

Detekce obličeje v obraze s využitím prostředí MATLAB[®]

T. Malach, P. Bambuch, J. Malach

EBIS, spol. s r.o.

Příspěvek se zabývá detekcí obličeje ve statických obrazových datech. Algoritmus detekce a trénování byl vytvořen v programovém prostředí MATLAB s podporou knihovny Image Processing Toolbox. V příspěvku je stručně popsán algoritmus objektového detektoru Viola-Jones. Cílem detekčního algoritmu je rozhodnout, zda se ve vstupním obraze nachází hledaný objekt. Obecný objektový detektor byl modifikován - přetrénován na detektor obličejů. Trénování je proces, při němž jsou nalezeny informativní příznaky dané množiny, v tomto případě obličejů. Vytvořené a převzaté detektory obličeje byly testovány na shodné testovací množině obrazů. Testovací obrazová množina (databáze) byla získána z videosekvencí z reálného prostředí. V závěru příspěvku jsou prezentovány výsledky testů detektorů implementovaných v jazyce MATLAB, C/C++ s podporou knihovny pro počítačové vidění - OpenCV a komerční produkt VeriLook Surveillance od firmy Neurotechnology. Tato práce vznikla v rámci řešení projektu IVECS spolufinancovaným MPO ČR, Program TIP ev.č. FR-TI3/170. Projekt IVECS se zabývá vývojem inteligentních videomodulů pro systém kontroly vstupů do objektů kritické infrastruktury. Jedním z videomodulů je modul FACE, jehož úkolem je detekce obličeje (obličejů) v obraze a výpočet příznaků pro rozpoznávání - identifikaci tváře. Na základě analýzy metod pro detekci obličeje byl vybrán detektor Viola-Jones [1].

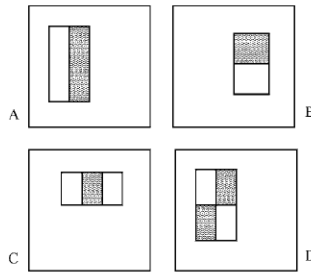
Detektor Viola-Jones

Detektor Viola-Jones patří do skupiny metod založených na vzhledu. Obecný detekční algoritmus je založený na výpočtu informativních příznaků, na základě kterých je rozhodnuto o existenci objektu v obraze. Klasifikátorem je jednouchý perceptron [2]. Detektor Viola-Jones používá jako příznaky odezvu na Haarovy filtry. Nalezení konkrétních Haarových filtrů, ze kterých jsou příznaky vypočteny, je proces nazývaný trénování. Trénování bylo realizováno pomocí algoritmu Adaboost.

Princip detekce obecného objektu spočívá ve výběru subobrazu ze vstupního obrazu, ze kterého jsou vypočteny příznaky a subobraz je klasifikován buď jako hledaný objekt nebo pozadí. Následně jsou vybírány další subobrazy, které jsou klasifikovány, až je prohledán celý vstupní obraz. Velikost vybraného subobrazu odpovídá velikosti objektu ve vstupním obraze. Chceme-li nalézt v obraze objekt o obecném rozměru, musíme několikrát změnit rozměr subobrazu - detektoru. Velikost vybraného subobrazu odpovídá velikosti objektu ve vstupním obraze.

Výpočet příznaků

Haarovy filtry jsou plošnými detektory charakteristických rysů daného objektu. Haarových filtrů může být mnoho druhů, základní sada Haarových filtrů je na obr.1. Existují i složitější filtry, které mohou pomoci detekovat objekt v obecnějších polohách a podmínkách. Odezva na Haarův filtr - příznak, je číslo, které je výsledkem rozdílu sum jasových hodnot pod světlou a tmavou částí Haarova filtru, viz Obr. 1. V subobrazu je vypočteno mnoho odezev - příznaků, na jejichž základě je rozhodnuto o přítomnosti objektu. Výpočet příznaků je v případě detektoru Viola-Jones velmi efektivní při zavedení tzv. integrálního obrazu [2].



