

# Cvičení z předmětu Biometrie

## Rozpoznávání otisků prstů

P. Vostatek

October 24, 2012

### 1 Úloha 2, Otisk prstu - předzpracování otisků (5 bodů)

Termín odevzdání: 14. 11. 2012, 23:59

Na coursewaru jsou k dispozici zdrojové kódy, stahněte si je, rozbalte a nastavte si rozbalený adresář jako aktuální v Matlabu. Předzpracování otisku se dělí na několik částí:

1. Segmentace otisku prstu od pozadí (*segmentimage2.m*)
2. Výpočet orientací papilárních linií v otisku (*computeorientationarray.m*)
3. Výpočet frekvenčního pole otisku (*computelocalfrequency.m*)
4. Vylepšení obrazu použitím Gaborových filtrů a tvorba kostry otisku (*enhance2ridgevalley.m*)
5. Vyčištění chyb v kostře a vyhledání markantů (*cleanskeleton.m* a *find-minutia.m*)

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

Úkolem cvičení je vybrat si z připravených databází otisky prstů od 5–10 uživatelů a od každého vybrat 2 otisky. Vyberte si otisky z různých databází a od různých lidí tak, aby jejich kvalita byla různá — kvalita obrazu, umístění v obraze, částečné otisky apod. Poté je předzpracujte pro pozdější identifikaci a nakonec zhodnoťte kvalitu nalezených markantů v jednom kvalitním otisku a v jednom nekvalitním.

#### Postup:

1. Segmentace otisku probíhá nejprve upravením kontrastu obrazu a poté výpočtem gradientu pro každý pixel. Prahováním velikosti gradientu jsou nalezeny místa v obraze obsahující samotný otisk. Hodnota prahu je v podstatě jedinou nutnou hodnotou, kterou je potřeba nastavit při předzpracování obrazu otisku.

Pokud nechcete nastavovat ručně hodnotu prahu pro každý obraz, můžete implementovat funkci, pro automatické určení prahu, případně úplně novou funkci/metodu na segmentaci otisku. Za takovou implementaci dám 3 bonusové body k úloze.

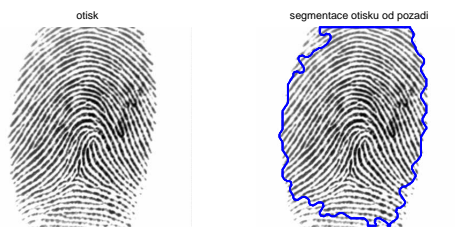


Figure 1: Segmentace, `segmentimage2(im, 4)`

Můžete zkoušet vliv dalších parametrů na výsledek segmentace otisku. Těmi parametry jsou: velikost okna na vyhlazení gradientního obrazu, velikost okna na vyhlazení výsledného obrazu a zákaz upravení kontrastu obrazu před segmentací.

Dále jsou na obrázcích ukázány další kroky při předzpracování obrazu:

2. Výpočet orientace papilárních linií:

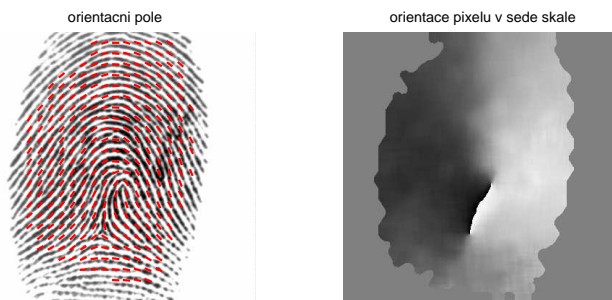


Figure 2: Orientační pole, `computeorientationarray(im, imSegmented, 10)`

3. Výpočet frekvenčního pole.

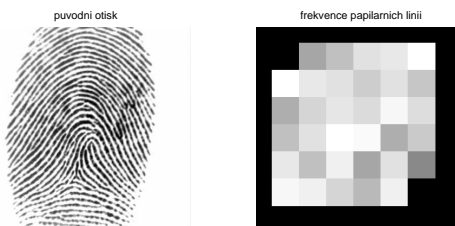


Figure 3: Frekvenční pole, `computelocalfrequency(im, imSegmented, orientationArray)`

4. Filtrace použitím Gaborových filtrů. Zde je za úkol do kódu `enhance2ridgevalley.m` dolžit rovnici pro výpočet masky Gaborova filtru. Vyzkoušejte G. filtraci vypnout a sledujte vliv na výsledný obrázek.

$$x = \langle -16, 16 \rangle, y = \langle -16, 16 \rangle$$

$$x_p = \sin(\text{angle}) \cdot x + \cos(\text{angle}) \cdot y$$

$$y_p = \sin(\text{angle}) \cdot y - \cos(\text{angle}) \cdot x$$

$$gab(x, y) = \exp\left\{-\frac{1}{2} \cdot \left[\left(\frac{x_p}{\sigma_x}\right)^2 + \left(\frac{y_p}{\sigma_y}\right)^2\right]\right\} \cdot \cos(2\pi f \cdot x_p)$$

obraz vylepseni pouzitim Gaborovych filtru



kostra otisku pred cistenim

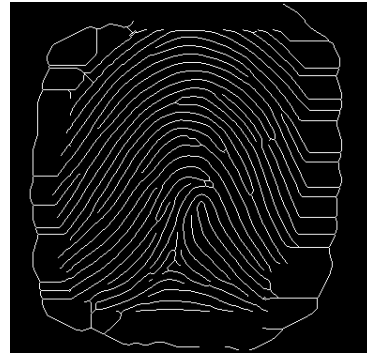


Figure 4: Gaborova filtrace, `enhance2ridgevalley(im, imSegmented, orientationArray, frequencyArray, 0)`

5. Po filtraci Gaborovými filtry jsou papilární linie ztenčeny na nejmenší šířku - vzniké kostra otisku. V kostře otisku jsou poté hledány markanty např. pomocí jednoduchých pravidel. Zhodnoťte kvalitu nalezených markantů ve dvou zvolených otiscích.

kostra otisku po vycistení



v otisku vyznacene markanty

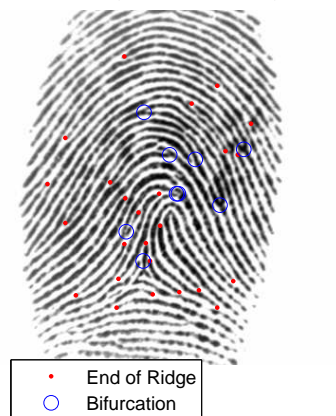


Figure 5: Finalní detekce markantů, defaultní nastavení.

Při správné implementaci masky Gaborova filtru by měla funkce ukazka vrátit obrázky podobné, jako jsou zde. O práci sepište stručný protokol, doplněný obrázky vybraných otisků a výsledné segmentace. Napšte, jaký měla podle Vás vliv na kvalitu segmentace počáteční úprava kontrastu obrazu a jaký vliv měla Gaborova filtrace. Odevzdání úlohy bude probíhat najednou i druhou částí.