

Cvičení z předmětu Biometrie

Porovnávání otisků prstů

P. Vostatek

November 2, 2011

1 Úloha 3, Otisk prstu - srovnávání otisků (15 bodů)

Termín odevzdání: 16. 11. 2011, 11:00 (odevzdání bude probíhat osobně na posledním cvičení bloku).

Na coursewaru jsou k dispozici zdrojové kódy k úloze Porovnávání otisků, stáhněte si je, rozbalte a nastavte si rozbalený adresář jako aktuální v Matlabu. Porovnávání otisků prstů bude mít 2 hlavní části:

1. Porovnání pomocí markantů
2. Porovnání pomocí FingerCodu (Gaborova filtrace)

Mezi zdrojovými kódy cvičení je opět připraven skript `main_batch.m` který shrnuje všechny části včetně obrazového výstupu.

Hlavním úkolem cvičení bude naprogramování 2 funkcí pro porovnání (matching) otisku prstu s databází obrázků. Otisky vstupují do funkcí už vzájemně zarovnané podle jejich jádra. Po implementaci kódů bude za úkol zhodnotit použitelnost uvedených algoritmů pro porovnání otisků.

Požadavky:

1. Implementujte funkci "match" pro výpočet shody mezi zarovnanými markanty mezi vstupním obrázkem a vzorem v databázi.

Syntaxe je následující:

```
[matchingScore, nbmatch, inputmatch, dbmatch] = match(mAi1,  
mAi2);
```

Vstupem jsou pole s vyznačenými markanty pro vzor (m_{Ai1}) a porovnávaný otisk (m_{Ai2}). Tato pole jsou přímo výstupem funkce `findminutia`, přičemž m_{Ai2} musí být zarovnaný s m_{Ai1} pomocí funkce `align2`. Ukázky jsou v `main_batch`.

Algoritmus: Vstupem jsou dvě sady markantů (bodů) v rovině: m_{Ai1} a m_{Ai2} a práh pro vzdálenost d . Vstupní počty markantů jsou: $|m_{Ai1}|$,

resp. $|m_{Ai2}|$, počet párů markantů iniciovaný $nbmatch = 0$ a celková vzdálenost $dist = 0$.

- Pro každý bod z m_{Ai2} naleznete nejbližší bod v m_{Ai1} . Pokud jsou markanty vzdáleny méně, než d , potom je označte za pár a vyjměte z m_{Ai1} i m_{Ai2} . Za každý nalezený pár $nbmatch + 1$

- Po vyčerpání všech markantů z m_{Ai2} možných spárovat vypočtete $matchingScore = \frac{2 \cdot nbmatch}{|m_{Ai1}| + |m_{Ai2}|}$, což je výsledné ohodnocení přiřazení otisků. Platí, že čím vyšší $matchingScore$, tím spíše patří otisky stejnému původci (interval je $< 0, 1 >$).

2. Implementujte funkci "fingercode_creation":

```
[fingercode Fmasked] = fingercode_creation(imOriginal, Gfilt, core, maskSize, dia),
```

imOriginal je vstupní obrázek pro výpočet, *Gfilt* je sada k Gaborových filtrů vytvořená pomocí `GaborFilter_creation`, *core* jsou souřadnice jádra otisku, *maskSize* je velikost bloku pro filtraci obrazu a *dia* jsou velikosti vnějšího a vnitřního poloměru ořezu okolo jádra.

Pomocí sady Gaborových filtrů s různým natočením filtrujte otisk prstu. Vznikne k filtrovaných obrazů. Každý filtrovaný obrázek zpracujte po blocích N velikosti $n \times n$ bez překryvu. Hodnotu pro každý blok spočítejte jako $f(N) = |mean(N) - std(N)|$, kde std značí směrodatnou odchylku, $mean$ střední hodnotu. Z každého bloku vznikne následně jeden pixel. Blokovaný obraz poté vyřízněte maskou pouze okolo jádra otisku, jak ukazuje Figure 1.



Figure 1: Postup při vytváření FingerCodu

Následný příznakový vektor f_1 (odpovídající vzoru, resp. f_2 odpovídající vstupnímu obrazu) vznikne seřazením hodnot ležících v mezikruží (nevynulované hodnoty Figure 1 vpravo) z každého z k obrazů a jejich seřazením za sebe. Porovnání 2 otisků poté proveďte jako $matchingScoreGabor = mean(abs(f_1 - f_2))$. Čím nižší je výsledná hodnota, tím lépe si otisky odpovídají.

2 Výsledky

Po implementaci funkcí vyzkoušejte jejich funkčnost na přiřazování otisků v přiložené databázi. Nastavte hodnoty pro získání segmentace a os-

tatních výstupů z minulého cvičení, které vám subjektivně připadaly ideální. Poté otestujte, zda je signifikantní rozdíl mezi hodnotami výsledného hodnocení u odpovídajících dvojic otisků od dvojic, které neodpovídají stejnému prstu. Zkuste kombinovat výstup markatového přiřazování a ohodnocení pomocí FingerCodu. Zkuste vymyslet příznak, který by nejlépe oddělil odpovídající otisky od neodpovídajících dvojic. Správnost zarovnání otisků u odpovídajících si dvojic kontrolujte vizuálně pomocí zarovnaných koster.

Dvě výše uvedené funkce zabalte do zipu a nahrajte do coursewaru. Odevzdání bude probíhat ústně na posledním cvičení bloku.