním vyjádřením počtu tváří v obraze, které byly detektorem správně detekovány,  $TPR \in \langle 0; 1 \rangle$ .

$$TPR = \frac{TP}{P} = \frac{TP}{TP + FN} \quad (1.5)$$

**Nespecifita** (*false positive rate, FPR*)  $FPR \in \langle 0; 1 \rangle$  vyjadřuje poměr chybně určených tříd  $C_P$  k celkové velikosti množiny  $C_F$ :

$$FPR = \frac{FP}{N} = \frac{FP}{FP + TN} \quad (1.6)$$

Dalším pojmem je **přesnost** (*accuracy, ACC*)  $ACC \in \langle 0; 1 \rangle$ , která udává poměr správně určených tříd k celkovému počtu tříd:

$$ACC = \frac{TP + TN}{P + N} \quad (1.7)$$

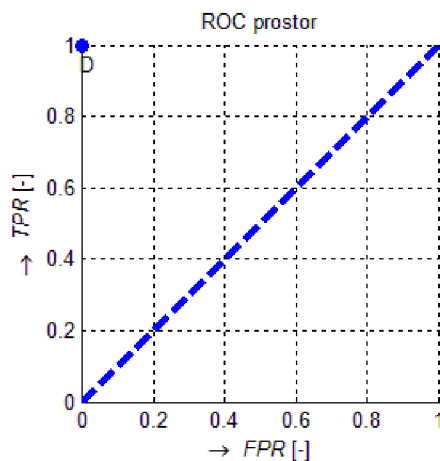
Významným ukazatelem je **pozitivní prediktivní hodnota** (*positive predictive value, PPV*). Určuje jaké procento z pozitivně označených je skutečně pozitivních.

$$PPV = \frac{TP}{TP + FP} \quad (1.8)$$

Nyní je zřejmé, že v ideálním případě je požadována maximální senzitivita a přesnost detektoru. Cílem je tedy dosáhnout maximálního detekčního poměru  $TPR$  při minimálním nespecifitě  $FPR$ . Z reálných aplikací ovšem vyplývá, že není možné dosáhnout obojího, a proto se hledá kompromisní řešení mezi těmito dvěma parametry. Velmi často se pak tyto hodnoty vykreslují do grafu, který bývá označován jako ROC křivka.

Pomocí tzv. **ROC** (*Receiver Operating Characteristic*) křivky lze popsat schopnost predikce klasifikátoru. Nejedná se však jen o křivku, ale o celé zobrazení a prostor. Důležitým ukazatelem je rovněž plocha pod ROC křivkou **AUC** (*Area Under the Curve*), kterou lze interpretovat jako ukazatel kvality klasifikátoru.

TPR jsou vynášeny na svislou osu, FPR na vodorovnou osu grafu a znázorňují různý poměr mezi správně a chybně klasifikovanými hodnotami. Dichotomní klasifikátor je popsán (FPR, TPR) a odpovídá jednomu bodu v **ROC prostoru** (viz obr. 1.13). V tomto prostoru existuje pár důležitých bodů. Spodní levý bod [0,0] představuje případ, kdy klasifikace neobsahuje pozitivní případy. Přesným opakem je pravý horní roh [1,1]. Bod [0,1] představuje dokonalou klasifikaci – nedochází k chybě. V obrázku tomu odpovídá bod D.



Obr. 1.13: ROC prostor

Osa prvního kvadrantu rozděluje ROC prostor na dvě části. Body ležící na ní lze chápat jako klasifikátory, které pracují čistě náhodně (chybovost je 50%). Horní levá část obsahuje klasifikátory, které klasifikují lépe než náhodný klasifikátor. Pravá dolní část pak klasifikátory klasifikující hůře.

Třída regresního klasifikátoru, tedy takového, jehož výstupem je spojitá hodnota, je z této odvozena pomocí prahování. Pak každé nastavení prahu, odpovídá jednomu bodu na křivce. Každá ROC křivka, která je generována na základě konečného množství instancí, je schodovitá funkce [8] .



## 2 Přístupy k detekci tváře

Přístupy lze rozdělovat různě a ne vždy se dá přesně říct, že daná metoda zcela spadá do dané skupiny – někdy může daná metoda patřit do více skupin. Dále je uvedeno rozdělení do čtyř tříd na základě použité metody dle [19].

### 2.1 Metody založené na znalosti

Tyto metody detekce byly vyvinuty na základě vědeckých poznatků o lidské tváři. Popisují základní rysy tváře (oči, nos, ústa) a jejich vzájemné vztahy. Příkladem mohou být oči, které jsou ve tváři umístěny symetricky, ovšem nejen k tváři, ale i nosu a ústům. Vztah mezi nimi může být popsán určitými pravidly, vycházejícími z jejich umístění vzhledem k tváři nebo relativní vzdáleností vůči sobě. Prvním krokem v tomto přístupu je pak extrakce rysů obličeje z obrazu, po které přichází identifikace tváře na základě znalosti vztahů mezi jednotlivými rysy.

Problémem tohoto přístupu je složitost převedení lidských znalostí do správně definovaných pravidel. Pokud totiž budou pravidla příliš přísná, může docházet k chybě v detekci tváří, které jim přesně neodpovídají. Naopak v případě, že pravidla umístění rysů budou příliš obecná, bude docházet k zvyšování chybného pozitivního poměru (tedy chybně označených tváří). Navíc je složité tyto metody rozšířit tak, aby detekovaly různé pózy tváře, neboť by pro každou pózu musely existovat přesně definována pravidla. Na druhou stranu je nutno říct, že tento přístup lze dobře použít pro detekci čelně orientovaných tváří v jednoduchých scénách.

Příkladem takového přístupu může být práce Yanga a Huanga [34], kteří použili více rozlišení jednoho obrázku (získaných pomocí podvzorkování a průměrování) a hierarchickou strukturu pravidel. Ta měla tři úrovně. Na nejnižší úrovni, v obraze

s nejnižším rozlišením, se hledá tvář pomocí rozdílu intenzit jednotlivých pixelů. Postupem k vyšší úrovni se rozlišení zvyšuje. Na vyšší úroveň přechází detekované oblasti z nižší úrovně a jsou podrobeny detekci hran. Na nejvyšší úrovni jsou pak zbylé případy ověřovány na přítomnost očí a úst. Hierarchickou strukturou bylo omezeno množství nutných výpočetních operací, ovšem detekční poměr byl nízký.

## 2.2 Metody založené na invariantních rysech

Tyto metody, na rozdíl od předchozí skupiny, jsou založeny na snaze nalézt invariantní obličejové rysy. Na základě pozorování, že člověk dokáže bez námahy detekovat obličej i objekty v různých pózách a světelných podmínkách, vzniká předpoklad, že musí existovat vlastnosti nebo příznaky, které jsou invariantní vůči těmto podmínkám. Obličejové rysy jako obočí, oči, nos a ústa jsou obvykle extrahovány za použití hranových detektorů. Na základě extrahovaných rysů je vytvořen statistický model, který popisuje jejich vzájemný vztah a ověřuje existenci tváře.

Problémem u těchto algoritmů je možné znehodnocení obrazu šumem nebo špatným osvětlením. Obličejové rysy mohou být náchylné právě na osvětlení, kdy stín může způsobit množství ostrých hran a tím zapříčinit špatnou detekci.

Han a spol. vyvinuli morfologicky založený postup, který nazvali „očím podobné segmenty pro detekci obličej“ (*eye-analogue segments for face detection*). Vychází z tvrzení, že oči a obočí jsou nejvýznačnější rysy lidské tváře. Definují segmenty podobné očím jako hrany v kontuře očí. Nejprve jsou provedeny morfologické operace jako je uzavření nebo prahování pro získání pixelů, jejichž intenzita se významně mění. Tyto pixely tvoří segment podobný očím. Tyto segmenty pak slouží jako vodítko v hledání možných oblastí obsahujících obličej spolu s geometrickou kombinací očí, nosu, obočí a rtů. Výsledné oblasti jsou ještě ověřeny neuronovou sítí [13].

## 2.2.1 Barevný model kůže

Možným způsobem přístupu k detekci obličeje je detekce oblastí obsahujících barvu lidské pokožky, které jsou dále podrobeny jemnějšímu zkoumání. Mezi metodami detekce tváří založených na invariantních rysech (*feature-based*) si tento přístup získal značnou popularitu. Důvodem k tomu je i fakt, že barvy umožňují rychlé zpracování, na rozdíl od jiných obličejových rysů, a jsou vysoce robustní vzhledem ke geometrickým odchylkám lidských tváří. Navíc má lidská kůže charakteristickou barvu, která může být poměrně jednoduše rozpoznána. Tyto vlastnosti tedy vyústily ve snaze vytvořit barevný model kůže.

Při návrhu a realizaci takového řešení se lze obvykle setkat se třemi hlavními problémy:

1. Výběr barevného prostoru
2. Separování barvy kůže v barevném prostoru
3. Způsob zpracování výsledků barevné segmentace

Metoda detekce kůže založená na pixelech obrazu (*pixel-based*) klasifikuje každý pixel obrazu do dvou tříd (kůže/ne-kůže) bez ohledu na okolní pixely a bude jí zde věnována pozornost. Taková klasifikace probíhá na základě barevného modelu kůže. Lze se také setkat s přístupem založeným na barevných oblastech (*region-based*), který bere v úvahu prostorové uspořádání obličejových pixelů.

Detekce barvy kůže může být rozdělena na

- **Neparametrické metody** - odhad rozložení barvy kůže z trénovacích dat bez získání explicitního popisu. Výhody neparametrických je rychlost trénování. Nevýhodou je poměrně velká paměťová náročnost. Jako příklad těchto metod lze uvést LookUpTable a Histogram Model).
- **Parametrické modely** - z trénovací množiny jsou získány parametry, které barvu kůže v barevném prostoru popisují. Většina metod spadajících do této kategorie pracuje pouze s chrominanční složkou (příkladem může být jednoduchá Gaussova funkce, jejímiž parametry jsou vektor středních hodnot a kovarianční matice).

