

Rozpoznávání obličejů

Vlastimil Hopjan, David Pilař, Jiří Sovadina, Marek Vojtek

1.2.2004

1. Rozpoznávání obličeje

Identifikace osob

Biometrické metody identifikace osob nacházejí uplatnění i v jiných oblastech než je kriminalistické hledisko, například v oblasti bezpečnostních a přístupových systémů do objektů, kdy jsou různé biometrické metody kombinovány s čipovými přístupovými kartami. Zde lze zařadit například metody biometrie dlaně, daktyloskopické metody, analýzu hlasu, porovnávání duhovky a rohovky apod. V poslední době se tyto biometrické metody uplatňují i v zabezpečení přístupu do systémů personálních počítačů a přístupů do firemních informačních sítí (klávesnice se snímáním daktyloskopického otisku prstu).

Úvod do problému

Rozpoznávání obličeje vždy přitahovalo značnou pozornost. Ovšem velmi často byly požadavky na obličejový biometrický systém nevhodně formulovány. Někdy byly požadavky na systém tak přemrštěné, že bylo obtížné, ne-li téměř nemožné takový systém realizovat. Jde o to, že na jedné straně je porovnávání dvou statických obrazů a na straně druhé praktická realizace vycházející například z požadavku ověřit identitu jednotlivce nacházejícího se ve skupině lidí. Atraktivnost rozpoznávání obličejů je z hlediska uživatele pochopitelná, ovšem je nezbytné být realistický ohledně vyhlídek této technologie. Doposud měli obličejové rozpoznávací systémy v praktických aplikacích omezený úspěch. Již dnes však vychází najevo, že rozpoznávání obličejů se bude řadit mezi primární technologie pro zajištění systémů vysokých rizik.

Z hlediska sociologického je rozpoznávání známých tváří nejběžnější a nečastější podvědomou činností lidského mozku. Existuje množství studií a různých konceptů v procesu obličejového rozpoznávání. Velké úsilí je vyvíjeno ve směru technologie neuronových sítí. Neurofyzikální studie ukazují, že rozpoznávání a analýza je paralelní proces nejlépe odpovídající teorii neuronových sítí.

Realtimové rozpoznávání obličeje je problém, kterým se zabývají jak univerzity, tak i komerční subjekty, protože se jedná o oblast bezpečnosti a identifikace,

kteřá se již v dnešní době začíná prosazovat do popředí zájmů organizací, ale i států.

Využití je zřejmé: identifikace nežádoucích osob v hlídaných, sledovaných objektech (budovách), na letištích, identifikace osob před vstupem na pracoviště, nebo např. automaický strih, apod.

V našem případě se zaměříme na možnost rozpoznávání obličeje za účelem identifikace jeho "majitele".

Verifikace obličeje

Tato biometrická metoda je centrem mnoha výzkumů. Problematika identifikace osob využívající rozpoznávání tváří je velmi obsáhlá. Rozpoznávání je založeno na srovnávání obrazu sejmутého kamerou s obrazem, který je uložen v paměti počítače. K identifikaci slouží většinou tvar obličeje a poloha opticky významných míst na tváři (oči, nos, ústa, obočí).

Automatizované systémy identifikace osob mohou být tedy řešeny dvěma základními přístupy:

- PRVNÍ PŘÍSTUP - STRUKTURÁLNÍ - rozpoznávání jednotlivých dominantních částí obličeje (oči, ústa, nos...) předkládaného vzoru, změření antropometrických veličin, jejich normalizace vzhledem k předpokládaným rušivým vlivům (šum, rušení, poloha ve scéně, velikost...), porovnání s databází známých fotografií použitím klasifikačních algoritmů, statistické rozhodnutí o relativní podobnosti s takto vybranou množinou obrazů.

Obraz v počítači je někdy uložen jako matice jasových úrovní, častěji je však diskriminován nějakou funkcí, která snižuje redundanci dat. Neuchovává se např. přesná poloha očí, nosu a rtů, ale ukládá se jen vzdálenost očí, vzdálenost rtů od nosu, úhel mezi špičkou nosu a jedním okem, atd. V současné době je známo mnoho technik rozpoznávání tváří.

