



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**  
BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA INFORMAČNÍCH TECHNOLOGIÍ**  
FACULTY OF INFORMATION TECHNOLOGY

**ÚSTAV INTELIGENTNÍCH SYSTÉMŮ**  
DEPARTMENT OF INTELLIGENT SYSTEMS

**AKCELERACE NEURONOVÉ SÍTĚ PRO DETEKCII OB-  
LIČEJE VE ZHORŠENÝCH SVĚTELNÝCH PODMÍNKÁCH**  
ACCELERATION OF A NEURAL NETWORK FOR FACE DETECTION IN LOW LIGHT CONDITI-  
ONS

**BAKALÁŘSKÁ PRÁCE**  
BACHELOR'S THESIS

**AUTOR PRÁCE**  
AUTHOR

**VOJTECH ORAVA**

**VEDOUCÍ PRÁCE**  
SUPERVISOR

**Ing. TOMÁŠ GOLDMANN**

**BRNO 2023**

## **Abstrakt**

Cílem této práce je vytvořit detektor obličejů v záznamech s důrazem na rychlosť a přesnosť s využitím technologie neuronových sítí a specializovaného hardwaru pro tyto sítě. Práce obsahuje sumarizaci dosavadních řešení detektorů a navrhuje řešení pro akceleraci detektorů. Toto řešení spočívá v [[TODO]]. Navržené řešení bylo implementováno a bylo dosaženo výsledků [[Výsledky experimentů]]. Na základě výsledků bylo docíleno [[zrychlení detekce o XX%]] Přínosem této práce je hlavně [[přínos]]

## **Abstract**

Do tohoto odstavce bude zapsán výtah (abstrakt) práce v anglickém jazyce.

## **Klíčová slova**

Detekce obličeje, akcelerace neuronových sítí, NCS 2, detekce v reálných podmírkách, neuronové sítě, Python, počítačové vidění

## **Keywords**

Face detection, Neural Networks acceleration, NCS 2, detection in real conditions, Neural Networks, Python, Compute Vision

## **Citace**

ORAFA, Vojtěch. *AKCELERACE NEURONOVÉ SÍTĚ PRO DETEKCI OBLIČEJE VE ZHORŠENÝCH SVĚTELNÝCH PODMÍNKÁCH*. Brno, 2023. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Tomáš Goldmann

# **AKCELERACE NEURONOVÉ SÍTĚ PRO DETEKCI OBLIČEJE VE ZHORŠENÝCH SVĚTELNÝCH PODMÍNKÁCH**

## **Prohlášení**

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana X... Další informace mi poskytli... Uvedl jsem všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpal.

.....  
Vojtěch Orava  
19. prosince 2022

## **Poděkování**

V této sekci je možno uvést poděkování vedoucímu práce a těm, kteří poskytli odbornou pomoc (externí zadavatel, konzultant apod.).

# Obsah

<b>1</b>	<b>Úvod</b>	<b>2</b>
<b>2</b>	<b>Kamery a systémy pro detekci obličejů</b>	<b>3</b>
2.1	Kamery . . . . .	3
2.2	Dostupná řešení . . . . .	3
<b>3</b>	<b>Detekce obličeje v reálných podmírkách</b>	<b>5</b>
3.1	Detekce obličeje . . . . .	6
3.2	Problémy a omezení . . . . .	7
3.3	Detektory obličejů . . . . .	9
3.4	Detektory zaměřující se na špatné světelené podmínky . . . . .	13
<b>4</b>	<b>Neuronové sítě pro detekci obličeje</b>	<b>14</b>
4.1	Neuronové sítě . . . . .	14
4.2	Datasetsy . . . . .	17
4.3	Detektory obličeje . . . . .	19
4.4	Akcelerace detekce . . . . .	22
<b>5</b>	<b>Implementace algoritmu pro akceleraci detekce obličeje</b>	<b>23</b>
5.1	Použité nástroje . . . . .	23
5.2	Trénování neuronové sítě . . . . .	23
5.3	Detekce obličejů natrénovanou sítí . . . . .	23
<b>6</b>	<b>Porovnání výkonnosti řešení s existujícími detektory</b>	<b>24</b>
6.1	Postup testování . . . . .	24
6.2	Porovnání výsledků . . . . .	24
6.3	Shrnutí . . . . .	24
<b>7</b>	<b>Závěr</b>	<b>25</b>
<b>A</b>	<b>Priloha</b>	<b>26</b>

# Kapitola 1

## Úvod

Neuronové sítě (anglicky neural networks) mají v dnešním světě mnoho využití. Jelikož se jedná o jednu z aplikací umělé inteligence (anglicky artificial intelligence), lze neuronové sítě použít například k rozpoznávání řeči, zpracování přirozeného jazyka či k detekci objektů. Tyto akce jsou pro běžného člověka poměrně snadné, avšak pro počítače znamenají relativně náročnou činnost.

Aby byly počítače schopné tyto akce vykonávat v rozumném čase (případně v reálném čase), je potřeba aby neuronové sítě byly dostatečně rychlé. Tato práce se zabývá akcelerací neuronových sítí v oblasti detekce obličeje. Zrychlení neuronové sítě lze dosáhnout buď optimalizací kódu, lepším trénováním neuronové sítě nebo také využitím speciálních hardwarových zařízení. Jedním z těchto specializovaných zařízení je Intel Neural Compute Stick 2, na něž se v této práci zaměřím.

Následující kapitola se obecně věnuje problematice detekce obličeje v reálných podmínkách s využitím neuronových sítí. Je zde detailně popsáno **[[CO JE DETAILNĚ POPSÁNO?]]**.

V kapitole ?? je nastíněn návrh akcelerace neuronové sítě pro detekci obličeje. **[[Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper. ]]**

Kapitola 5 se věnuje implementaci programu ke zrychlení detekce, také obsahuje informace o využitých prostředcích.

Předposlední kapitola (kapitola č.6) poskytuje přehled experimentů a testů provedených s implementovaným řešením a s řešeními již existujícími. Hlavním tématem v této části je porovnání výkonnosti a zobrazení dosažených výsledků.

## Kapitola 2

# Kamery a systémy pro detekci obličejů

Detekci obličejů s využitím počítačových programů lze provádět nad snímkem (fotografií, obrázkem), sadou snímků, videozáznamem nebo tzv. real-timově pomocí kamer a kamerových systémů. V této kapitole jsou popsány aktuálně využívané prostředky pro vytváření podkladů k detekci obličejů (kamery) a také dostupná řešení zabývající se detekcí, a to jak placené komerční, tak neplacené open-source systémy.

### 2.1 Kamery

Nezbytnou součástí oboru detekce obličejů jsou kamery a kamerové systémy. Existují kamery specializované k detekci či rozpoznávání obličejů a kamery obyčejné, které pouze zprostředkovávají obraz dále ke zpracování.

Specializované kamery se používají například k zabezpečení objektů nebo jako domovní videozvonky, kdy kamera (respektive její software) v zachyceném obrazu detekuje a rozpozná obličej, a následně může vykonat přiřazenou akci (spustit alarm, poslat notifikaci, umožnit osobě vstup...) [?].