Úspěch detekce barvy kůže významně závisí na výběru barevného prostoru, v němž bude model popsán, protože různé prostory poskytují různou složitost výběru barvy kůže. Existuje velké množství použitých barevných prostorů, nejčastějšími jsou RGB, normalizovaný RGB, YCbCr, HIS, HSV, HSL a další.

#### Výhody

- Rychlé zpracování
- Robustní vůči geometrickým rozdílům tváří
- Robustní vůči částečnému zakrytí tváře
- Invariantní vůči změně rozlišení

#### Problémy

- Pozadí má barvu odpovídající barvě kůže
- Osvětlení tváře se může měnit
- Barva pleti je velice rozmanitá v závislosti na stáří a rase
- Různá snímací zařízení mohou snímat stejný obraz odlišně
- Stíny a okolní světlo mění barvu tváře na snímku

## 2.3 Metody porovnávání se šablonou

Tyto metody používají standardní vzor tváře - šablonu, která je ručně předdefinována nebo parametrizována funkcí. Pak se z této vzorové tváře a daného vstupního obrazu počítají korelační hodnoty pro obrys tváře, oči, nos a ústa nezávisle na sobě. Existence tváře v obraze je pak založena na korelačních hodnotách. Tedy do jaké míry odpovídá vstupní obraz standardní vzorové tváři, šabloně.

Výhoda těchto metod je v jednoduchosti jejich implementace. Přesto je takovýto přístup k detekci nedostačující, neboť je neefektivní v případech, kdy má tvář různé rozměry, tvar nebo orientaci. Pro dosažení nezávislosti na velikosti a tvaru může být použito více šablon tváře (více rozměrů, deformované šablony pro detekci natočení, atd.).

Lanitis a spol. popsali metodu reprezentace tváře pomocí informace o tvaru a intenzitě obrazu [18] . Začali se souborem trénovacích obrazů, pro který byly manuálně označeny vzorové kontury (např. oblast očí, nos, brada/tvář). Vektor takto získaných bodů je použit pro reprezentaci tvaru obličeje. Různost jednotlivých tvarů je vyjádřen pomocí modelu rozdělení bodů (*point distribution model*, PDM) [3] . Ten je pak dále použit pro detekci tváře pomocí modelu aktivního tvaru (*active shape model*, ASM) [4] .

## 2.4 Metody založené na vzhledu

Na rozdíl od metody porovnávání se šablonou, kde šablona byla předdefinována experty podle vlastnosti lidského obličeje a jeho rysů, šablona v těchto metodách je odvozena strojovým učením od ukázkových obrazů. Opírá se o technologii statistické analýzy a již zmíněného strojového učení, pomocí nichž hledá charakteristiky, které odlišují obrazy obsahující tváře a obrazy tváře neobsahující. Naučené charakteristiky mají tvar distribučních modelů nebo rozlišující funkce a jsou použity pro detekci obličeje. Často však vzhledem k velkému množství hodnot je ještě prováděna redukce dimenze (např. PCA, viz kap. 1.5.5), čímž je dosažena lepší výpočetní efektivita a zároveň účinnost detekce. Metod tohoto typu existuje velké množství. Příkladem může být přístup pomocí zde zmíněné metody SVM [21] , metody vlastních tváří (*Eigenfaces*) [16] , jež je založena na PCA, pomocí HMM (*Hidden Markov Model*) [22] nebo následující dva zmíněné přístupy.

### 2.4.1 Neuronová síť

Neuronové sítě byly úspěšně aplikovány v mnoha úlohách rozpoznávání vzorů, jako je rozpoznávání písma nebo objektů. Na detekci obličeje, jak již bylo zmíněno, lze však nahlížet jako na problém rozpoznávání vzorů s tříděním do dvou tříd. Z čehož vyplývá, že lze použít různé architektury neuronových sítí. Jejich nevýhodou je však složité nastavování (počet vrstev, počet uzlů, atp.), aby bylo dosaženo požadované detekce.

Nejvýznamnější metodu využití neuronové sítě použil Rowley a spol. [26] , [25] . Používá vícevrstvou neuronovou síť, jež je trénována skupinou obrazů, které obsahují a neobsahují tváře. Zahrnuje dvě základní komponenty. První tvoří vícenásobná neuronová síť, která slouží k detekci vzoru tváře. Jejím vstupem je výřez obrazu o velikosti 20 x 20 pixelů a výstupem hodnota od -1 do 1, kde -1 indikuje, že ve vstupu se nenachází tvář, naopak 1 indikuje přítomnost tváře na vstupu. Aby bylo možno detekovat tvář kdekoli v obraze, vybírá se jako vstup neuronové sítě postupně každá oblast obrazu (pomocí tzv. okna). K detekci tváří větších než 20 x 20 pixelů je obraz podvzorkován a předchozí přístup použit pro další velikosti. Druhou komponentou toho přístupu je modul rozhodování, který provádí konečné rozhodnutí z více detekcí. Tento výběr může být založen např. na logických operacích (součet, součin). Omezením byla detekce pouze vzpřímených, čelně natočených tváří. Toto bylo odstraněno natáčením okna a prováděním detekcí na takto získaném obrazu [9] .

Dalším příkladem použití neuronové sítě je Viola-Jones detektor. Detailnímu popisu toho přístupu se věnuje kapitola 3.4.

Přestože existuje mnoho metod přístupů k detekci obličeje, nelze vybrat jednu, o které by šlo říci, že je lepší než ostatní. Každá metoda má své pro a proti a záleží na implementaci, která metoda bude pro daný účel výhodnější. Zároveň nelze přesně porovnat všechny metody, neboť dosažené výsledky byly často získány trénováním i testováním na jiných datech.

## 3 Praktická část

Z předchozí kapitoly si lze odnést poznatek, jaké množství možných řešení a přístupů zatím bylo publikováno. Vzhledem k náročnosti byly pro tuto práci vybrány pouze dvě metody. Tato kapitola jim bude věnována podrobněji.

### 3.1 Detektor barvy kůže

Jak již bylo zmíněno, pro detektor barvy kůže je důležitý výběr vhodného barevného prostoru a dále pak modelu barvy kůže. Proto je další kapitola shrnutím nejčastěji použitých barevných prostorů.

#### 3.1.1 Barevné prostory

##### RGB

Jedná se o aditivní barevný model, kde jednotlivé barvy lze získat smícháním tří základních barev (*Red - Green - Blue*, tedy červená, zelená, modrá). Tento model je jedním z nejvíce používaných při zpracování a uložení digitálních dat, neboť se kromě počítačových formátů používá ve většině monitorů a projektorů.

Jeho výhodou je, že data získaná z kamery již jsou ve formátu RGB, takže není nutná žádná další transformace, která by prodlužovala dobu klasifikace - jako tomu je u jiných prostorů. Vzhledem k vysoké korelaci jednotlivých kanálů, kdy jsou chrominanční a luminanční data smíchána, by se mohlo zdát, že není příliš vhodný pro tento účel. Přesto však s ním lze dosáhnout uspokojivých výsledků a je použit i v této práci.

## Normalizovaný RGB

Normalizovaný RGB prostor lze jednoduše získat z RGB modelu normováním jeho jednotlivých složek:

$$r = \frac{R}{R + G + B} \quad g = \frac{G}{R + G + B} \quad b = \frac{B}{R + G + B} \quad (3.1)$$

Protože součet tří normovaných složek je známý ( $r + g + b = 1$ ), poslední složka nenese žádnou informaci, tudíž jí lze vynechat, čímž se sníží prostorová dimenze dat. Zbylé dvě složky bývají označovány jako „čisté barvy“ (*pure colors*), protože normalizací je snížena jejich závislost na jasu. Z čehož vyplývá důležitá vlastnost, že pro matné povrchy je tento model invariantní (pod určitými předpoklady) vzhledem k orientaci povrchu ke světelnému zdroji.

## YCbCr

Tento prostor je všeobecně používán v televizní technice a při kompresi digitálních fotografií. Odděluje od sebe luminanční složku Y, někdy označovanou jako „luma“ a chrominanční složky Cb a Cr. Lze jej získat nelineárním převodem z RGB modelu:

$$\begin{aligned} Y &= 0,299R + 0,587G + 0,114B \\ Cr &= R - Y \\ Cb &= B - Y \end{aligned} \quad (3.2)$$

Právě toto oddělení složek nesoucích barevnou informaci od složky, která představuje vyjádření jasu, je pro detekci barvy tváře žádané.

### 3.1.2 Model barvy kůže

Model barvy kůže v této práci je založen na explicitním vyjádření barvy kůže. Tato metoda spočívá v definici pravidel, jimiž se ohraničí oblast barvy kůže v barevném prostoru. Obrovskou výhodou této metody je výrazná jednoduchost detekce a tedy zároveň i s ní související vysoká rychlost. Její přesnost závisí na výběru barevného prostoru a kvalitě empiricky definovaných explicitních pravidel. K získání těchto empirických dat, lze použít strojového učení, jako v případě [17].



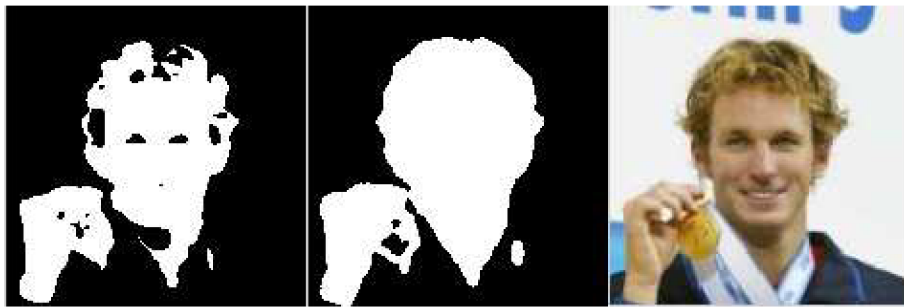
Byly vybrány dva modely barvy kůže, které se často doplňují, jak lze vidět např. na obr. 3.1 a obr. 3.2 (levý obrázek je detekován pomocí RGB modelu, prostřední YCbCr modelu, poslední je originál obrázku). Snímky jsou vybrány z databáze *Labeled Faces in the Wild* Massachusettské univerzity.