Obr. 1 Základní sada Haarových filtrů

Integrální obraz je reprezentace vstupního obrazu, v níž jsou jasové hodnoty jednotlivých pixelů kumulativně sečteny v řádcích a sloupcích. Tato skutečnost je vyjádřena rovnicemi:

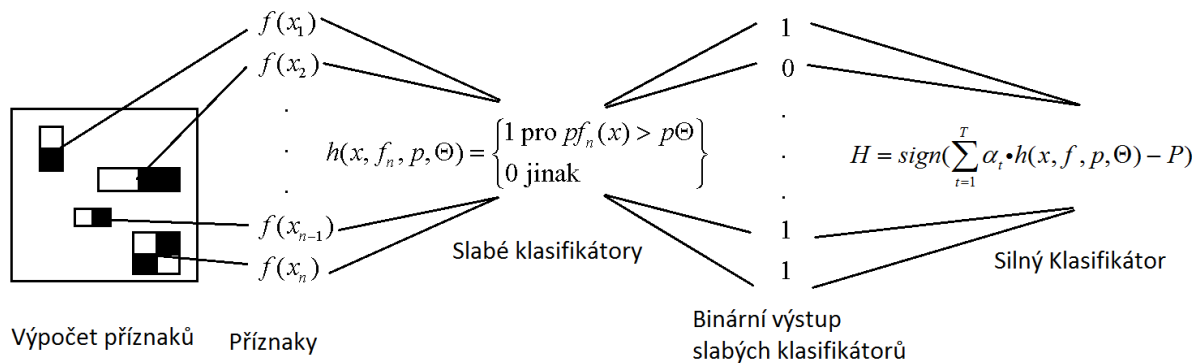
$$s(x, y) = s(x, y-1) + i(x, y),$$

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

kde $s(x,y)$ je kumulovaný součet jasových hodnot v řádku a $i(x,y)$ jsou hodnoty původního vstupního obrazu. Výsledný integrální obraz je reprezentován hodnotou $ii(x,y)$.

Klasifikace

Nechť je informativní příznak jakýkoliv příznak, pomocí něhož je možné rozlišit tvář od pozadí s úspěšností větší než 50%. Klasifikátor, který využívá takové informativní příznaky, je nazýván slabým klasifikátorem [2]. Pokud použijeme více slabých klasifikátorů, pak je možné ve výsledku dosáhnout vysoké klasifikační schopnosti. Slabé klasifikátory jsou spojeny v silném klasifikátoru (perceptronu). Vstupem silného klasifikátoru je váhovaný binární výstup slabých klasifikátorů. Tato skutečnost je vyobrazena na obrázku 2.



Výstup slabých klasifikátorů je vypočten dle vztahu:

$$h(x, f, p, \Theta) = 1 \text{ pro } f(x)p > p\Theta, \text{ jinak } 0,$$

kde $h(x,f,p,\Theta)$ je binární výstup slabého klasifikátoru, $f(x)$ je příznak, Θ je práh slabého klasifikátoru a p je parita. Slabé klasifikátory vstupují do silného klasifikátoru - perceptronu; váhy α_i jsou úměrné chybovosti jednotlivých slabých klasifikátorů při trénování. Můžeme tedy říci, že každý slabý klasifikátor má na celkovou klasifikaci vliv úměrný jeho chybovosti.

Silný klasifikátor potom má tvar:

$$H = \text{sign}\left(\sum_{t=1}^T \alpha_t \cdot h(x, f, p, \Theta) - P\right),$$

kde α_t jsou váhy slabých klasifikátorů, P je práh silného klasifikátoru vypočtený při trénování a T je celkový počet slabých klasifikátorů, ze kterých je složen silný klasifikátor.

Trénování

Trénovací algoritmy mají za úkol nalézt na základě statistického vyhodnocení data nutná pro klasifikaci. V případě detektoru Viola-Jones se jedná o konkrétní informativní příznaky (např. typ, pozice a rozměr Haarova filtru), prahové hodnoty slabých klasifikátorů Θ a váhy slabých klasifikátorů α .

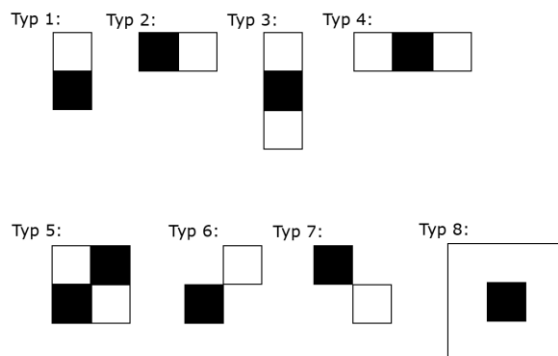
K trénování byl použit algoritmus Adaboost [2, 3, 4]. Pro natrénování je nutná trénovací databáze vstupních obrazů, která obsahuje tisíce obrazů obsahujících obličeje (pozitivní obrazy) a tisíce obrazů pozadí (negativní obrazy). Adaboost vybírá jednotlivé informativní příznaky podle toho, jakou mají chybovost při klasifikaci obrazů z trénovací databáze. V jednotlivých cyklech Adaboostu je provedeno nastavení důležitosti správné klasifikace jednotlivých obrazů - váh obrazů, výpočet všech možných příznaků z celé trénovací množiny, stanovení optimálního prahu Θ minimalizujícího chybovost slabého klasifikátoru používajícího daný příznak, výběr slabého klasifikátoru s nejmenší chybovostí a jeho zařazení mezi klasifikátory použité pro detekci. Jednotlivým obrazům jsou upravovány váhy podle toho, zda byly vybraným klasifikátorem správně klasifikovány. Tento cyklus se opakuje, dokud není dosažen žádaný počet slabých klasifikátorů.