- DRUHÝ PŘÍSTUP - HOLISTICKÝ - POROVNÁNÍ - identifikace vzorku pomocí globálních reprezentací opět s následným statistickým vyhodnocením relativní pravděpodobnosti. Příznačné pro tento přístup jsou kombinace metody backpropagation (metoda zpětného učení neuronové sítě), základní analýzy komponent (principal component analysis - PCA) a dekompozice jedinečných hodnot (singular value decomposition - SVD). (Řešení úlohy identifikace zájmových osob může být kombinací obou těchto metod).

Samotnou oblast identifikace lze rozdělit do několika procesů a aktivit:

proces *lokalizace* - detekce obličeje

proces *zpracování* ohraničeného prostoru v obraze definovaného jako objekt třídy obličej (hlava)

proces *rozpoznání* dominantních částí obličeje

proces *zjišťování* charakteristických a jedinečných vlastností obličeje

proces *identifikace* - porovnání se vzorem známých fotografií, a to buď statistickými metodami pravděpodobnosti použitím klasifikačních algoritmů, nebo vyhodnocováním jiných významných reprezentací zkoumaného obrazu k přiřazení nebo setřídění množiny fotografií známých osob v pořadí od největší pravděpodobnosti ztotožnění (tedy od největší podobnosti) k nejnižší.

2. Programátorská dokumentace

Návrh řešení, co je použito

Metodami založenými na počítačové grafice se nebudeme zabývat, protože tyto metody nebudeme implementovat. Metody založené na statistice jsou obecně založené na výpočtu charakteristických rysů obličeje. Tyto rysy, jak už bylo podotknuto, odpovídají např. oblasti očí, nosu, úst, obočí, ale také to může být barva kůže. V našem softwaru jsme tedy použili pro identifikaci osob podle obličejů metodu založenou na statistice a konkrétně jsme využili *metodu PCA*.

Automatizované rozpoznávání lidských obličejů je obtížný komplexní úkol z důvodů proměnlivosti základních fyzikálních veličin obrazu, jakosti a fotometrie, geometrie - úhlu natočení a přiblížení, morfologie změn - emoční výrazy obličeje a stárnutí - a "přestrojení" (čepice, brýle, vousy). Odtud vyplývá nutnost vytvoření normalizovaného modelu lidského obličeje tak, aby scénář rozpoznávání nebyl ovlivněn těmito reálnými, nicméně rušivými vlivy.

Tyto rušivé vlivy, jako úhel natočení, přiblížení či posunutí nejsou v našem softwaru přímo řešeny. Námi získané obrázky obličejů jsou pro demonstraci staženy z databáze obrázků ATT. Je jich 400 kusů, kde každému jednotlivci odpovídá 20 obrázků s různou intenzitou jasu či pootočením apod.

Volba databáze pro ukládání vektorů popisujících obličeje. V první fázi implementace budeme rozhraní pro vkládání implementovat jednoduchým polem. Popřípadě když bude čas a jako variantu testování, bychom se pokusili databázi vyřešit např. pomocí M-stromu.

Výpočet kovarianční matice je snadný. Z kovarianční matice budeme počítat vlastní čísla a vlastní vektory. Výpočet vlastních čísel a vektorů ponecháme prozatím na některém z matematických programů nebo použijeme program z Rka (*autor: M. Krátký*). Nakonec jsme tento výpočet vyřešili použitím programu z Rka.

Počet vektorů, které tvoří bázi by se měl pohybovat mezi 10-15.

Zabývali jsme se volbou metrik pro měření vzdáleností mezi vektory. Pro testování budeme mít implementovány dvě metody: euklidovou a kosínovou.

Projekt bude implementován v jazyce C++. Uživatelské rozhraní je řešeno příkazovým řádkem pomocí jednoduchých příkazů. Bude implementována perzistence databáze.