Ohraničení barev v barevném modelu RGB na základě [15] :

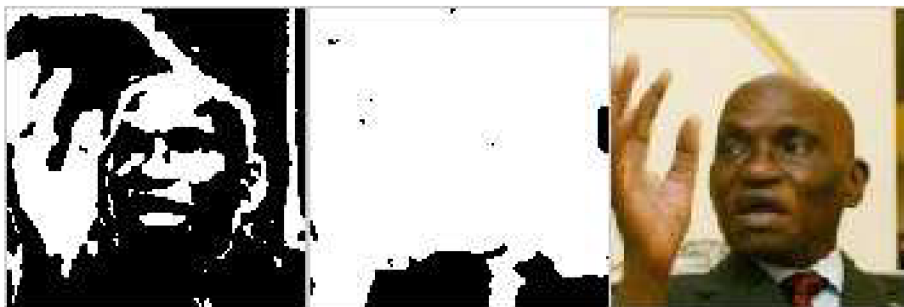
$$\begin{aligned}
 R &> 95, G > 40, B > 20 \\
 \text{Max}\{R, G, B\} - \text{min}\{R, G, B\} &< 15 \\
 |R - G| > 15, R > G, R > B
 \end{aligned}
 \tag{3.3}$$

Ohraničení barvy kůže v barevném modelu YCbCR na základě [5] :

$$77 \leq Cb \leq 127 \cup 133 \leq Cr \leq 173
 \tag{3.4}$$



Obr. 3.1: Úspěšná detekce barvy kůže



Obr. 3.2: Ukázka selhání detektoru kůže

## 3.2 AdaBoost

Jedná se o metodu, které zde bude věnována největší pozornost (pro její přesnost detekce a zejména rychlost), a která bude implementována v rámci této a další práce. AdaBoost [31] je algoritmus, který vychází z metody strojového učení zvaného boosting (viz kap. 1.5.4), název je zkratkou pro *Adaptive Boosting*. K učení využívá slabých klasifikátorů  $h_t(x)$ , které jsou vybírány z množiny klasifikátorů  $\mathbf{H}$ , a jejichž lineární kombinací vzniká nelineární silný klasifikátor  $H(x)$ , viz algoritmus 1. Vstup algoritmu tvoří trénovací množina  $S$ , která je složena z dvojic  $(x_i, y_i)$ , kde  $x_i$  je získaná hodnota příznaku a  $y_i$  je třída odpovídající příznaku  $y_i \in \{-1; 1\}$ ,  $i=1, \dots, m$ , kde  $m$  je velikost trénovací množiny. AdaBoost na rozdíl od boostingu používá ovážení trénovací množiny váhami  $D_t$ , které jsou na začátku nastaveny rovnoměrně, viz Algoritmus 1, krok 2. V každé smyčce algoritmu se pak:

- vybírá slabý klasifikátor s nejmenší chybou při daných váhách  $D_t$
- ověří, že chyba nepřekročila hodnotu 0,5
- vypočte koeficient slabého klasifikátoru v lineární kombinaci  $H(x)$
- aktualizují váhy trénovací množiny  $D_t$

Výraz  $I[\text{výrok}]$  vrací 1, pokud je výrok pravdivý a 0 pokud je *výrok* nepravdivý. Slabý klasifikátor je vybírán tak, aby jeho klasifikace byla jen o něco lepší, než kdyby klasifikoval náhodně. Pokud tedy chyba  $\varepsilon_t$  překročí hodnotu 0,5, pak by takový požadavek nebyl splněn a nebylo by zaručeno, že algoritmus bude konvergovat. Aktualizace vah (viz algoritmus 1, krok d) způsobí to, že **váha špatně klasifikovaných měření se zvětší** a váha dobře klasifikovaných se zmenší. V následujícím kroku bude tedy hledán slabý klasifikátor, který bude muset lépe klasifikovat doposud chybně provedená měření [33]. Adaboost redukuje trénovací chybu exponenciálně v závislosti na rostoucím počtu klasifikátorů. Zvyšování počtu klasifikátorů ve skupině ale může vést k tzv. **přetrénování** (ztrátě schopnosti generalizovat vlivem přílišného zaměření klasifikátorů na rozeznávání pouze konkrétních trénovacích dat). Simulační experimenty však ukázaly, že k tomu relativně zřídka dochází i pro extrémně vysoké hodnoty [24].

1. Vstup:

$$S = \{(x_1, y_1), \dots, (x_m, y_m)\}, \text{ počet iterací } T$$

2. Inicializace vah:

$$D_1(i) = \frac{1}{m}$$

3. Cyklus pro  $t = 1, \dots, T$ :

a. Výběr klasifikátoru na základě vážené trénovací chyby

$$\varepsilon_j = \sum_{i=1}^m D_t(i) I [y_i \neq h_j(x_i)]$$

$$h_t = \arg \min_{h_j \in H} \varepsilon_j$$

b. Pokud

$$\varepsilon_t = 0 \text{ nebo } \varepsilon_t \geq \frac{1}{2}, \text{ pak konec cyklu}$$

c. Nastavení

$$\alpha_t = \frac{1}{2} \log \left( \frac{1 - \varepsilon_t}{\varepsilon_t} \right)$$

d. Úprava vah

$$D_{t+1}(i) = \frac{D_t(i) e^{-\alpha_t y_i h_t(x_i)}}{Z_t}$$

$$\text{kde } Z_t = \sum_{i=1}^m D_t(i) e^{-\alpha_t y_i h_t(x_i)}$$

4. Výstup

$$H(x) = \text{sign} \left( \sum_{t=1}^T \alpha_t h_t(x) \right)$$

Algoritmus 1: Diskrétní AdaBoost

### 3.3 Učení perceptronu

Slabý klasifikátor v případě AdaBoostu, může být jakýkoliv algoritmus, který rozděluje množinu na pozitivní a negativní případy s chybou menší než 50%. Přestože se mluví o jakémkoliv algoritmu, není to tak úplně pravda, protože právě výkonnost slabého klasifikátoru významně ovlivňuje celkovou výkonnost klasifikace. Proto je častým požadavkem na slabý klasifikátor jednoduchost a přesnost.

V této práci byla pro rozhodování slabého klasifikátoru vybrána jednoduchá dopředná neuronová síť - perceptron. Jak již bylo zmíněno, perceptron klasifikuje vstupní vzorky  $x_i$  na základě skalárního součinu mezi nimi a váhovým vektorem  $w$ :

$$h(x_i) = \text{sign}(w \cdot x_i) \quad (3.5)$$

Proces trénování spočívá v úpravě vah  $w$  a to na základě dvojic z trénovací množiny  $S = \{(x_1, y_1), \dots, (x_n, y_n)\}, y_i \in \{-1; 1\}$ . Tato úprava se provádí v případě, že byl vzorek špatně klasifikován. Původní algoritmus (Rosenblatt, [24]) upravoval váhy následovně:

$$\begin{cases} \tilde{w} = w + yx_i & \text{pokud } h(x_i) \neq y_i \\ \tilde{w} = w & \text{pokud } h(x_i) = y_i \end{cases} \quad (3.6)$$

Toto pravidlo se postupně opakuje pro všechny dvojice z  $S$ . Pokud jsou prvky trénovací množiny lineárně separovatelné, tak po konečném počtu opakování je nalezen váhový vektor, který správně klasifikuje každý vzorek z trénovací množiny. Chyba na trénovací množině bude tedy nulová  $E_{\text{train}}$ .

Situace se komplikuje v případě, kdy trénovací množina obsahuje prvky, které lineárně separovatelné nejsou. Pro tento případ uvedený způsob úpravy vah k řešení nevede. Možným řešením tohoto problému je algoritmus *Pocket algorithm with ratchet* (Gallant, [10]), jehož podstatou je uložení váhového vektoru, který zůstal nezměněn pro nejvíce iterací a současně dosáhl minimální chyby  $E_{\text{train}}$ .

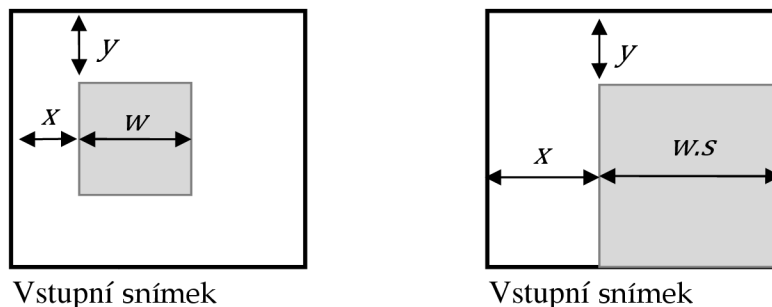
### 3.4 Viola-Jones detektor

Viola-Jones detektor [33] je založen na algoritmu AdaBoost a je upraven tak, aby bylo možno provádět úlohu detekce obličeje. Slabý klasifikátor používá příznak podobný **Haarově vlnce** (*Haar-like feature*, viz obr. 3.3). Hodnota takového příznaku se pak vypočítá jako suma pixelů obrazu odpovídající světlé části, od které je odečtena suma pixelů, které pokrývá tmavá část. Tyto vlnky mohou být tvořeny dvěma (hranový příznak), třemi (čárový příznak) nebo čtyřmi (diagonální příznak) obdélníkovými oblastmi.