Další oblastí, kde se kamery s detekcí a rozpoznáváním obličejů uplatní je bezesporu dohled ve veřejných prostorách (anglicky CCTV surveillance). Detekce obličeje může sloužit k hledání podezřelých osob v záznamech z dohledových kamer. Tyto záznamy jsou shromážďovány na serveru, pomocí detekce jsou z obrazu vyřezány fragmenty s obličeji lidí a poté jsou tyto fragmenty porovnány rozpoznávacím algoritmem s obličeji hledaných osob. Při shodě dochází k informování administrátora systému, který podnikne další kroky [?].

Kamery tedy mají v doméně detekce obličejů nezastupitelnou roli, na jejich praktické využití v systémech a řešení pro detekci se zaměřuje následujících podkapitol.

### 2.2 Dostupná řešení

Řešení umožňující detekci (a často i rekognici) obličejů lze rozdělit do dvou kategorií: komerční (placené, profesionální) a nekomerční (zdarma, open-source, amatérské). V následujících dvou podkapitolách jsou popsány konkrétní systémy poskytující detekci obličejů s jejich výhodami a nevýhodami.

## Komerční

Komerčně využívaná zařízení pro detekci, případně rekognici lze běžně zakoupit a používat ve firemním nebo domácím prostředí. Mezi zástupce těchto zařízení patří například produkty firem Netatmo, Google, Nest [?].



Obrázek 2.1: Nest Hello [?]

## Nekomerční

Nekomerční řešení pro detekci obličejů zahrnují frameworky s otevřeným zdrojovým kódem (open-source). Tyto frameworky využívají neuronové sítě, které jsou trénovány pomocí datasetů a následně je framework využit k detekci obličeje [?]. Frameworky TinaFace [?] a MTCNN [?] mají zdrojové kódy volně dostupné na webu GitHub.com.

## Kapitola 3

# Detekce obličeje v reálných podmírkách

Detekce obličeje (anglicky face detection) [?, ?] je technologie, která umožňuje v digitálním obrázku lokalizovat lidský obličej. Detekce obličeje patří do skupiny technologií HCI (Human–Computer interaction). Detektovat obličej je poměrně jednoduchý úkol pro lidi, ale zároveň se jedná o relativně náročný úkol pro počítače. Detekce obličeje je výchozím bodem pro další algoritmy analyzující lidský obličej, jako je například (v závorce za pojmem následuje anglický překlad):

- rozpoznávání obličeje (face recognition),
- zarovnání obličeje (face alignment),
- ověřování pomocí obličeje (face verification/authentication),
- sledování pohybu hlavy (head pose tracking),
- určování věku nebo pohlaví (age/gender recognition),

a mnoho dalších.

Samotná detekce obličeje se v realním prostředí využívá například v oblasti fotografování (automatické ostření na tvář), marketingu (zjišťování zájmu zákazníku o produkty podle počtu výskytu obličejů) nebo bezpečnosti (bezpečnostní kamery a systémy).

Následující podkapitoly se zabývají principem fungování detekce obličeje v reálných podmírkách a problémy a omezeními, které se v běžném světě vyskytují a detekce by si s nimi měla umět poradit (špatné světelné podmínky, příliš členité pozadí, přílišný počet obličejů v obrázku, barva kůže, nízké rozlišení atd.). Na konci této kapitoly se nachází popis algoritmů a detektorů, které ke svému fungování nepoužívají přímým způsobem neuronové sítě.



Obrázek 3.1: Příklad detekce obličeje

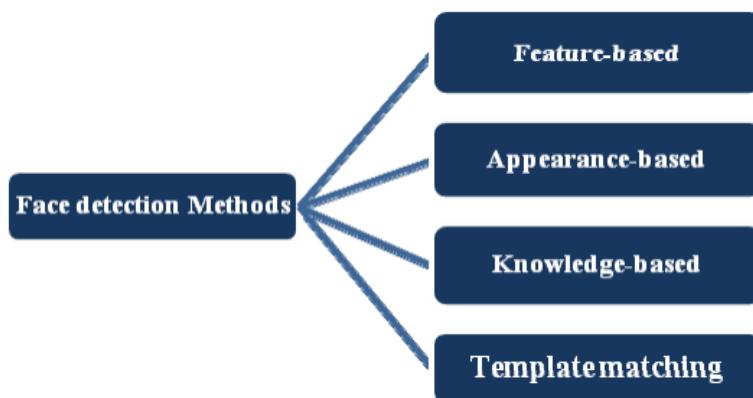
### 3.1 Detekce obličeje

Detekci obličeje lze rozdělit do několika přístupů, vědecké práce na toto téma se liší a nelze jasně říci zda to či ono dělení je jediné korektní.

Dle [?] existují 2 různé přístupy k hledání tváří v obrázcích, a to **přístup založený na vlastnostech** (anglicky feature based approach) a **přístup založený na obrázku** (anglicky image based approach). **Přístup založený na vlastnostech** nepoužívá přímo k detekci obličeje umělou inteligenci a neuronové sítě. Využívá vlastnosti obličeje jako takového (rysy, pozice očí, uší, obočí, barvu kůže...). Efektivita tohoto přístupu se snižuje s výskyty problémů popsaných v sekci 3.2, protože může docházet například k zakrytí nebo špatné viditelnosti některých vlastností obličeje.

Naproti tomu **obrazový přístup** uplatňuje schopnosti neuronových sítí a umělé inteligence k natrénování modelu neuronové sítě a následné přímé detekci pomocí tohoto modelu.

Podle [?] lze rozdělit metody detekce obličeje do 4 základních kategorií (viz obrázek 3.2) a 2 zvláštních kategorií (Haarovy vlastnosti a umělá inteligence).



Obrázek 3.2: Dělení metod detekce obličeje dle [?]

**Feature-based methods** (metody založené na vlastnostech) opět pracují s vlastnostmi obličeje. Vyhledávají v obraze rysy obličeje, což může výrazně ztížit či znemožnit neviditelnost některých rysů. Výhodou těchto metod je rychlosť v porovnání s ostatními metodami.

**Appearance-based methods** (metody založené na vzhledu/obrázku) využívají klasifikace vlastností tváře do 2 tříd v podobrázku celého obrázku. Klasifikátory podle nichž se daná metoda rozhoduje, zda se jedná o tvář či nikoli, mají různé váhy, metoda postupuje od slabých klasifikátorů k silnějším.

Metody založené na znalostech (**Knowledge-based methods**) se uplatňují při detekci obličeje v obrázku s čelnitým/komplexním pozadím. Znalosti které k detekci pomáhají jsou například ty, že obličeji má 2 uši, jeden nos, jedny ústa nebo vzdálenosti mezi jednotlivými rysy obličeje.