Načtení obrázků

Formát obrázků je pgm o velikosti 112x92. Digitální obraz (fotografie) je reprezentován obrazovou maticí, která vznikne z analogového obrazu procesem vzorkování (diskretizace - rozdělení - v ploše obrazu). Obrázek čteme bit po bitu, přičemž převedme obrázek na vektor. Z matice 112x92 vytvoříme pomocí tzv. C-KŘIVKY (kdy každý nový načtený řádek matice se dává za předchozí) vektor o velikosti 10304. Pro rychlé výpočty je nutné snížit tuto dimenzi (900/1600). Tuto dimenzi jsme se snažili snížit jakousi metodou převzorkování, ovšem problém při převzorkování spočívá v tom, aby se z výsledného vektoru neztratily charakteristické vlastnosti obličeje. Těchto n obrázků převedených na vektory pak vstupuje do metody PCA.

Metoda PCA

Metoda PCA - ZÁKLADNÍ ANALÝZY KOMPONENT (PRINCIPAL COMPONENT ANALYSIS - PCA). Počítá pomocí vlastních vektorů charakteristický vektor pro obličej. Tím dojde k řádovému snížení dimenze vektoru a k řádovému zrychlení výpočtu.

Do metody vstupuje n obrázků převedených na vektory. Z vektorů obrázků se spočítá tzv. kovarianční matice. Z kovarianční matice se spočítají vlastní čísla a k nim vlastní vektory. Z vlastních vektorů se spočítá vektor o snížené velikosti - cca 10 (když nastane v porovnávání velký zlom, už se to dále nebere v úvahu). Vektor o velikosti 10 je dostatečně přesný pro charakterizaci obličeje a zároveň je dostatečně malý pro rychlé zpracování a výpočty.

Důležité je že výpočet vlastních čísel a vektorů probíhá pouze jednou v rámci předzpracování a pro nový obličej se používá pouze transformace obličeje na vektor o velikosti 10. Výpočet vlastních čísel a vlastních vektorů je ponechán na podprogramu z Rka (*p. M. Krátký*). Bázi tvoří 10 vektorů.

Vytvoření báze z části obrázků:

- 20 libovolných obličejů
- převod na vektory - s_i
- průměrný vektor - $p = \sum s_i / 20$
- 20 rozdílů od průměrného vektoru - $r_i = p - s_i$
- 20 matic $A_i = r_i * r_i^T$
- průměrná matice $C = \sum A_i / 20$
- výpočet vlastních čísel a vlastních vektorů
- počet vlastních čísel - míra nepřesnosti

Podproblémy spojené s PCA a implementací

Nevyužili jsme knihoven OpenCV, software je implementován v C++ s vyjímkou výpočtu vl. čísel a vektorů v Rku.

Převod obrázku na vektor a upravení velikosti vektoru - zde používáme tzv.

c-křivku a dostaneme vektor o velikosti 10304. Poté použijeme nějakou metodu (převzorkování) na redukci velikosti vektoru na hodnotu o řád nižší. Problém je ve volbě té správné a vhodné metody. Pro nekompatibilitu spolupráce s OpenCV jsme nevyužili metodu *Konvoluce s Gaussianem* pro převzorkování (ta spolupracuje s OpenCV).

Zařazení či nezařazení výpočtu průměrné matice do projektu. Pro testování zkusíme před výpočet kovarianční matice výpočet zařadit a výsledky porovnat s výsledky při nezařazení výpočtu průměrné matice.

Větší problémy se zpracováním programu na řešení rozpoznávání obličejů jsme neměli.

3. Varianty a testování

Primárně budeme testovat výsledky pro metodu PCA, která může být doplněna o výpočet průměrné matice.

Vektory redukované pomocí PCA budou měřené pomocí minimálně dvou metrik. Volba metrik pro měření vzdáleností mezi vektory. Euklidova a kosínova metrika. Testování bez a s filtrem na převzorkování vstupního vektoru.

4. Implementace

Software je implementován v jazyce C++ (v prostředí MS VISUAL STUDIO .NET 2003), jen s výjimkou výpočtu vlastních čísel a vlastních vektorů je použito Rko.