Obr. 3.3: Příznaky podobné Haarově vlnce (haar-like feature)

Vstupní obraz není zpracováván jako celek, ale po částech. Tyto části jsou vybírány pomocí posuvného pod-okna, které nejen že mění svou pozici, ale i velikost, přičemž se respektuje určitý krok mezi pozicemi a měřítkem. Příznaky jsou vhodně umístěny v tomto pod-okně. Vysvětlit se to snaží obr. 3.4, kde  $x$ ,  $y$  jsou souřadnice posunu pod-okna,  $w$  je jeho základní šířka a  $s$  je faktor zvětšení. Pod-okno je vyplněno šedě.



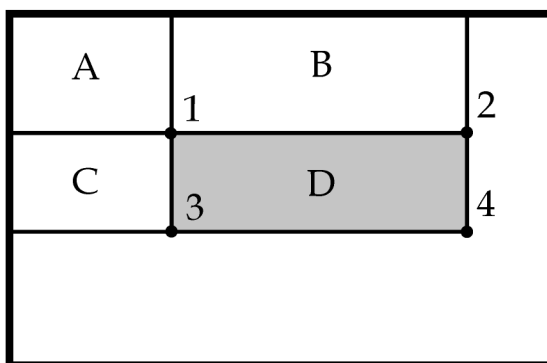
Obr. 3.4: Umístění pod-okna v původním obrázku

Aby suma hodnot pixelů nemusela být počítaná pro každý příznak, je obraz převeden do reprezentace, která umožňuje její efektivní výpočet. Je nazvána **integrální obraz** (*integral image*) a každý bod takového obrazu odpovídá součtu hodnot všech předcházejících bodů podle rovnice (3.7).

$$\begin{aligned} s(x, y) &= s(x, y - 1) + I(x, y) \\ I_{\text{int}}(x, y) &= I_{\text{int}}(x - 1, y) + s(x, y) \end{aligned} \quad (3.7)$$

kde  $s(x, y)$  je kumulovaný součet hodnot v řádku obrazu,  $I(x, y)$  představuje hodnoty původního snímku a  $I_{\text{int}}(x, y)$  je integrální obraz. Dále pak platí  $s(x, -1) = 0$  pro  $\forall x$  a  $I_{\text{int}}(-1, y) = 0$  pro  $\forall y$ .

Výpočet příznaků se tak výrazně zefektivní, protože výpočet sumy intenzit v obraze se zjednoduší na dvě operace sčítání a jednu odčítání. Součet hodnot pixelů obrazu uvnitř oblasti  $D$  (viz obr. 3.5) je počítán pomocí bodů 1, 2, 3 a 4 (bod 1 vznikl součtem všech intenzit v oblasti  $A$ , bod 2 vznikl součtem hodnot pixelů v  $A$  a  $B$ , atd.) integrálního obrazu pomocí vztahu  $\sum_D = 4 + 1 - (2 + 3)$ .



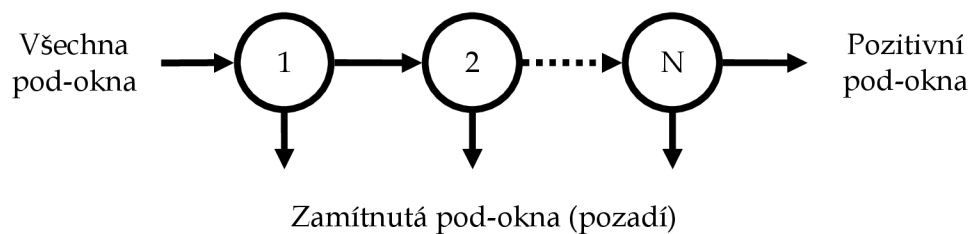
Obr. 3.5: Suma pixelů počítaná pomocí integrálního obrazu

Takto vypočtená hodnota příznaku tvoří slabý klasifikátor podle vzorce (3.8).

$$h_j(Z) = \begin{cases} 1 & \text{pokud } p_j f_j(Z) < p_j \theta_j \\ 0 & \text{v ostatních případech} \end{cases} \quad (3.8)$$

kde  $f_j(x)$  je odezva příznaku,  $\theta_j$  je práh,  $p_j$  je parita signalizující orientaci znaménka nerovná se a  $j \in \mathbb{Z}, j = 1, \dots, J$ , kde  $J$  označuje celkový počet příznaků.

Tím, že je snímek prohledáván pod-okno za pod-okno, stává se detekce výpočetně náročná (např. pro obraz o rozměrech 320 x 240 pixelů je nutno zpracovat přes 500 000 pod-oken). Snaha snížit trvání celkové detekce je řešena snížením průměrné doby, kterou detektor věnuje prohledávání každého pod-okna. Toho je dosaženo **kaskádovým zapojením klasifikátorů** [8]. Myšlenka tohoto řešení je založena na pozorování, že k vytvoření klasifikátoru, který dokáže vybrat téměř všechny pozitivní případy a zároveň množství (20 - 50%) negativních případů, stačí jen pár příznaků. Dalším důležitým postřehem pro vytvoření kaskády, je skutečnost, že většina pod-oken v obraze jsou negativní, tedy neobsahují tvář. Proto se první stupně kaskády snaží zamítnout co nejvíce negativních pod-oken, zatímco do dalších stupňů kaskády přechází jen pod-okna označena za pozitivní, viz obr. 3.6.



Obr. 3.6: Obrázek kaskády

Při tvorbě kaskády (viz algoritmus 2) je potřeba určit dovolenou senzitivitu jednotlivých stupňů  $TPR_D$ , dovolenou  $FPR_D$  a požadovanou (cílovou)  $FPR_P$  nespecifitu. Každý stupeň kaskády je trénován pomocí AdaBoostu. Trénování začíná s jedním příznakem, přičemž se přidávají další příznaky, dokud není dosaženo  $TPR_D$  a  $FPR_D$ . AdaBoost je schopen minimalizovat chybu, ale není navržen k tomu, aby jím bylo dosaženo vysoké senzitivity na úkor větší nespecifity. To lze zajistit prahováním výstupní hodnoty silného klasifikátoru  $H(x)$  tak, že pokud

$$H(x) \geq \theta, \text{ pak } x \text{ je tvář,} \quad (3.9)$$

kde  $x$  je hodnota příznaku a  $\theta$  je práh.

Snižování prahu způsobuje zvyšující se detekční poměr, ovšem za cenu zvýšení chyb I. typu (chybné detekce tváří). Senzitivita a nespecifita jsou získány testováním detektoru na testovací množině. Další klasifikátory v kaskádě jsou trénovány na základě množiny, která úspěšně prošla přes předchozí úroveň. Zároveň mají těžší úlohu, protože musí přesněji vybrat pozitivní a negativní případy.

1. **Vstup:** senzitivita  $TPR_D$ , dovolená  $FPR_D$  a požadovaná  $FPR_P$

2. **Inicializace:**  $F_0 = 1, D_0 = 1$

3. **Opakuj dokud**  $F_i < FPR_P$

a. Trénování klasifikátoru pomocí AdaBoost dokud

$$\begin{aligned} FPR_i &< FPR_D \\ TPR_i &> TPR_D \end{aligned}$$

b.  $F_{i+1} = F_i \times FPR_i$

c.  $D_{i+1} = D_i \times TPR_i$

d. Zrušení chybně detekovaných tváří a vytvoření nových dat ze snímků, co neobsahují tváře.

#### Algoritmus 2: Postup pro vytvoření kaskády

Celková citlivost kaskády  $TPR_{\text{CELK}}$  je dána kombinací detekčních poměrů  $TPR_i$  všech stupňů  $N$ , podle vzorce (3.10). Obdobně se počítá celkový chybný pozitivní poměr [33].

$$TPR_{\text{CELK}} = \prod_{i=1}^N TPR_i \quad (3.10)$$

Celková nespecifita  $FPR_{\text{CELK}}$  je vyjádřena obdobně pomocí jednotlivých  $FPR_i$ :