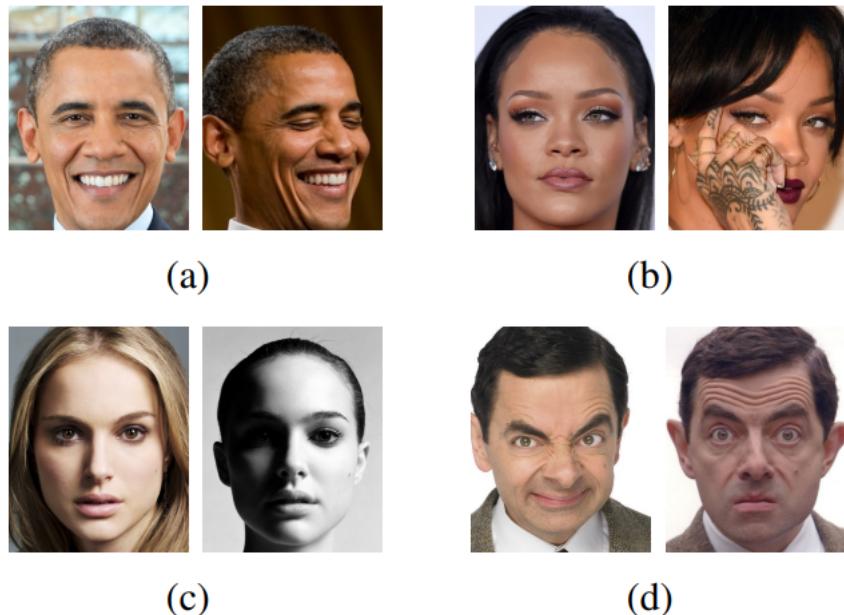
**Templatematching** (šablonování, maskování) aplikuje na obrázek předem danou masku obličeje a snaží se detektovat obličeji pomocí postupného maskování. Tato metoda je snadná na implementaci, jejím nedostatkem je však závislost na přímém pohledu tváře na obrázku.

Haarovy vlastnosti a budou popsány později. Detekce pracující na principu rozpoznávání vlastností v obraze jsou popsány v sekci 3.3, detektory s využitím neuronových sítí popisuje sekce ??.

## 3.2 Problémy a omezení

Algoritmy pro detekci obličeji čelí několika výzvám a omezením spojených s ne vždy perfektním zobrazením obličeje v obrázku. Lidské obličeje na fotografiích a obrázcích mohou být částečně zakryté (např. sluneční brýle), mohou být pořízené za nevhodných světelných podmínek (např. zastínění části tváře) nebo mohou nabývat nedostatečné kvality (nízké rozlišení).

Jelikož tedy vstupní obrázek detekce obličeje nemusí být vždy ideální, nemusí být obličeji vždy správně detektován. V této sekci jsou popsány některé problémy [?, ?], které mohou bránit v úspěšné detekci. Minimalizace dopadu těchto jevů na detekci je klíčem k navýšení uspěšnosti detekce. Mezi problémy a omezení (viz obrázek 3.3) pro detekci patří pozice hlavy, zakrytí části/částí obličeje, špatně osvětlená scéna nebo výraz tváře.



Obrázek 3.3: Vybrané problémy při detekci obličejů. Převzato z [?]. (a) Pozice hlavy; (b) Zakrytí části obličeje; (c) Špatné světelné podmínky; (d) Výraz tváře

### Pozice hlavy

Hlava může být na fotografii různě natočena, takže obličej nemusí být zachycen v přímém pohledu do kamery, ale může být zaznamenán z profilu nebo ze šikma (poloprofil) jako na obrázku 3.3 část (a).

### Zakrytí části obličeje

Výsledek detekce může být ovlivněn i zakrytím části obličeje (rukou, brýlemi, vlasy, šátkem apod.).

### Výraz tváře

Výraz lidského obličeje mohou ovlivnit emoce a nálady. Detektor zabývající se detekcí a rozpoznáváním emocí [?] dosahuje přesnosti 96 %.

### Orientace obrázku

Problémem pro detekci může být různá orientace obrázku (vzhůru nohama, zrcadlově otočený, natočený do strany apod.). Na obrázek je tak nutno aplikovat některé transformační operace pro zarovnání.

### Nedostatečně výkonná detekce

Velmi důležitým faktorem při detekci obličejů, zvláště v real-timových aplikacích, je rychlosť detekce. Pokud má algoritmu vysokou účinnost, ale je příliš pomalý pro vybranou aplikaci,

stává se nepoužitelným. Zrychlováním detekčních algoritmů se zabývá několik vědeckých studií.

### Příliš členité pozadí

Pokud se v o brázku nachází příliš mnoho objektů, může dojít ke snížení přesnosti a rychlosti detekce.

### Přílišný počet obličejů v obrázku

Výskyt velkého počtu obličejů v jednom obrázku, často překrývajících se, může představovat výzvu pro detekční algoritmus.

### Nízké rozlišení

Obrázky a fotografie s nízkým rozlišením nemusejí obsahovat dostatek informace nutné ke správnemu detekování tváře.

### Špatné světelné podmínky

Na detekci mohou mít vliv světelné podmínky panující při pořizování zkoumané fotografie či videa. Aspekty jež světlo ovlivňuje jsou mj. jas, kontrast, barvy, stíny, ostrost. Tato práce se zabývá detekcí obličejů v záznamech, v nichž některý z těchto faktorů omezuje detekci.



Obrázek 3.4: Příklad stejného obličeje vyfoceného při různých světelných podmínkách. Převzato z [?]

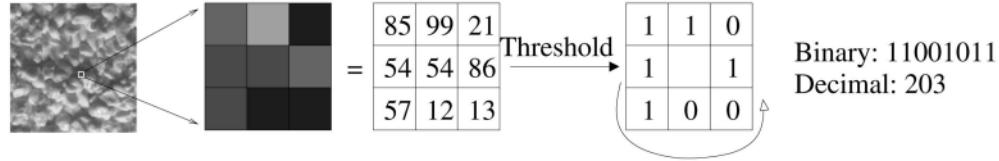
## 3.3 Detektory obličejů

Existuje několik různých přístupů k detekci obličeje. Tato sekce se zaměřuje na detekci s využitím detektorů nezaložených primárně na neuronových sítích (neuronová síť není použita vůbec, nebo není použita k přímé detekci). Detektory obličeje využívající principy neuronových sítí k přímé detekci budou popsány v další kapitole.

### Local Binary Patterns

Algoritmus využívající **lokální binární vzory** (anglicky Local Binary Patterns – dále jen LBP) pro popis struktury/textury obrázku má mnoho aplikací [?]. Jedná se o jeden z nejvýkonnějších algoritmů pro popis textur v obrázcích. LBP algoritmus má vysokou účinnost detekce (89 %) [?] a nízkou výpočetní náročnost. LBP pracuje s černobílými (anglicky gray-scale) obrázky a v originální verzi funguje tak, že každému pixelu přiřadí binární číselnou

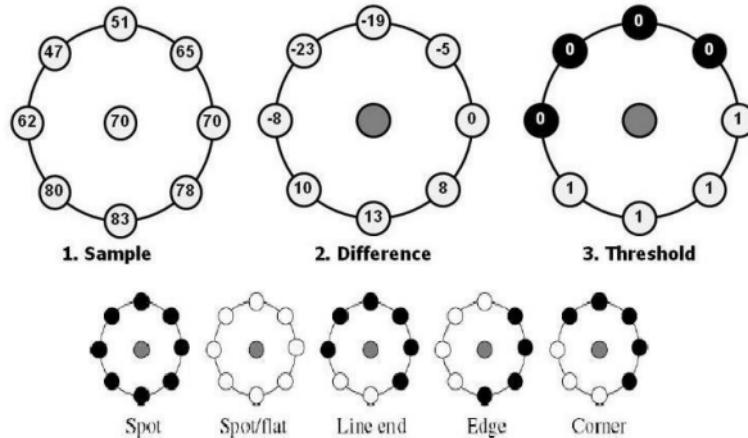
hodnotu, vypočítanou dle hodnot pixelů v  $3 \times 3$  okolí daného pixelu. Každý tento pixel v okolí je ohodnocen buď 1 nebo 0 v závislosti na tom, zda jeho hodnota překročila stanovený práh (anglicky threshold), kterým je hodnota prostředního pixelu (obrázek 3.5).