Implementace detektoru a trénovacího algoritmu

Pro implementaci detektoru Viola-Jones i trénovacího algoritmu Adaboost bylo vybráno programovací prostředí MATLAB. Důvodů pro výběr bylo několik, první z nich byl široký rozsah funkcí pro zpracování obrazu a funkce pro realizaci shlukové analýzy, dále efektivní výpočet maticových operací a práce s maticemi.

Při implementaci byl zvolen postup, kdy jsou jednotlivé kroky programu řešeny sekvenčně, pro výpočty byly použity výše uvedené matematické vztahy. Sekvenční postup byl zvolen proto, že v jednotlivých krocích všech cyklů je možné kontrolovat výpočty a je stále zřejmá funkčnost algoritmu. Účelem implementace v MATLABu bylo odladění detekčních schopností algoritmu a nalezení parametrů, které zvýší robustnost detekce.

Obecný objektový detektor Viola-Jones byl natrénován jako detektor obličejů. Specifikace objektu, který bude detekován (v tomto případě obličeje) je určena trénovací množinou. První funkční implementace byla realizována s natrénovanými daty převzatými od spoluřešitele projektu, VUT v Brně [6]. Za účelem zvýšení detekčních schopností algoritmu, byl detektor přetrénován [4]. Pro přetrénování byla použita databáze, která byla použita při vývoji detektoru jeho autory Paulem Violou a Michaelem Jonesem.



Obr. 3 Sada Haarových filtrů, které byly použity při implementaci.

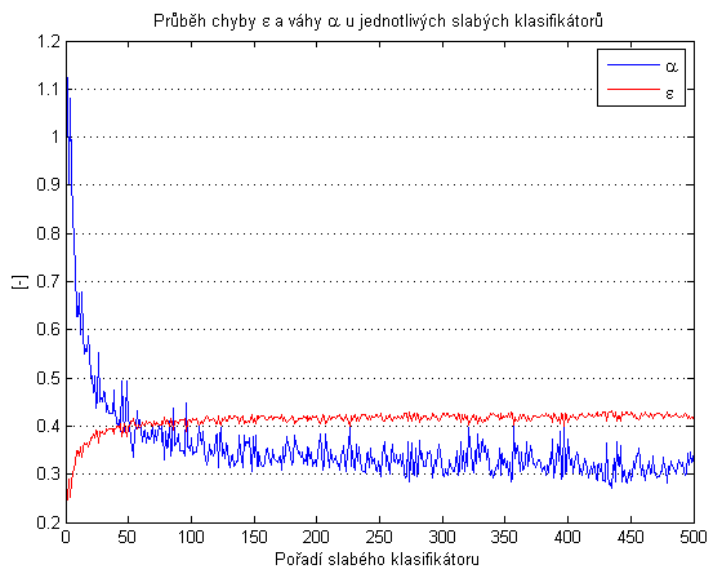
Trénovací množina (databáze) obsahuje 4 916 pozitivních obrazů a 7 872 negativních obrazů, celkem 12 788 obrazů. Následně bylo provedeno několik experimentů, kdy byla originální databáze upravena s cílem zlepšení detekčních schopností detektoru.



Obr. 4 Ukázka rozšíření trénovací databáze.

Algoritmus Adaboost má schopnost exponenciálně snižovat chybu výsledného klasifikátoru a dokáže produkovat v relativně krátkém trénovacím čase klasifikátory s velmi malou chybou i za použití jen velmi jednoduchých slabých klasifikátorů viz obr. 3 [4].

Silný klasifikátor byl sestaven z pětiset slabých klasifikátorů. Během jeho trénování byla zaznamenávána chybovost jednotlivých slabých klasifikátorů. Průběh chyby ε a váhy α je na obr. 5.



Obr. 5 Chyby a váhy jednotlivých slabých klasifikátorů při trénování.