$$FPR_{\text{CELK}} = \prod_{i=1}^N FPR_i \quad (3.11)$$



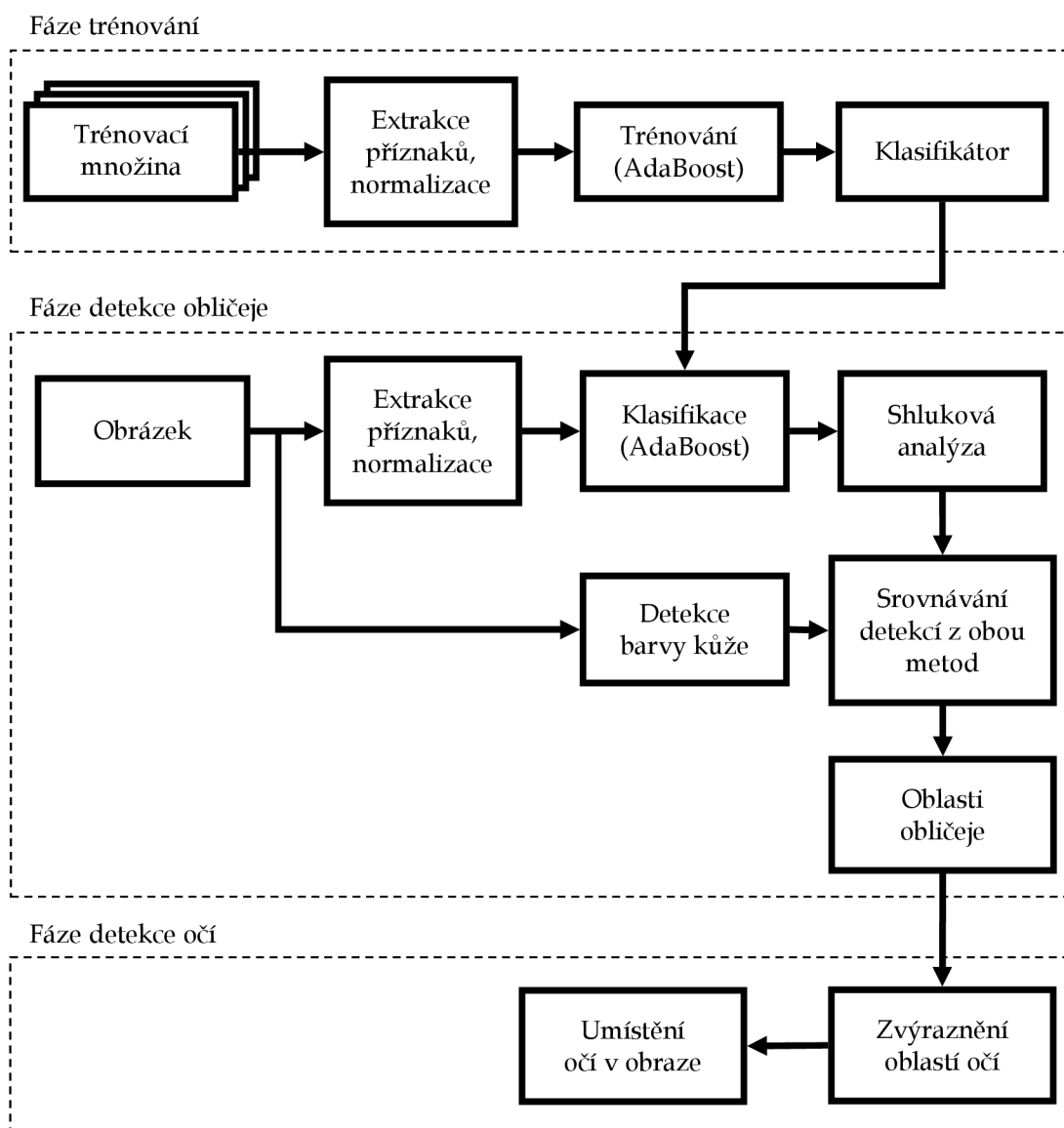
## 3.5 Realizace detektoru

Program implementuje detektor barvy kůže a Viola-Jones detektor, který je použit pro detekci jak obličej, tak následně i očí. Obecně se detekce barvy kůže řadí mezi úkoly předzpracování. V této práci je jí však přisouzena jiná role, což bude vysvětleno v dalším textu. Jednotlivé kroky programu jsou zobrazeny na obr. 3.7.

První krok detekce tvoří fáze trénování. Z trénovací množiny jsou vyextrahovány příznaky a zároveň jsou normalizovány, jak je popsáno v kap. 3.5.1. Každému příznaku odpovídá jeden perceptron, který ve fázi trénování nastavuje své váhy podle trénovacích dat. Z těchto slabých klasifikátorů jsou pak některé vybrány a AdaBoostem kombinovány v jednotlivých stupních kaskády – silných klasifikátorů (viz kapitola 3.4). Vzniklá struktura kaskády představuje kýžený klasifikátor. Fáze trénování stačí provést zpravidla jednou. Poté je k samotné detekci potřeba pouze soubor popisující kaskádu, viz obr. 3.8.

V druhém kroku detekce vstupují do systému obrázky a úkolem detektoru je nalézt v nich všechny tváře. Postup je takový, že se ze vstupního snímku extrahují příznaky, které jsou dále klasifikovány. Výsledkem této klasifikace jsou oblasti, kde byl detekován obličej. Protože je trénovací množina pouze šedotónová, pro extrakci příznaků je třeba obraz převést do odstínu šedi. To jednoduše zajistí funkce `rgb2gray()`, která je součástí prostředí MATLAB (přesněji jeho součást *Image Processing Toolbox*). Zároveň probíhá v obrázku detekce barvy kůže, jejímž výsledkem je binární mapa, jak je vidět např. na obr. 3.1. Detekce tváře je pak ověřena vzhledem k detekci barvy kůže. Pokud se shodují, je část obrazu prohlášena za tvář.

Třetí krok zpracovává už jen oblasti detekovaných tváří z předchozího kroku. V nich pak provádí úpravu podle kapitoly 3.5.4, čímž se oči zvýrazní. V takto zvýrazněných místech je pak aplikován opět Viola-Jones detektor.



Obr. 3.7: Jednotlivé kroky programu

### 3.5.1 Efektivní normalizace rozptylu v obraze

Z důvodu snížení vlivu osvětlení na detekci, je rozptyl každého pod-okna trénovací množiny normalizován. Rozptyl  $\sigma^2$  integrálního obrazu  $I_{\text{int}}(x, y)$  se vypočítá podle (3.12), kde  $E$  představuje střední hodnotu.

$$\sigma^2(I_{\text{int}}(x, y)) = E(I_{\text{int}}^2(x, y)) - E(I_{\text{int}}(x, y))^2 \quad (3.12)$$

Hodnota příznaku v pod-okně je pak normalizována pomocí rozptylu následovně:

$$f'_j(\mathbf{Z}) \leftarrow \frac{f_j(\mathbf{Z})}{\sigma^2}, \quad (3.13)$$

kde  $f_j(\mathbf{Z})$  je hodnota  $j$ -tého příznaku v pod-okně  $\mathbf{Z}$  a  $f'_j(\mathbf{Z})$  je hodnota normalizovaného příznaku. Na základě normalizovaných příznaků byl detektor trénován.

### 3.5.2 Adaptační fáze - trénování

Základní velikostí pod-okna, která se shoduje s velikostí snímků trénovací množiny, je 19 x 19 pixelů. Počet příznaků v tomto okně odpovídá číslu 53 130. Typy příznaků použité v programu odpovídají těm na obr. 3.3, s výjimkou toho, že je použit pouze jeden čárový příznak. K trénování i testování vlastností detektoru tváře byla použita databáze MIT CBCL Face Database, viz tab. 1.1. Ukázka snímků z této databáze pro prvních 100 pozitivních i negativních vzorků je na obr. 1.12.

Vzhledem k obrovskému množství příznaků v základním pod-okně a velikosti databáze (počet tváří 2429, pozadí 4548), je jasné, že fáze trénování je časově i výkonově náročná. Z důvodů již zmíněné extrémní výpočetní náročnosti bylo shledáno programové prostředí Matlab k implementaci trénovací fáze zcela nevyhovující. To vedlo k vytvoření programu pro trénování detektoru v jazyce C/C++, který byl schopen provést fázi trénování mnohem rychleji.

Výstupem fáze trénování jsou soubory

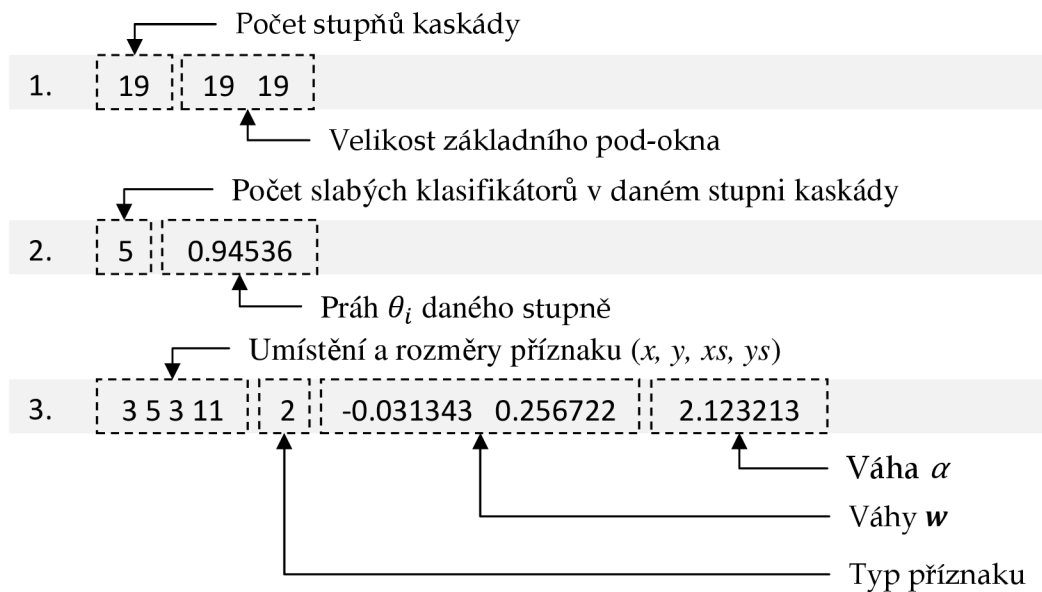
- *Acc.txt* – představuje přesnost výstupního klasifikátoru na trénovací množině dat. Tři čísla na prvním řádku představují celkovou přesnost *ACC*, úspěšnost detekce pozitivních vzorů (obličejů) a úspěšnost negativních vzorů (pozadí). Na dalších řádcích následuje výčet chybně klasifikovaných vzorů.
- *Classifier.txt* – popis nastavení kaskády, použit v pracovní fázi
- *Features\_struct.txt* – obsahuje popis všech příznaků v základním pod-okně
- *Input\_data.txt* – trénovací množina
- *Results.txt* – zobrazuje průběh trénování (přesnosti v jednotlivých)

Program obsahuje 3 základní struktury, které popisují jednotlivé komponenty algoritmu. *InputData* obsahuje integrální obraz, druhou mocninu integrálního obrazu (ta je potřebná pro výpočet rozptylu, viz (3.12)) a třídu, do které obraz náleží. *WeakClassifier* nese informace o vstupních vahách perceptronu a jeho chybě, zároveň obsahuje pořadové číslo, aby bylo možno jej přiřadit k danému příznaku. *StrongClassifier* představuje jednotlivé příznaky kaskády. Obsahuje navíc váhu slabého klasifikátoru, přesnost a popisuje příznak, pro který je natrénován.

```
1.  struct InputData {
2.      int **piIntegrallImage;
3.      float fSqrIntegral;
4.      char cLabel;
5.  };
6.  struct WeakClassifier {
7.      float fWeights [2];
8.      float fEpsilon;
9.      int iIndex;
10. };
11. struct StrongClassifier {
12.     float fWeights [2];
13.     float fAlphas;
14.     float fAccuracy;
15.     int piPosition[4];
16.     int iType;
17. };
```

### 3.5.3 Pracovní fáze

Důležitým výstupem trénovací fáze je soubor *Classifier.txt*. Ten popisuje jednotlivé stupně kaskády. Pro každý stupeň je uveden rozhodovací práh  $\theta$ , pro který platí (3.9). Dále jsou v rámci každého stupně popsány použité příznaky, které se podílí na vytvoření silného klasifikátoru. U každého včetně jejich váhy ve výsledném klasifikátoru, viz obr. 3.8.