Obrázek 3.5: Ukázka ohodnocení pixelu algoritmem LBP. Převzato z [?]

Algoritmus byl později vylepšen tak, aby uměl pracovat s různě velkým okolím, které navíc nemusí mít čtvercový tvar. Používané okolí ve tvaru kružnice je popsáno dvojicí  $(P, R)$ , kde  $P$  je počet vzorkovacích bodů a  $R$  je poloměr kružnice. Další vylepšení LBP algoritmu se týkalo definování tzv. uniformních vzorů (anglicky uniform patterns). Tyto vzory jsou ty vzory v nichž se vyskytuje nejvýše 2 přechody z 1 na 0 a opačně. Příkladem uniformního vzoru na okolí  $(8, 2)$  je 11100000 (1 přechod), či 00111000 (2 přechody).

Lokální primitiva (obrázek 3.6) a histogramy vytvořené z takto získaných hodnot se využívají například k detekci obličejů.



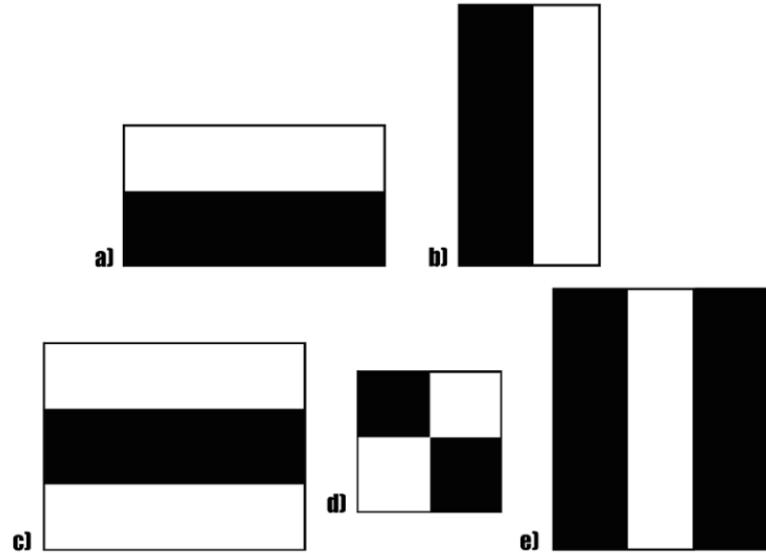
Obrázek 3.6: Lokální primitiva detekované algoritmem LBP. Převzato z [?]

## Viola–Jones algoritmus

Algoritmus **Viola–Jones** [?, ?], někdy také nazývaný **Haar Cascades**, je obecně technika pro detekci objektů, vytvořená před vynalezením metody hlubokého učení. Používá se k detekci obličejů, částí těla, očí, úst atd. Dosahuje přesnosti detekce kolem 90 % [?].

Princip fungování algoritmu spočívá v detekci hran a čár (obecně vlastností) v černobílém obrázku (hodnoty pixelů jsou na intervalu  $< 0; 1 >$ ). Vybere se jedna z vlastností (některé z nich zobrazuje obrázek 3.7) a vypočítá se průměrná hodnota pixelů ve všech obdélnících (obdélníky jsou dva, tři nebo čtyři). Rozdíl těchto hodnot pak určuje zda se

jedná o hranu, čáru tzn. zda je daná vlastnost detekována. Pokud například budeme hledat vlastnost b) z obrázku 3.7 a vypočtený rozdíl hodnot je blízký 1, detekce byla úspěšná (viz ukázka v obrázku 3.8).



Obrázek 3.7: Haarovy vlastnosti. Převzato z [?] a) horizontální hrana; b) horizontální hrana; c) horizontální čára d) diagonála e) vertikální čára

0.1	0.2	0.2	0.4	0.2	0.4
0.1	0.8	0.7	0	0.1	1
0.3	1	0.6	0.2	0	0.4
0.5	0.7	0.9	0.1	0	0.5
0.7	1	1	0.1	0.3	0.6
0.2	0.3	0.7	0.4	0.5	1

$$\text{Suma pixelů v černém obdélníku} = \\ 0.8 + 0.7 + 1 + 0.6 + 0.7 + 0.9 + 1 + 1 \\ = 6.7$$

$$\text{Suma pixelů v červeném obdélníku} = \\ 0 + 0.1 + 0.2 + 0 + 0.1 + 0 + 0.1 + 0.3 \\ = 0.8$$

$$\text{Průměrná hodnota v černém obdélníku} = \\ 6.7 / 8 = 0.8375$$

$$\text{Průměrná hodnota v červeném obdélníku} = \\ 0.8 / 8 = 0.1$$

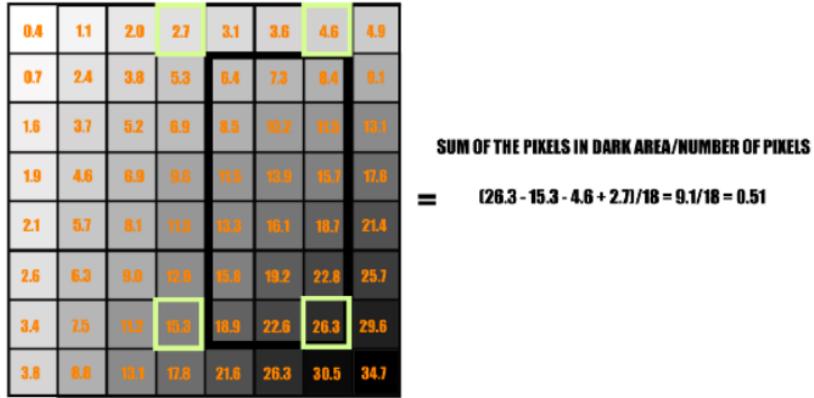
$$0.8375 - 0.1 = 0.7375$$

DETEKOVÁNA HRANA

Obrázek 3.8: Příklad výpočtu detekce hrany dle vlastnosti b) z obrázku 3.7. Bílý obdélník je nahrazen červeným pro lepsí viditelnost.

Algoritmus postupně prochází celý obrázek a hledá výskyt některé z vlastností. Toto procházení u obrázků s velkým počtem znamená enormní výpočetní nároky, protože je potřeba vždy počítat s hodnotami všech dotčených pixelů. Proto Viola a Jones [?] navrli vylepšení nazvané anglicky **Integral Image**. To spočívá v tom, že všechny pixely stačí projít

pouze 1x, a každý tento pixel lze ohodnotit sumou hodnot pixelů směrem nalevo a nahoru (důsledkem je, že pixel s nejnižším ohodnocením se nachází v levém horním rohu a pixel s nejvyšším ohodnocením v pravém dolním rohu). Následný výpočet průměrné hodnoty ukazuje obrázek 3.9.



Obrázek 3.9: Ukázka výpočtu hodnoty obdélníku v Haarově vlastnosti za pomocí vylepšení **Integral Image**. Převzato z [?].