Popis jednotlivých tříd:

USERINTERFACE, implementace třídy uživatelského rozhraní

PICTUREREADER, implementace třídy pro načítání obrázku ze souboru

MATRIX, třída, jenž implementuje matici

DATABASE (řešeno polem), ukládá matice do paměti a vybírá podobné

EUCLIDSMETRIC, COSINEMETRIC, impl. tříd euklidovy a kosínovy metriky

DATABASEPERZISTENCY, pro práci s databází (ukládání a nahrávání databáze)

BASE, báze redukce velikosti vektorů

BASEPERZISTENCY, pro ukládání a nahrávání báze

Podrobnější popis tříd a jejich metod je přímo v kódu implementace.

5. Používaná databáze

Používaná databáze obrázků je z archívu Cambridgské univerzity, viz <http://www.uk.research.att.com/facedatabase.html>

6. Uživatelská dokumentace

Program se spouští příkazem (souborem) *faces.exe* a ovládá se pomocí příkazové řádky různými typy přepínačů.

Syntaxe příkazů je následující:

faces -help - vypíše nápovědu

faces -insert picture.pgm - vloží obrázek do databáze

faces -create picture1.pgm picture1.pgm ... - vytvoří bázi pro výpočet PCA ze zadaných obrázků

faces -db dbname - určí databázi, se kterou se bude pracovat

faces -find picture.pgm - vyhledá obrázky, které jsou zadanému obrázku podobné

faces -base basename - určí, která base bude použita při redukci dimenze vektoru

faces -metric metricId - nastaví používanou metriku (implicitně euklidova-1, kosinova-2)

faces -th number - nastaví minimální hodnotu vzdálenosti, která bude považována za úspěšné vyhledání (implicitně 0.85).

PŘÍKLADY POUŽITÍ PROGRAMU:

faces -find picture1.pgm -db database1 -base base - hledá picture1.pgm v databázi database1 a pro redukci použije bázi base

faces -create picture1.pgm picture2.pgm -base base - z picture1.pgm a picture2.pgm vytvoří bázi base

PŘÍKLADY POUŽITÍ JSOU V DÁVKOVÝCH SOUBORECH:

base.bat, find.bat, insert.bat, insert2.bat

7. Rozdělení práce mezi členy týmu

Vlastimil Hopjan analýza, návrh, uživatelské rozhraní, výstupy; odladění projektu

David Pilař implementace databáze, metriky, třída matrix; odladění projektu

Jiří Sovadina propojení PCA s programem na výpočet vl. čísel a vl. vektorů

Marek Vojtek zpracování obrázku, převod obrázku na vektor, dokumentace

8. Výsledky

Báze je tvořena 20-ti obrázky obličejů s1/1.pgm, s1/2.pgm, ..., s1/10.pgm, s2/1.pgm, s2/2.pgm, ..., s2/10.pgm.

V databázi je uloženo 23 obrázků obličejů s1/1.pgm, s1/2.pgm, ..., s1/10.pgm, s2/1.pgm, s2/2.pgm, ..., s2/10.pgm, s3/1.pgm, s5/2.pgm, s7/3.pgm.

Vyhledávaly se 4 obrázky obličejů s5/3.pgm, s1/6.pgm, s2/2.pgm, s4/9.pgm s nastavením hodnoty trashold 0,9925 pomocí euklidovy metriky a následně kosínovy metriky.

Vzdálenosti u euklidovy metriky (tučně označeny vzdálenosti u vybraných obličejů):

obličej	s5/3.pgm	s1/6.pgm	s2/2.pgm	s4/9.pgm
s1/1.pgm	0.992061	0.991283	0.990658	0.993631
s1/2.pgm	0.993425	0.994224	0.989643	0.993092
s1/3.pgm	0.988783	0.986252	0.99107	0.990442
s1/4.pgm	0.991207	0.994869	0.985787	0.988831
s1/5.pgm	0.991036	0.992606	0.989843	0.991012
s1/6.pgm	0.992779	0.999999	0.987479	0.990606
s1/7.pgm	0.993792	0.993443	0.988044	0.99188
s1/8.pgm	0.987704	0.989189	0.988416	0.988972
s1/9.pgm	0.990557	0.993016	0.9862	0.988948
s1/10.pgm	0.988254	0.98931	0.990505	0.989986
s2/1.pgm	0.989232	0.989816	0.989621	0.989394
s2/2.pgm	0.98947	0.987479	0.999999	0.991601
s2/3.pgm	0.989153	0.98851	0.993609	0.990452
s2/4.pgm	0.988745	0.986721	0.994257	0.990554
s2/5.pgm	0.990869	0.99097	0.991782	0.991946
s2/6.pgm	0.991064	0.989236	0.995864	0.992874
s2/7.pgm	0.988187	0.988543	0.98861	0.988582
s2/8.pgm	0.990987	0.989876	0.993267	0.991649
s2/9.pgm	0.988459	0.98676	0.992815	0.991577
s2/10.pgm	0.990961	0.991253	0.99282	0.991872
s3/1.pgm	0.993226	0.990471	0.987996	0.993487
s5/2.pgm	0.996368	0.991499	0.989777	0.995907
s7/3.pgm	0.990858	0.991736	0.988913	0.990125