Obr. 3.8: Popis prvních tří řádků souboru *Classifier.txt*

Před samotným spuštěním detekce je třeba tento soubor načíst do prostředí MATLAB pomocí skriptu *LoadClassifier.m*. Tím se vytvoří klasifikátor AdaBoost. Dále je třeba načíst obrázek do proměnné *Input*, po té je možno spustit samotnou detekci.

```
1. Classifier=LoadClassifier('Classifier.txt');  
2. Input = imread('img\obr02.jpg');  
3. Class = FaceDet(Input,Classifier);
```

Na základě parametrů klasifikátoru jsou extrahovány a vyhodnocovány příznaky.

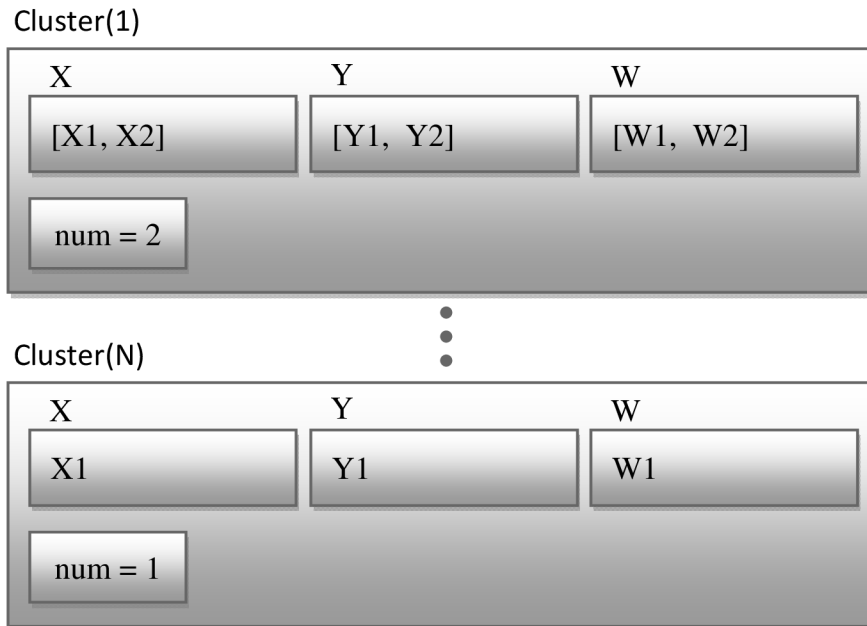
V obrazu mohou vznikat mnohonásobné detekce, což znamená, že tvář je zahrnuta v několika překrývajících se oblastech, které jsou od sebe jen málo posunuty nebo mají podobnou velikost. Stejně tak se mohou, ovšem v menší míře, vyskytnout chybně detekované oblasti. Toho lze využít k efektivnímu zjištění, zda se tvář na daném místě opravdu vyskytuje.

Aplikace se s mnohonásobnými detekcemi vyrovnává pomocí **shlukové analýzy** (*cluster analysis*), což je metoda dělení **jednotek** do **kategorií** (shluků) tak, aby si jednotky náležící do jedné kategorie byly podobnější než objekty z ostatních kategorií. Toto dělení se provádí na základě určitých ukazatelů. Příkladem těchto ukazatelů mohou být korelační koeficienty, metriky a jiné. V této práci byla jako metrika použita **euklidovská vzdálenost**

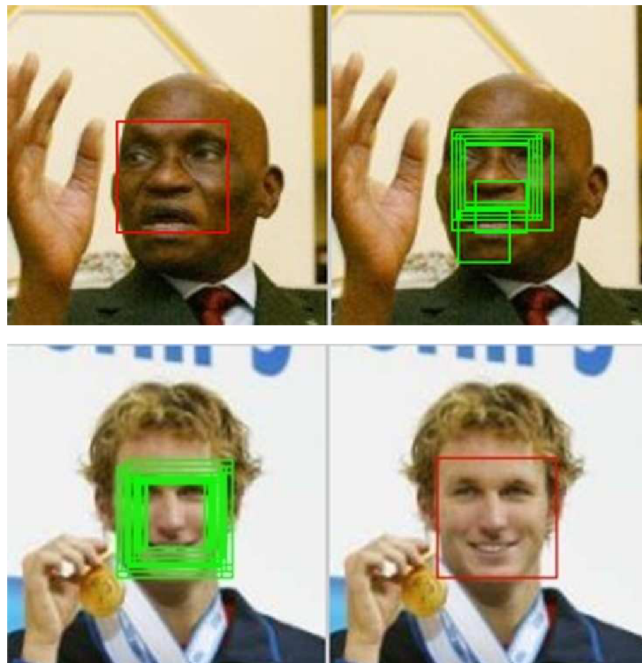
$$D_E((x_1, y_1), (x_2, y_2)) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (3.14)$$

, kde  $(x_1, y_1), (x_2, y_2)$  představují souřadnice dvou obrazových bodů.

Vzhledem k tomu, že každá detekce Viola-Jones detektorem představuje vytvoření tripletu  $[X, Y, W]$ , kde  $X$  a  $Y$  představují obrazové souřadnice středu detekovaného pod-okna a  $W$  jeho šířku. První triplet vytvoří první kategorii shluku Cluster(1), viz obr. 3.9. Pokud nastane další detekce, program zkoumá metriku. K porovnávání metriky slouží proměnná PosDist a jedná se vlastně o maximální přípustné oddálení středů detekovaného pod-okna od středů, které jsou uloženy v proměnné Cluster. Kromě metriky je pro každou detekci také zkoumán poměr velikosti okna vzhledem k velikosti pod-okna uložené v každé jednotce každé kategorie. Jestliže je tento poměr menší než WinDist, a zároveň pokud je splněna předchozí podmínka, pak je detekované pod-okno zahrnuto do aktuální kategorie. V dalším kroku jsou hodnoty z každé kategorie průměrovány. Proto se uvažují jen kategorie, které mají více jednotek než je hodnota proměnné ClustersReq. Tento průměr je opět podroben shlukové analýze, jak byla popsána výše. Výsledkem bude struktura CulCluster, která bude obsahovat už jen pár kategorií, popisujících pravděpodobný výskyt tváře v obraze. Tímto krokem tedy došlo ke snížení dimenze dat, viz obr. 3.10.



Obr. 3.9: Znárodnění struktury proměnné Cluster



Obr. 3.10: Ukázka odstranění mnohonásobné detekce

CulCluster nyní obsahuje možné kandidáty na tváře. Pod-okna, které popisuje, jsou podrobeny detekci barvy kůže, jak je popsána v kap. 3.1.2. Pro zlepšení výsledků je snímek rozostřen, čímž se zjemní barevný přechod oblastí. Pomocí obou detektorů barvy kůže se vypočítají binární masky (maskaRGB a maskaYCBCR), sečtou se s 50% zastoupením, dále se počítá suma tohoto součtu a ta se porovnává s hodnotou maximálního počtu pixelů v pod-okně. Dovolenou chybu představuje proměnná ColorTresh. Pokud dané pod-okno obsahuje dostatečný počet detekovaných oblastí barvy kůže, je prohlášeno za obličej a postoupeno do poslední fáze programu.

Následuje přehled parametrů a jejich typických hodnot, jež lze v rámci programu nastavit

```
1. Rect_size=19;
2. scale_fact=1.25;
3.
4. Min_Face=19;
5. Max_Face=100;
6.
7. PosDist=10;
8. WinDist=1.3;
9. ClustersReq=2;
10. ColorTresh=0.7;
11. ResFactor=0.2;
```

Rect\_size je základní velikost pod-okna a proměnná scale\_fact představuje koeficient jejího zvětšování. Další pod-okna se tedy zvětší vždy o 25 %. Min\_Face a Max\_Face jsou minimální a maximální uvažovaná velikost tváře v obraze. Pohledem na vzory trénovací množiny (viz obr. 1.12), lze vidět, že jsou tváře těsně ořezány. Pokud je požadavek detekce celé hlavy, pak je dobré detekované pod-okna zvětšit. Proměnná ResFactor představuje zvětšení o 20 %.



### 3.5.4 Fáze detekce očí

Obličejové rysy mají široké pole využití, mohou sloužit k automatickému modelování tváře, animaci tváře, rozpoznávání výrazu tváře, a dalším. Oči, jako jeden z hlavních obličejových rysů, hrají důležitou roli při rozpoznávání tváře a analýze výrazu tváře. Ve srovnání s ostatními obličejovými rysy je možno oči považovat za relativně stabilní. Proto je užitečné, při detekci obličejových rysů, detekovat nejprve oči a poté ostatní rysy. Ty totiž mohou být odhadnuty na základě pozice očí [35].