Jelikož Haarových vlastností může být velké množství, bylo vybráno 6000 nejvíce vyhovujících, které se k detekci obličejů používají. Detekce je rozdělena na několik etap, v každé etapě je vyhledáván v části obrázku výskyt několika vlastností (počet s každou etapou roste), pokud se vyhledání nezdaří, není již dále daná část obrázku prohledávána.

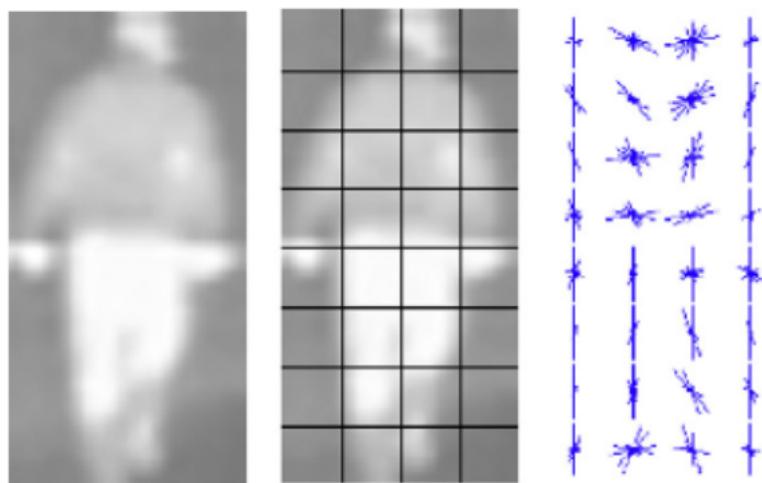
### Histogram orientací gradientů

Metoda HOG (anglicky Histograms of Oriented Gradients) [?, ?] používá k detekci obličejů či postav histogramy orientovaných přechodů v obrázku, rozděleném dle mřížky na několik bloků (často například  $8 \times 8$  nebo  $4 \times 4$  pixelů). Pro každý pixel v takovémto bloku jsou vypočítány hodnoty gradientů – velikost (magnitude) a směr (direction). Tyto hodnoty jsou pak přiřazeny do jednoho či více sloupců v histogramu popisujícím celý blok.

Tento histogram bývá rozdělen na 9 sloupců (rozmezí – anglicky bin) vyjadřujících směr gradientu v úhlu na škále od  $0^\circ$  do  $180^\circ$ , kdy každý sloupec odpovídá intervalu  $20^\circ$ . Výslednému vektoru, který udává velikosti gradientů v jednotlivých úhlech se říká HOG deskriptor.

HOG využívá pro zlepšení detekce při zhoršených světelných podmínkách normalizaci HOG deskriptorů. Tato normalizace je prováděna na vektorech 4 sousedících bloků. Pokud je tedy vektor složen z 9 hodnot, vypočítá se normalizační vektor z  $9 \times 4 = 36$  hodnot.

Výsledkem metody je obrázek (viz obrázek 3.10 reprezentovaný orientovanými gradienty, které umožňují detektovat osoby nebo obličeje).



Obrázek 3.10: Zleva výřez originálního obrázku převedený do šedotónové reprezentace, rozdělený obrázek na bloky, výstup metody HOG. Převzato z [?].

### 3.4 Detektory zaměřující se na špatně světelené podmínky

Tématem této práce je detekce obličeje ve špatných světelných podmínkách. Detekcí za těchto okolností se zabývala řada prací a bylo vytvořeno několik detektorů. V této sekci jsou popsány metody a algoritmy zaobírající se touto problematikou, včetně jejich úspěšnosti detekce při stanovených podmínkách.

## Kapitola 4

# Neuronové sítě pro detekci obličeje

Tato kapitola popisuje neuronové sítě a jejich využití pro detekci obličejů. Sekce 4.1 se věnuje popisu neuronových sítí obecně, v sekci 4.2 jsou popsány datasety a jejich využití k trénování neuronových sítí. Sekce 4.3 se zabývá konkrétními detektory obličejů s využitím neuronových sítí a v sekci 4.4 jsou zmíněny možnosti akcelerace detekčních algoritmů.

### 4.1 Neuronové sítě

Neuronové sítě [?, ?] umožňují nalezení neznámého řešení problému pomocí naučení se z podobných problémů u nichž známe řešení. Tyto umělé sítě jsou inspirovány biologickou nervovou soustavou lidí (lidským mozkem). Síla neuronových sítí se projevuje v úlohách zaměřených na detekci, rozpoznávání (objektů, lidí, obličejů, obecně vzorů – anglicky patterns) a zpracování dat. Existuje řada druhů neuronových sítí (konvoluční, hluboké, dopředné, rekurentní), pro detekci obličejů v obrázcích se nejčastěji používají konvoluční neuronové sítě (anglicky Convolutional Neural Networks – CNN).

Na princip fungování neuronových sítí může být nahlíženo jako na nelineární matematickou funkci, která převádí  $X$  vstupů na  $Y$  výstupů. Proces transformace vstupních informací na výstup je ovlivňován váhováním hodnot vně sítě. Tyto hodnoty vah jsou určovány při tzv. učení/trénování neuronových sítí. Pro správné natrénování neuronové sítě je potřeba dostatečného počtu vstupních trénovacích dat (obrázků, textů, hodnot) a dostatečně výkonný hardware.

#### Biologický neuron

Jak již bylo zmíněno, inspirací neuronových sítí je biologická nervová soustava. V lidském mozku se nachází okolo  $10^{11}$  neuronů (elektricky aktivních buněk spracovávajících signály). Vstupem těchto neuronů jsou tzv. dendrity, výstupy pak nazýváme axony. Jednotlivé neurony jsou navzájem propojeny tisíci spoji pojmenovanými synapse. Synapse zajišťují komunikaci mezi neurony.

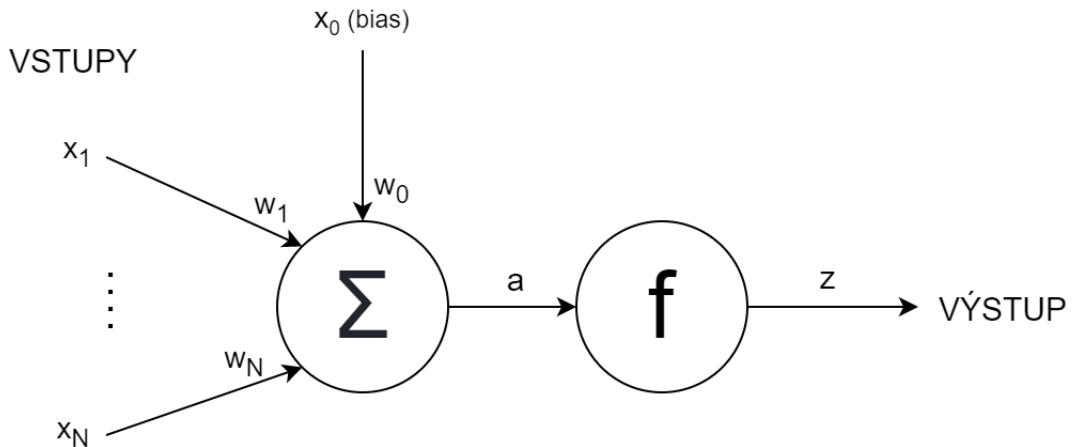
Takto vytvořený paralelismus poskytuje mozku schopnost rychle zpracovávat informace. Neurony biologické i umělé pracují s váhovanými vstupy. Po překročení určitého prahu na vstupech je adekvátně upraven výstup neuronu. Klíčovou vlastností potom je způsobilost měnit hodnoty jednotlivých vah na základě externích vlivů. Tím dochází k učení sítě neuronů.