Výsledky experimentů

Pro testování algoritmů byla vytvořena vlastní testovací množina, která obsahuje obrazy z definovaného prostředí. Vzhledem k zaměření projektu IVECS, není cílem vytvořit obecný detektor pracující v komplexním prostředí, ale detektor dobře pracující v omezené množině pozadí [4]. Výsledky testů z této definované testovací množiny jsou pro účely projektu více relevantní. Na základě výsledků byl detektor několikrát přetrénován na rozšířené databázi. Rozšíření databáze spočívalo v přidání obličejů, které nebyly detekovány,

a přidáním vzorů pozadí, které byly špatně detekovány (byly označeny jako obličeje). Souhrn výsledků testů uvedený v tabulce níže.

Detektor	Trénování	Trénovací databáze	Výsledky testování - kritériální ukazatelé [-]					
			TP	FP	TPR	FPR	PPV	ERR
IVECS	VUT	x	916	195	0,852	0,187	0,824	0,335
VK	VK	x	987	60	0,918	0,057	0,943	0,139
VK	IVECS	Viola-Jones	1005	102	0,935	0,098	0,908	0,163
VK	IVECS	Viola-Jones + rozšíření	918	0	0,854	0	1	0,146
OpenCV	OpenCV	FERET*	1038	151	0,965	0,14	0,873	0,18
VUT	OpenCV	FERET*	1031	125	0,959	0,12	0,892	0,16

IVECS - vlastní implementace

VK - implementace Vahida Kazemiho

VUT- převzato od UTKO VUT v Brně

TP - True Positives (detekce obličeje v místě, kde je obličej)

FP - False Positives (detekce obličeje v místě, kde není obličej)

FN - False Negatives (chybějící detekce obličeje v místě, kde je obličej)

TPR - True Positives Rate (TP/celkový počet obličejů v testovací databázi)

FPR - False Positives Rate (FP/celkový počet snímků v databázi)

PPV - Positive Predictive Value -TP/(TP+FP))

ERR - celková chyba $ERR=(1-TPR)+FPR$

* pravděpodobně

V rámci projektu IVECS byl vytvořen detektor, používající výstup trénování z VUT. Jeho celková chyba ERR byla na testovací množině 0,335. Opakovaným trénováním a přizpůsobením trénovací množiny prostředí, ve kterém bude detektor pracovat, byly vlastnosti detektoru podstatně zlepšeny [5]; bylo dosaženo podstatného snížení celkové chyby na 0,146. Výstup trénování byl použit jako vstup pro detektor Vahida Kazemiho. Detektor natrénovaný Vahidem Kazemim [7] dosáhl na testovací množině nejnižší celkové chyby 0,136. Algoritmus firmy Neurotechnology bylo možné testovat jen na živých videosekvencích z kamery, nebylo možné testování na statických obrazech. Obličeje detekoval vždy, nedocházelo k falešným detekcím, algoritmus je velmi spolehlivý a robustní. Detektor OpenCV dosáhl nízké celkové chyby ERR 0,18, je implementován v jazyce C a je možné jeho použití v aplikacích pracujících v reálném čase.

Reference

- [1] MALACH, Tobiáš. *Detekce obličeje v obraze*. Brno, 2011. 69 s. Bakalářská práce. VUT v Brně.
- [2] VIOLA, Paul; JONES, Michael. Robust Real-Time Face Detection. In *International Journal of Computer Vision*. The Netherlands: Kluwer Academic Publishers, 2003. s. 137-154.
- [3] MALACH, Tobiáš. *Implementace detektoru Viola-Jones*. Brno, 2011, výzkumná zpráva EBIS, spol. s r.o.
- [4] BAMBUCH, Petr. *Trénování Adaboost*. Brno, 2011, výzkumná zpráva EBIS, spol. s r.o.

- [5] BAMBUCH, Petr, MALACH, Tobiáš. *Testování algoritmů pro detekci obličeje v obraze*. Brno 2011, výzkumná zpráva EBIS, spol. s r.o.
- [6] PŘINOSIL, Jiří. *Analýza emocionálních stavů na základě obrazových předloh*. Brno, 2008. 90 s. Dizertační práce. Vysoké učení technické v Brně.
- [7] *School of computer science and communication* [online]. 2011 [cit. 2011-10-16]. Vahid Kazemi. Dostupné z WWW: <<http://www.csc.kth.se/~vahidk/>>.

Kontaktní informace

Tobiáš Malach
tmalach@ebis.cz

Petr Bambuch
pbambuch@ebis.cz

Jindřich Malach
jmalach@ebis.cz