Úkolem detekce pozice očí je získat souřadnice místa v obraze, kde se oči nachází. Současné metody pro určení pozice očí lze rozdělit do dvou kategorií:

- **Aktivní přístup** je založený na speciálním typu osvětlení. Pod infračerveným (IR, *infrared*) osvětlením se projevují fyzické vlastnosti zorniček, které mohou být využity k lokalizaci. Výhodou této kategorie je robustnost a přesnost jejích metod. Nevýhodou je požadavek speciálního zdroje světla a zvýšení chybných detekcí při venkovním použití, kde je IR zdroj ovlivněn venkovním světlem [29].
- **Pasivní metody** detekují oči ze snímků za běžného osvětlení. Přístupy k tomuto problému jsou podobné jako přístupy k detekci tváře. Příkladem mohou být přístupy založené na vytvoření šablony, kdy je vytvořena šablona oka, s kterou se porovnávají oblasti v obraze. Přestože tento přístup může dosahovat příznivých výsledků, je náročný na čas [35].

Před samotným procesem detekce očí je provedeno **zvýraznění snímku** tak, jak bylo popsáno v [14]. Tato operace se týká pouze pod-okna. Nejprve se pomocí chrominančních složek vytvoří mapa

$$EyeMapC = \frac{1}{3} \left\{ (C_b^2) + (\tilde{C}_r)^2 + \left( \frac{C_b}{C_r} \right) \right\}, \quad (3.15)$$

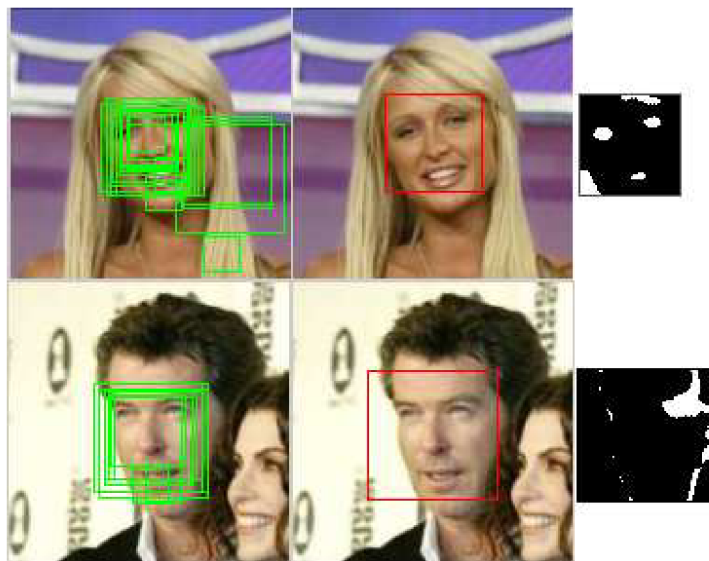
kde  $C_b^2$ ,  $(\tilde{C}_r)^2$ ,  $\frac{C_b}{C_r}$  jsou normované v rozsahu  $[0, 255]$  a  $\tilde{C}_r$  je doplněk  $C_r$  (tzn.  $255 - C_r$ ).

Dále se vytvoří mapa i z lumenanční složky

$$EyeMapL = \frac{Y(x, y) \oplus g_{\sigma}(x, y)}{Y(x, y) \ominus g_{\sigma}(x, y) + 1} \quad (3.16)$$

kde  $\oplus$  označuje šedotónovou dilataci,  $\ominus$  šedotónovou erozi a strukturní element  $g: \mathcal{G} \subset \mathbb{R}^2 \rightarrow \mathbb{R}$ .

Obě mapy se nakonec vynásobením skombinují. Tato výsledná mapa se dále pomocí binárních morfologických operací upraví a na základě prahu se vytvoří binární mapa, která je základem pro detekci očí. Protože v rámci této práce byl implementován Viola-Jones detektor. Bylo snahou použít jej i v případě detekce očí. Pro tento účel byla vytvořena databáze očí. Tato trénovací množina vznikla výřezem oblastí očí ze snímků databáze BioID (viz tab. 1.1) na základě popisu databáze. Při trénování AdaBoostu na těchto vzorcích však výsledky nebyly optimální, proto by k tomuto účelu bylo lépe použít kvalitnější databázi. Taková databáze však zatím neexistuje. Funkčnost detekce očí by ovšem byla stejná jako při detekci obličejů. S tím rozdílem, že se neprohledává celý obraz, ale pouze místa označená jako tváře v prvních dvou fázích aplikace, které navíc odpovídají oblastem zvýrazněných očí. Ukázky zvýraznění spolu s další ukázkou vyrovnání se s vícenásobnou detekcí jsou na obr. 3.11.



Obr. 3.11: Ukázka správně a chybně zvýrazněných očí

## 3.6 Experimentální výsledky

Fáze trénování je jednoznačně náročnější, časy dosažené při trénování v závislosti na počtu příznaků jsou vypsány v tab. 3.1. Při trénování kaskády byly použity hodnoty  $TPR_D = 0,99$  a  $FPR_D = 0,3$ . Výsledná kaskáda měla 19 stupňů. Doba detekce v závislosti na velikosti vstupního obrázku je v tab. 3.2. Trvání detekce je ovlivněno tím, že bylo implementováno v prostředí MATLAB. Rychlost trénovací části napsané v jazyce C/C++ podporuje předpoklad, že lze čas potřebný k detekci snížit.

Tab. 3.1: Závislost doby trénování na velikosti trénovací množiny

Počet pozitivních vzorů	Počet negativních vzorů	Přibližný čas trénování [min]
500	500	1
1000	1000	5
2429	4548	19

Pozn.: testováno na Intel Core 2 Duo, T9300, 2,5 GHz

Tab. 3.2: Závislost doby detekce na velikosti obrázku

Velikost obrázku	Doba detekce [sec]
150x150	13
350x350	81
500x500	130

Detektor vykazuje na trénovací množině  $TPR \cong 0.9$  a  $FPR \cong 0.1$ . Trénovací množina se z pohledu testování výsledků klasifikátoru jeví jako malá, neboť míra přesnosti klasifikátoru na testovacích datech není tak příznivá. Ukázky správných i chybných detekcí jsou umístěny v přílohách. Obrázky jsou vybrány z databáze *Labeled Faces in the Wild*, viz tab. 1.1.

## 4 Závěr

Tato práce se snaží nahlížet na problém detekce obličeje z komplexního hlediska a postupuje tak od obecných poznatků k detailnějším. V první kapitole proto byly nastíněny obecné vztahy mezi základními pojmy, jako je umělá inteligence, počítačové vidění a strojové učení, rozpoznávání vzorů, zpracování obrazu a detekce tváře. Byl zmíněn a popsán obecný systém rozpoznávání vzorů. Jako příklad byl uvedený stručný popis některých metod strojového učení. Byly popsány jednotlivé parametry pro hodnocení detekce a zároveň popsána metoda určování kvality detekce. Druhá kapitola již byla soustředěna pouze na známé publikované metody. Tyto byly rozděleny do čtyř tříd: metody založené na znalosti, na invariantních rysech, na vzhledu a metody porovnávání se šablonou. Byli zmíněni nejvýznamnější zástupci těchto metod. Zvláštní pozornost byla věnována barevnému modelu kůže a neuronové síti. Poslední kapitola se zaměřuje pouze na popis důležitých součástí realizovaného detektoru. Hlavní pozornost byla věnována Viola-Jones detektoru, který vychází ze čtyř hlavních komponent: algoritmu strojového učení AdaBoost, vyjádření snímku pomocí integrálního obrazu, výběr příznaků podobných Haarově vlnce a použití kaskádového zapojení, jejichž spojením se dosáhlo dobrých vlastností (nízké výpočetní náročnost, rychlosti a robustnosti detekce). Zároveň byly zmíněny dva typy možnosti detekce barvy kůže v obraze. Dále byl samotný program rozdělen do tří fází: trénování, pracovní, detekce očí. Každá pak byla vysvětlena. Součástí práce je vytvoření trénovacího programu v jazyce C/C++ a pracovní část je implementována v prostředí MATLAB. Vzhledem k náročnosti celé problematiky, byl po domluvě se školitelem požadavek na detekci rtů vypuštěn. Úspěšné i neúspěšné detekce jsou uvedeny v kap. 3.6. K neúspěšným detekcím dochází zejména z důvodu příliš malé tváře, zakrytí tváře nebo naopak příliš členitého pozadí, které vypadá jako tvář. Většina takových detekcí je potlačena kombinací s detektorem barvy kůže.

# Použitá literatura

- [1] BRUCE, V., YOUNG, A. Understanding face recognition. In *The British Journal of Psychology*. 1986. 77. s. 305-327.
- [2] BRUNELLI, Roberto and POGGIO, Tomaso. Face recognition: features versus templates. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*. 1993, Vol. XV, 10.
- [3] COOTES, T. F., et al. Training Models of Shape from Sets of Examples. In *British Machine Vision Conference*. [s.l.] : [s.n.], 1992. s. 9-18.
- [4] COOTES, T.F. Active Shape Models - their training and applications. In *Computer Vision and Image Understanding*. [s.l.] : [s.n.], 1995
- [5] DUDA, Richard O., HART, Peter E. and STORK, David G. *Pattern Classification*. s.l. : Wiley-Interscience, 2000. 0471056693.
- [6] DUDA, Richard O., HART, Peter E., STORK, David G.. Unsupervised Learning and Clustering, Ch. 10 in *Pattern classification*. [s.l.] : Wiley. 2001. 571 s. ISBN 0-471-05669-3, 2001.
- [7] ETHEM, Alpaydin. *Introduction to Machine Learning*. [s.l.] : [s.n.], 2004. 445 s. ISBN 978-0-262-01211-9.
- [8] FAWCETT, T. An introduction to ROC analysis. *Pattern Recognition Letters*. 2006, 27.
- [9] FREUND, Yoav, SCHAPIRE, Robert R. A decision-theoretic generalization of on-line learning and an application to boosting. In *Journal of Computer and System Sciences*. [s.l.] : [s.n.], 1997. s. 119-139. Dostupný z WWW:

<<http://www.cs.princeton.edu/~schapire/uncompress-papers.cgi/FreundSc95.ps>>.