## Perceptron

Nejjednoduším modelem umělého neuronu je perceptron. Perceptron má  $N$  vstupů, jejichž hodnoty  $x$  jsou vynásobeny váhmi  $w$  a sesumovány dle vzorce 4.1.

$$a = \sum_{i=1}^N (w_i * x_i) + w_0 * x_0 \quad (4.1)$$

Hodnota  $w_0$  se nazývá bias a jedná se o neměnnou vstupní hodnotu (často má hodnotu +1). Váhy a vstupy (včetně biasu) mohou být jak kladné, tak i záporné. Hodnota  $a$  je po vypočtení předána tzv. **aktivační funkci**.



Obrázek 4.1: Perceptron s  $N$  vstupy, biasem  $x_0$  a aktivační funkcí  $f$

## Aktivační funkce

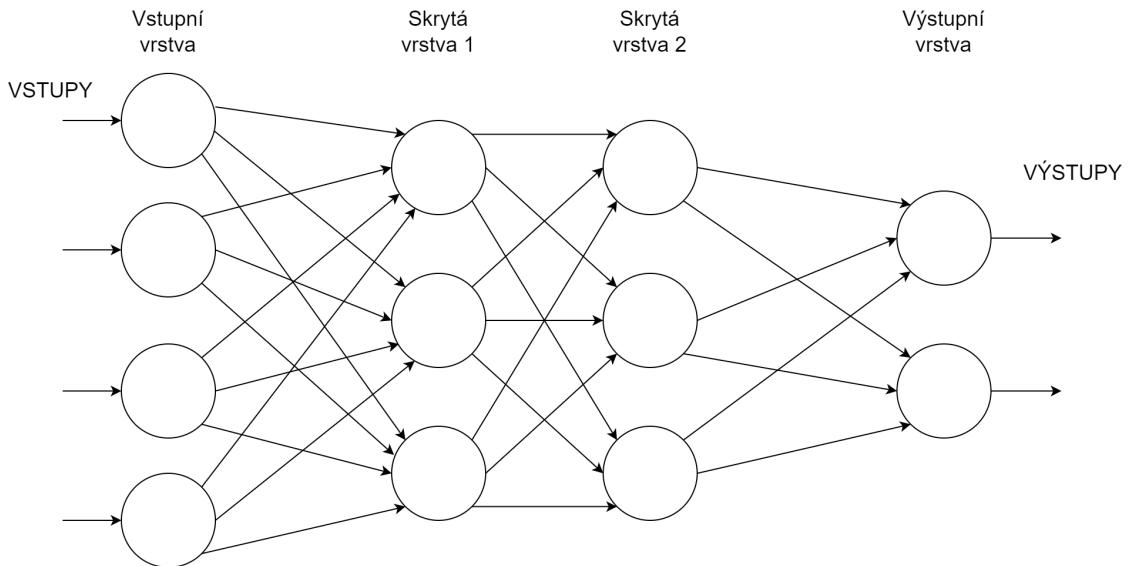
Aktivační funkce  $f$  je matematická funkce, která určuje výstup neuronu (vzorec 4.2). Na výběru vhodné aktivační funkce závisí přesnost neuronové sítě. Funkce může být různá, pro příklad je zde uveden výstup rozlišující zda se jedná o kladné či záporné číslo nebo nulu (viz vzorec 4.3). V praxi se často používá aktivační funkce nazvaná **rectified linear unit** (ReLU) [?] definovaná jako  $f(a) = \max\{0, a\}$ .

$$z = f(a) \quad (4.2)$$

$$f(a) = \begin{cases} 0 & \text{pro } a = 0 \\ 1 & \text{pro } a > 0 \\ -1 & \text{pro } a < 0 \end{cases} \quad (4.3)$$

## Spojování perceptronů

Spojením několika preceptronů lze vytvořit tzv. vrstvu (anglicky layer) perceptronů. Tyto vrstvy se dělí na vstupní (anglicky input), výstupní (anglicky output) a skryté (anglicky hidden). Konkatenací vstupní, několika skrytých a výstupní vrstvy je možno vytvořit neuronovou síť připravenou k trénování.



Obrázek 4.2: Propojení vrstev perceptronů do neuronové sítě

## Učení

Aby neuronová síť mohla fungovat, musí se natrénoval (tzn. nastavit co nejlépe váhy na vstupech perceptronů). K trénování se používají data z datasetů (viz sekce 4.2). Trénování neuronových sítí lze rozdělit do 3 kategorií [?]:

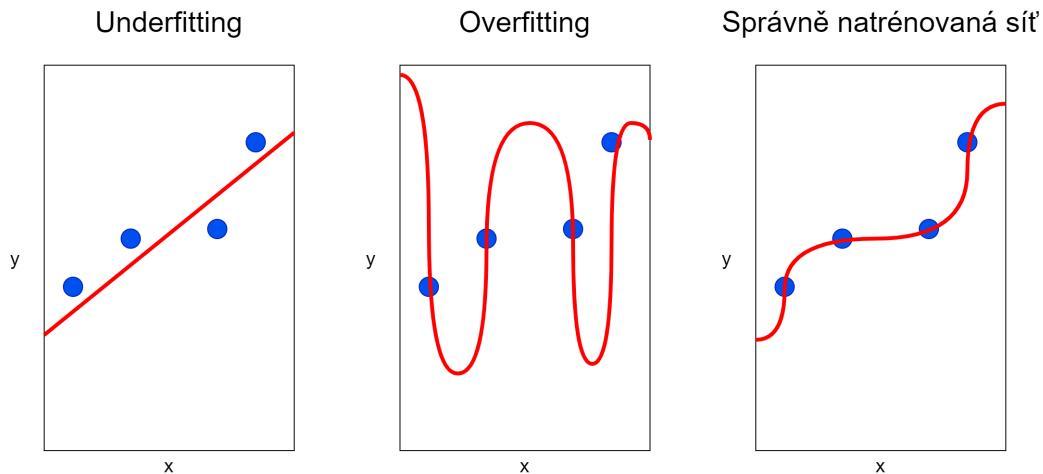
- **Učení bez učitele** (anglicky unsupervised learning) – tyto algoritmy procházejí data z datasetu a provádějí nad nimi shlukování do tzv. clusterů
- **Učení s učitelem** (anglicky supervised learning) – každým datům z datasetu je přiřazena informace o požadovaném výstupu. Vstupy jsou zpracovány neuronovou sítí a podle chyby (rozdílu vstup/výstup) jsou upraveny parametry v neuronové síti
- **Posilované učení** (anglicky reinforcement learning) – tento druh učení není vázán pouze na data z datasetu, ale navíc interaguje s prostředím

## Chyby

Abychom byli schopní aktualizovat váhy vstupů jednotlivých perceptronů, je potřeba měřit chybovost výstupu neuronové sítě. Jednou z možností měření chyby (využitelné například při lineární regresi) je *mean squared error*. Dle vztahu 4.4 je vypočítána chybovost a jsou upraveny váhy. Snahou je chybu co nejvíce zmenšit.

$$MSE = \frac{1}{N} \sum_i^N (vystupNS - spravnyVystup)^2 \quad (4.4)$$

Při trénování může dojít k situaci, kdy neuronová síť zvládá zpracovávat trénovací data s velmi vysokou přesností, ale při použití testovacích (validačních) dat se chybovost zvyšuje. V takovémto případě mluvíme o **přetrénování** (anglicky overfitting). Druhým nežádoucím jevem, který může nastat je **nedotrénovanost** (anglicky underfitting) – síť není dostatečně natrénovala a dochází tak k vysoké chybovosti (příčinou je například málo trénovacích dat).