Vzdálenosti u kosínovy metriky (tučně označeny vzdálenosti u vybraných obličejů):

obličej	s5/3.pgm	s1/6.pgm	s2/2.pgm	s4/9.pgm
s1/1.pgm	0.0500842	0.0500672	0.0501752	0.0501237
s1/2.pgm	0.0500612	0.0500557	0.0501323	0.0500872
s1/3.pgm	0.0501253	0.0500831	0.0502514	0.0501707
s1/4.pgm	0.0500285	0.0500434	0.0500692	0.050033
s1/5.pgm	0.0500729	0.0500753	0.0501646	0.050101
s1/6.pgm	0.0500357	0.0500512	0.0500865	0.0500459
s1/7.pgm	0.0500586	0.0500458	0.0501095	0.0500731
s1/8.pgm	0.0500628	0.0500694	0.0501744	0.0501058
s1/9.pgm	0.0500371	0.0500468	0.0500896	0.0500489
s1/10.pgm	0.050087	0.0500883	0.050214	0.050134
s2/1.pgm	0.0500878	0.0500834	0.050195	0.0501178
s2/2.pgm	0.0501199	0.0500865	0.0502785	0.0501683
s2/3.pgm	0.0501011	0.0500834	0.0502426	0.0501426
s2/4.pgm	0.0501352	0.0500998	0.0502851	0.0501821
s2/5.pgm	0.0500877	0.0500781	0.0501987	0.0501252
s2/6.pgm	0.0501004	0.0500719	0.0502349	0.0501432
s2/7.pgm	0.0500816	0.0500753	0.0501897	0.0501145
s2/8.pgm	0.0500916	0.0500705	0.0502127	0.0501256
s2/9.pgm	0.0501096	0.050078	0.0502535	0.050169
s2/10.pgm	0.0500849	0.0500769	0.0502031	0.0501209
s3/1.pgm	0.0500686	0.0500356	0.0501226	0.0500985
s5/2.pgm	0.0500758	0.0500357	0.0501332	0.0501022
s7/3.pgm	0.0500838	0.0500809	0.0501672	0.050105

9. Závěr

Po analýze a zhodnocení zpracovat a implementovat problém identifikace člověka, a to podle rozpoznávání obličejů nám jako neoptimálnější v našem případě přišlo použití metody PCA k nalezení charakteristických vlastností obličeje. Problém by se dal samozřejmě řešit i za pomoci jiné metody (např. SVD), nebo jiného přístupu či zpracování celé problematiky rozpoznávání osoby podle obličeje, např. některou z grafických metod, ovšem to by bylo více náročnější a rozsáhlejší.

Námi vyhotovený program na rozpoznávání obličejů je proveden v jednoduché podobě, má různá omezení, soubory obrázků obličejů jsou ve formátu *.pgm, program má omezenou databázi, a nezpracovává rušivé vlivy, jako jsou úhel natočení, přiblížení či posunutí, ale svou demonstrační funkci ze zadání plní.

Autoři:

Vlastimil Hopjan, hop007, hopjan@luvasoft.com

David Pilař, pil032, pilar.david@seznam.cz

Jiří Sovadina, sov015, ji.so@post.cz

Marek Vojtek, voj086, old-shatterhand@email.cz