- [10] GALLANT, Stephen I. *Perceptron-based learning algorithms*. 2, 1990, IEEE Transactions on Neural Networks, Vol. I. 1045-9227.
- [11] GOMEZ G., MORALES E. *Automatic feature construction and a simple rule induction algorithm for skin detection*. 2002. Proc. of the ICML Workshop on Machine Learning in Computer Vision.
- [12] GONZALES, Rafael C., WOODS, Richard E. and EDDINS, Steven L. *Digital Image Processing Using MATLAB*. s.l. : Prentice Hall, 2003. 0130085197.
- [13] HAN, C.-C. Fast Face Detection via Morphology-Based Pre-Processing. In *Proc. Ninth Int'l Conf. Image Analysis and Processing*, [s.l.] : [s.n.], 1998. s. 469-476.
- [14] HSU, Rein-Lien, ABDEL-MOTTALEB, Mohamed and JAIN, Anil K. *Face Detection in Color Images*. 5, s.l. : IEEE, 2002, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. XXIV.
- [15] CHAIN, D. and NGAN, K. N. Face segmentation using skin-color map in videophone applications. *Circuits and Systems for Video Technology*. 1999, Vol. IX.
- [16] KOHONEN, Teuvo. *Self-Organization and Associative Memory*. [s.l.] : Springer, 1989. 312 s. ISBN 978-0387513874.
- [17] KOVAČ, Jure, PEER, Peter. and SOLINA, Franc. *Human skin color clustering for face detection*. s.l. : IEEE, 2003. EUROCON 2003. Computer as a Tool. The IEEE Region 8. Vol. 2.
- [18] LANITIS, A., TAYLOR, C. J., COOTES, T. F. An Automatic Face Identification System Using Flexible Appearance Models. In *Image and Vision Computing*. [s.l.] : [s.n.], 1995. Volume 13. s. 393-401.
- [19] MING-HSUAN, Yang, KRIEGMAN, David J., AHUJA, Narendra. Detecting faces in images: a survey. In *Pattern Analysis and Machine Intelligence*. [s.l.] : [s.n.], 2002. Volume 24. s. 34-58.
- [20] NELSON, Charles A. The development and neural bases of face recognition. In *Infant and Child Development : Volume 10*. [s.l.] : John Wiley & Sons, Ltd., 2001. s.

3-18. Dostupný z WWW:

<[http://www.biac.duke.edu/education/courses/spring03/cogdev/readings/C.A.%20Nelson%20\(2001\).pdf](http://www.biac.duke.edu/education/courses/spring03/cogdev/readings/C.A.%20Nelson%20(2001).pdf)>.

- [21] OSUNA, E., FREUND, R., GIROSI, F. Training Support Vector Machines: An Application to Face Detection. In *Computer Vision and Pattern Recognition*. [s.l.] : [s.n.], 1997. s. 130-136.
- [22] RAJAGOPALAN, A. Finding Faces in Photographs. In *Sixth IEEE Int'l Conf. Computer Vision*. [s.l.] : [s.n.], 1998. s. 640-645.
- [23] ROJAS, Raul. *Neural Networks - A Systematic Introduction*. Berlin : Springer-Verlag, 1996. 502 s. ISBN 978-3540605058.
- [24] ROSENBLATT, Frank. *The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain*. 6, místo neznámé : American Psychological Association, 1958, *Psychological Review*, Sv. LXV.
- [25] ROWLEY, H., BALUJA, S., KANADE, T. Human Face Detection in Visual Senes. In *Neural Information Processing Systems 8*. [s.l.] : [s.n.], 1996. s. 875-881.
- [26] ROWLEY, H., BALUJA, S., KANADE, T. Neural Network-Based Face Detection. In *IEEE Conf. Computer Vision and Pattern Recognition*. [s.l.] : [s.n.], 1996. s. 203-208.
- [27] ROWLEY, H., BALUJA, S., KANADE, T. Rotation Invariant Neural Network-Based Face Detection. *IEEE Conf. Computer Vision and Pattern Recognition*. [s.l.] : [s.n.], 1998. s. 38-44.
- [28] RUSSELL, Stuart, NORVIG, Peter. *Intelligence: A Modern Approach*. [s.l.] : Prentice Hall, 2002. 1132 s. ISBN 978-0137903955
- [29] SHUYAN, Zhao a GRIGAT, R. R. *Robust Eye Detection under Active Infrared Illumination*. místo neznámé : *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on, 2006*.
- [30] SCHAPIRE, Robert E. The boosting approach to machine learning: An overview.. In *MSRI Workshop on Nonlinear Estimation and Classification*. [s.l.] : [s.n.], 2002. s. 23. Dostupný z WWW: <[www.ccls.columbia.edu/compbio/geneclass/non\\_html\\_files/Schapire\\_boosting\\_review.pdf](http://www.ccls.columbia.edu/compbio/geneclass/non_html_files/Schapire_boosting_review.pdf)>.

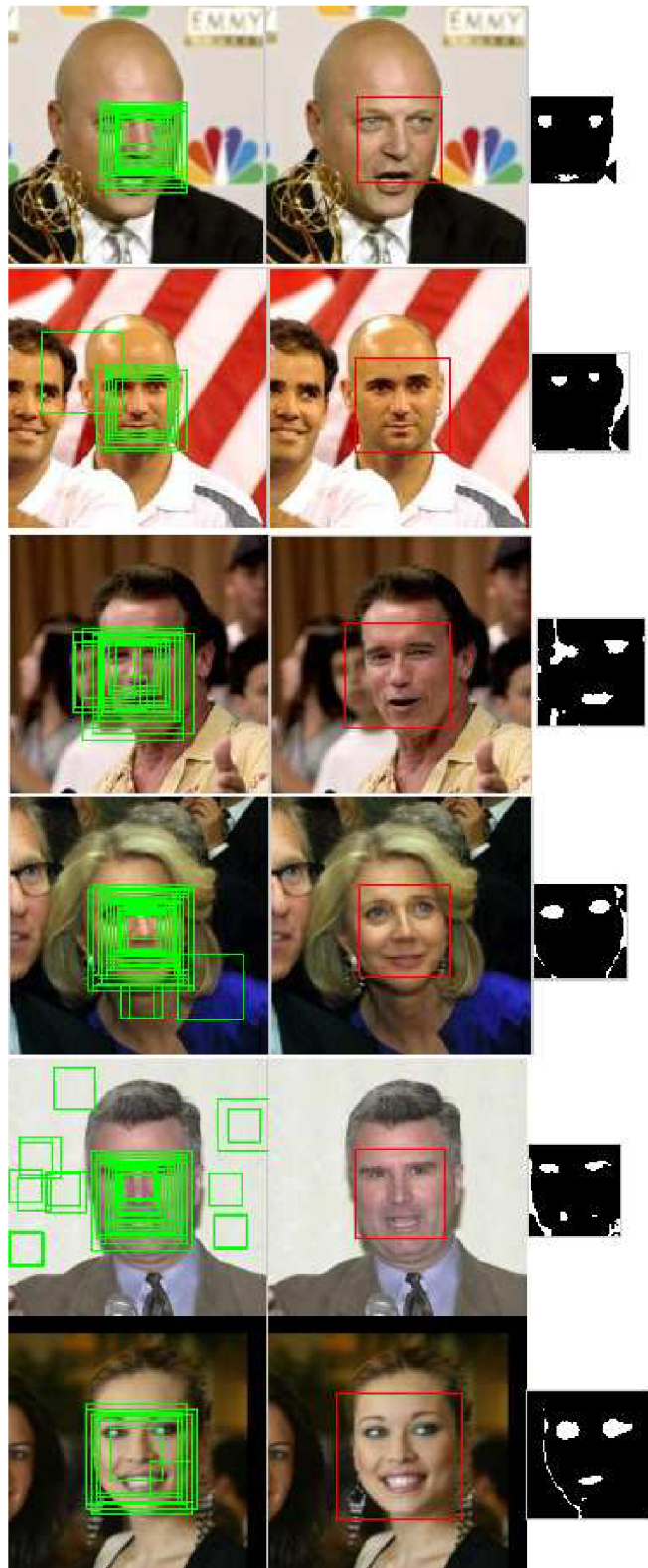
- [31] ŠOCHMAN, Jan. Adaboost. *Cvičení z RPZ* [online]. 2005 [cit. 2005-05-26].  
Dostupný z WWW:  
<[http://cmp.felk.cvut.cz/cmp/courses/recognition/Lab\\_archive/RPZ\\_04-05s/adaboost/adaboost.pdf](http://cmp.felk.cvut.cz/cmp/courses/recognition/Lab_archive/RPZ_04-05s/adaboost/adaboost.pdf)>.
- [32] VAPNIK, V. N., LERNER, A. Pattern Recognition Using Generalized Portrait Method. In *Automation and Remote Control*. [s.l.] : [s.n.], 1963. s. 774-780.
- [33] VIOLA, P., JONES, M. Robust Real-time Object Detection. July 13, 2001.  
Vancouver, Canada [cit. 2007-04-13]. Dostupný z WWW:  
<<http://cs223b.stanford.edu/notes/viola01robust.pdf>>.
- [34] YANG, G., HUANG, T. S. Human Face Detection in Complex Background. In *Pattern Recognition*. [s.l.] : [s.n.], 1994. Volume 27. s. 53-63.
- [35] YUILLE, Alan L., HALLINAN, Peter W. and COHEN, David S. *Feature extraction from faces using deformable templates*. 2, s.l. : Springer Netherlands, 1992, Vol. VIII. 0920-5691.



# Seznam příloh

A	Ukázky správných detekcí	59
B	Ukázky chybných detekcí	60

# A Ukázky správných detekcí



## B Ukázky chybných detekcí