Obrázek 4.3: Ukázka výstupu neuronové sítě při nedoučení, přeучení a dostatečně dobrém množství dat pro učení

## 4.2 Datasetsy

Datasetem rozumíme soubor podobných dat (například obrázků obličejů, číslic, předmětů nebo textů). Pro detekci obličejů se využívají datasety obsahující fotografie a videa lidí z veřejně dostupných zdrojů (internet, televize) nebo jsou datasety přímo účelně vytvářeny (fotografování lidí) a následně je možné si je koupit. Data v datasetech je nutné tzv. oanotovat (označit na daném obrázku je). V případě tváří se může jednat například o věk osoby, pohlaví, rasu, aby bylo možné určit zda po klasifikaci neuronovou sítí výstup odpovídá požadovanému výsledku.

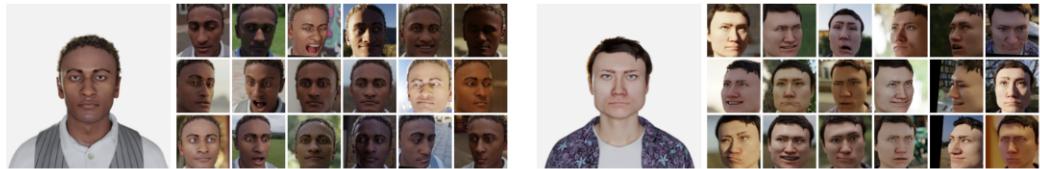
### Datasetsy lidských obličejů

Dataset **CelebFaces Attributes Dataset** (CelebA) [?] obsahuje více než 200000 obrázků známých osobností. Obrázky obsahují fotografie pořízené z různých úhlů a jsou velmi dobře anotovány. Tento dataset je volně přístupný pro nekomerční účely.



Obrázek 4.4: Příklady obličejů z datasetu CelebA

Dataset **DigiFace1M** [?] se skládá z 1 milionu snímků digitálně vytvořených obličejů (viz obrázek 4.5), címž se vyhýbá případným právním a etnickým problémům, které mohou být spojeny s využití tváří fyzických osob. Při použití tohoto datasetu umělých tváří společně s 200 až 2000 fotografiemi tváří reálných lidí lze dosáhnout podobných výsledků jako s datasety tvořenými čistě reálnými obličeji.



Obrázek 4.5: Příklady vygenerovaných obličejů pod různými úhly a různým osvětlením z datasetu DigiFace1M [?]

**Multi-PIE** [?] je dataset zaměřený na fotografie obličejů pořízených za různých světelních podmínek a pod různými úhly. Obsahuje přes 750000 snímků, které byly vytvořeny vyfotografováním 337 lidí po dobu několika měsíců. Tento dataset je placený.

Dataset **UTKFace** [?] obsahuje přes 20000 fotek obličejů lidí různých věků, pohlaví a ras z různých úhlů a za rozličných světelných podmínek. Dataset je volně dostupný pro nekomerční použití. Data jsou podrobně anotovány dle vzorce `[věk]_[pohlaví]_[rasa]_[datum a čas].jpg`, kde věk je v rozmezí 0–116 let, pohlaví muž/žena, rasa je jedna z výčtu běloch, černoch, asiat, ind, ostatní a datum a čas uchovává informaci o okamžiku zařazení fotografie do datasetu.

### 4.3 Detektory obličeje

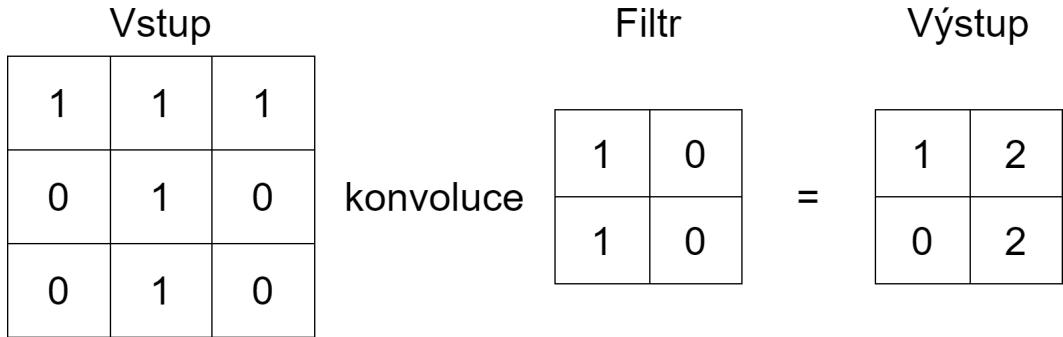
Detekce obličeje je zároveň samostatnou disciplínou i vstupním bodem několika dalších analýzačních úkonů (rozpoznávání tváře, ověřování pomocí obličejů nebo identifikace obličejů). Zpracovávaný obrázek bývá rozdělen na několik menších oblastí (výřezů) a tyto výřezy bývají zpracovávány jednotlivě. Existuje řada různých druhů a architektur neuronových sítí využívaných k detekci obličejů [?]. Patří mezi ně například:

- **Rotation invariant neural network** (RINN, rotačně invariantní neuronová síť) – dokáže detektovat obličeje bez ohledu na úhel natočení tváře (není nutná normalizace v preprocessingu). Systém tohoto typu neuronové sítě obsahuje několik sítí, s tím že první se nazývá *router network*. Tato síť určuje orientaci (natočení) výřezu obrázku a připravuje (normalizuje) tak výřez pro 1 či více detekčních sítí.
- **Fast neural network** (FNN, rychlá neuronová síť) – obrázek je rozdělen na malé podobrázky a každý podobrázek je zvlášť otestován na výskyt tváře rychlou neuronovou sítí. Cílem je snížení výpočetního času nutného pro nalezení obličeje.
- **Polynomial neural network** (PNN, polynomiální neuronová síť) – oblasti v obrázku jsou klasifikovány jako tvář/ne-tvář pomocí binomické projekce této oblasti do prostoru vlastností obličeje naučeného tzv. základní analýzou komponent (anglicky principal component analysis, PCA). Zkoumáním vlivu PCA na vzorky lze detektovat zda se jedná o obličej.
- **Convolutional neural network** (CNN, konvoluční neuronové sítě) – samostatně spomínány v další podsekci.

### Konvoluční neuronové sítě

Konvoluční neuronové sítě (CNN) [?, ?, ?] jsou použity ve většině systémů v oblasti počítačového vidění. Jelikož můžeme hodnoty pixelů vyjádřit čísla (používá se šedotónová varianta obrázku → při 8 bitové barevné hloubce může pixel nabývat hodnot 0 – 255), lze tyto čísla použít jako vstupy neuronové sítě. Při vysokém rozlišení obrázku však nastává problém. Pokud bychom chtěli každou hodnotu pixelu použít na vstupu neuronové sítě pro celý obrázek naráz, měla by neuronová síť již u obrázku s rozměry  $500 \times 500$  pixelů celkem 25000 vstupních neuronů, což je příliš výpočetně náročné. Proto se využívá konvoluce.

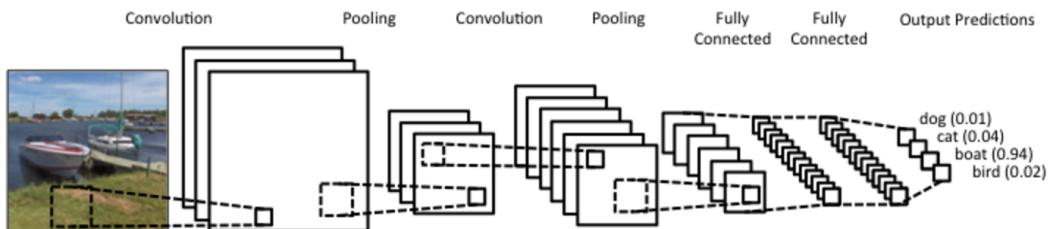
**Konvoluce** je okenní matematická funkce, při níž je na každou část obrázku (okno, window) násobením aplikována matici (filtr, kernel, konvoluční jádro). Princip konvoluce zachycuje následující obrázek.



Obrázek 4.6: Princip konvoluce 2D obrázku s rozměry  $3 \times 3$  a filtru  $2 \times 2$

Konvoluce dovoluje zásadně snížit počet nutných vstupů neuronové sítě. Navíc jsou konvoluční operace rychlé, protože tato funkce je hardwarově implementována na grafické kartě.

Jak lze vidět na obrázku 4.7 konvoluční neuronová síť se skládá z několika vrstev.



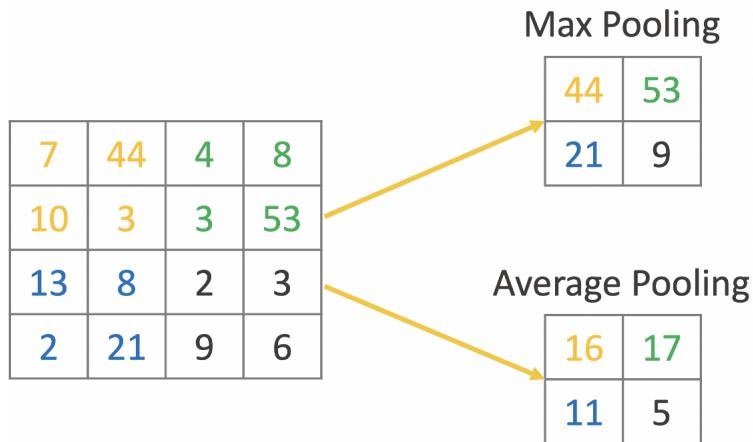
Obrázek 4.7: Řetězec vrstev konvoluční neuronové sítě [?]

**Konvoluční vrstva** provádí výše popsanou konvoluci. Otázkou zůstává jak zpracovávat okraje obrázku. Existují dva přístupy:

- *Wide convolution* – přidání nul kolem dokola obrázku a provedení konvoluce krajních hodnot (výstup bude stejně velký jako vstup).
- *Narrow convolution* – provedení konvoluce jako na obrázku 4.6. Dojde ke zmenšení velikosti na výstupu.

Dalším parametrem konvolučních neuronových sítí je tzv. **stride** velikost. Určuje o kolik hodnot se konvoluční jádro posune při každém kroku. Na obrázku 4.6 je použit stride 1. Větší hodnota vede k menšímu výstupu. Běžně se používají hodnoty 1 nebo 2.

**Poolingová vrstva** (anglicky pooling layer) je vrstva aplikovaná po konvoluční vrstvě. Používá se ke snížení výpočetní náročnosti. Tím, že je aplikována statistická funkce (maximum, průměr) na výstup konvoluční vrstvy (většinou filtrem o velikosti  $2 \times 2$  se stride = 2), dojde ke zredukování počtu hodnot výsledné matice (viz obrázek 4.8).



Obrázek 4.8: Řetězec vrstev konvoluční neuronové sítě [?]

V reálné aplikaci jsou konvoluční a poolingové vrstvy poskládány za sebe. Na konci řetězce (obrázek 4.7) se pak nachází část plně propojených vrstev neuronů (fully-connected layer).

## Detektory

Tato část popisuje vybrané existující detektory obličejů založené na neuronových sítích a prověření úspěšnosti detekce těchto detektorů s detektory popsánými v sekci 3.3.

**MTCNN** (Multi-Task Cascaded Convolutional Network) [?] provádí detekci obličejů pomocí kaskády konvoluční neuronových sítí. Algoritmus má dvě části. První část vytváří několik verzí vstupních obrázků, každa taková verze má jiné rozlišení. Druhá část složená z kaskády neuronových sítí se stará o detekci obličeje v různých verzích obrázku. Použití více verzí obrázku umožňuje zvýšit efektivitu detekce.

Single shot detector **SSD** [?] je konvoluční neuronová síť opět složená ze 2 částí, sloužící obecně k detekci objektů. Tyto části jsou: a) natrénovaný detekční model (VGG-16 nebo jiný) a b) část SSD – další konvoluční vrstva, která rozděluje obrázek pomocí mřížky (*grid*) s rozměry  $4 \times 4$  nebo  $8 \times 8$ . V každém takto vytvořeném bloku je pak hledán objekt (obličej).

Již výše zmíněný detekční/rozpoznávací algoritmus/neuronová síť **VGG-16** [?] používá k detekci konvoluční filtry o rozměrech  $3 \times 3$  v několika konvolučních vrstvách za sebou (počet filtrů se s každou další vrstvou navýšuje). Používá se zde stride o hodnotě 1.

**YOLO** (You Only Look Once) [?] je stejně jako SSD tzv. single shot detektor – rozděluje vstupní obrázek do mřížky a provádí detekci případně rekognici nad jednotlivými bloky mřížky. Pro každý blok je určena hodnota jistoty výskytu objektu. Na vstupu sítě YOLO jsou barevné obrázky o velikosti  $448 \times 448$  pixelů. Architektura neuronové sítě je složena ze 7 konvolučních vrstev následovaných pooling vrstvami a 3 plně propojenými vrstvami na výstupu. YOLO má několik verzí, které se liší složením neuronové sítě a úspěšností detekce.

Detektory zaměřující se na špatně světelené podmínky

#### 4.4 Akcelerace detekce

Intel Neural Compute Stick 2

## Kapitola 5

# Implementace algoritmu pro akceleraci detekce obličeje

5.1 Použité nástroje

5.2 Trénování neuronové sítě

5.3 Detekce obličejů natrénovanou sítí

## Kapitola 6

# Porovnání výkonnosti řešení s existujícími detektory

6.1 Postup testování

6.2 Porovnání výsledků

6.3 Shrnutí

## Kapitola 7

### Závěr

**Příloha A**

**Priloha**