

Chapter 1

Detectia zonelor de piele

1.1 Punerea problemei

Detectia pixelilor de piele este o problema de segmentare. In condițiile in care de multe ori se urmareste detectia tuturor zonelor de piele (față, gat, antebraț, palme, etc.) care sunt uneori partial acoperite (deci cu forma neregulata), abordarea nu se poate baza pe metode de contur sau în care forma regiunii este cunoscută și utilizată. In principiu, metodele de detectie a pielii se bazeaza pe algoritmi de praguire pe culoare.

Ca și utilitate practică detectia pielii este folosită în:

- Algoritmi de detectie a fețelor. De multe ori este utilizat'a pentru a elimina diverse obiecte care au structura geometrica a unei fete dar carora le lipseste croma potrivita. De asemenea, cnd se urmareste o fata intr-o secventa video, iar platforma pe care este implementat algoritmul nu este suficient de puternica, algoritmul de urmarire utilizeaza in special culoarea.
- Infrumusetarea artificiala a pielii. In special in cazul persoanelor de sex feminin, din punct de vedere estetic, este de dorit un ten neted, fara riduri, cosuri, cicatrici sau alte artefacte care dau un aspect neplacut. De obicei, un prim pas intr-o asemenea aplicatie este detectia zonelor de interes (piele).
- Aplicatii pentru control parental automat. In acest caz se doreste limitarea accesului minorilor la imagini sau secvente video cu caracter erotic sau obscen explicit. Un criteriu de baza pentru identificarea unor asemenea imaginii este numarul de pixeli de culoarea pielii existent in imagine.

1.2 Acuratetea unui algoritm

Obiectivul oricarui algoritm de detectie a pielii este identificarea czt mai corecta a pixelilor de acest tip. De obicei comparatia se face cu imagini marcate manual.

1.2.1 Masurarea acuratetii

In continuare vom considera ca I^a este o imagine binara ce contine zonele de piele si de fundal obtinuta in mod automat, iar I^r este imaginea binara obtinuta prin marcare manuala. Ambele imagini vor avea valoarea 1 in locatiile identificate ca fiind de interes (in cazul de fata zonele de piele) si 0 in rest.

Pentru validarea unui algoritm se utilizeaza, 'in principal, urmatoarele masuri :

- Rata de detectie corecta:

$$CR = \frac{100 \sum_i^N \sum_j^M (I^a(i, j) \& I^r(i, j))}{\sum_i^N \sum_j^M I^r(i, j)} \quad (1.1)$$

Rata de corectie poate avea doar valori subunitare (mai mici de 100%). O valoare mica implica faptul ca algoritmul utilizat are performante slabe.

- Precizie:

$$Precision = \frac{TP}{TP + FP} \quad (1.2)$$

Precizia este raportul dintre câte eşantioane pozitive a găsit algoritmul şi suma tuturor eşanţionelor etichetate ca pozitive . În cazul nostru câţi pixeli din cei detectaţi de algoritm chiar sunt pixeli de piele?

- Recall:

$$Recall = \frac{TP}{TP + FN} \quad (1.3)$$

Recall-ul reprezintă raportul dintre eşantioanele etichetate ca fiind pozitive şi toate eşantioanele care sunt într-adevăr pozitive. În cazul nostru, dintre toţi pixelii marcaţi ca fiind piele câţi a reuşit algoritmul să recunoască?

- Procentul de falsuri negative (numarul de pixeli de piele identificati ca fiind de fundal):

$$FN = \frac{100 \sum_i^N \sum_j^M ((\neg I^a(i, j)) \& I^r(i, j))}{\sum_i^N \sum_j^M I^r(i, j)} \quad (1.4)$$

O valoare mare pentru FN presupune ca algoritmul este prea strict.

- Procentul de true positive (numarul de pixeli de piele identificati ca fiind piele):

$$TP = 1 - FN \quad (1.5)$$

- Procentul de falsuri pozitive (numarul de pixeli de fundal identificati ca fiind de piele):

$$FP = \frac{100 \sum_i^N \sum_j^M (I^a(i, j) \& (\neg I^r(i, j)))}{\sum_i^N \sum_j^M I^r(i, j)} \quad (1.6)$$

O valoare mare presupune ca algoritmul este prea relaxat. Teoretic, ambele erori (FN si FP) pot avea si valori supraunitare.

- Procentul de true negative (numarul de pixeli de piele identificati ca fiind piele):

$$TN = 1 - FP \quad (1.7)$$

ACTIVITATE: Construiti o functie Python care sa primeasc' a la intrare doua imagini, una segmentat'a automat si una segmenatata manual (imagini binare) si sa returneze cele trei valori care confera o masura a acuratetii unui algoritm. Ce inseamnă o precizie mare? Dar un recall mic?

1.3 Algoritmi pentru detectia pielii

1.3.1 Spatiul de culoare utilizat

Spatiul RGB Cel mai utilizat spatiu de culoare este RGB. In acest spatiu, o metoda foarte des utilizata datorita simplitatii ei este cea propusa de Peer et al.¹. Condițiile propuse de ei se bazeaza pe respectarea simultana a urmatoarelor principii:

1. In cazul unui iluminant natural, ziua, culoarea pielii este data de conditiile :

- (a) Nuantele generale:

$$C_1 = (R > 95) \& (G > 40) \& (B > 20) \quad (1.8)$$

- (b) Componentele de R, G, B nu trebuie sa fie prea apropiate una de alta, eliminand astfel gri-urile:

$$C_2 = (\max\{R, G, B\} - \min\{R, G, B\} > 15) \quad (1.9)$$

- (c) Pentru o complexitate normala, componentele de rosu si verde nu trebuie sa fie prea apropiate:

$$C_3 = (|R - G| > 15) \quad (1.10)$$

- (d) Cea mai puternica nuanta trebuie sa fie cea de rosu:

$$C_4 = (R > G) \& (R > B) \quad (1.11)$$

¹P. Peer, F. Solina, "An Automatic Human Face Detection Method", Proceedings of the 4th Computer Vision Winter Workshop (CVWW-99), pp. 122-130, Rastenfeld, Austria, 1999

2. Daca iluminantul este lateral sau de tip flash, atunci conditiile sunt:

(a) Nuantele sunt date de:

$$C_5 = (R > 220) \& (G > 210) \& (B > 170) \quad (1.12)$$

(b) Componentele de rosu si verde trebuie sa fie similare (nuanta trebuie s'a fie maronie):

$$C_6 = (|R - G| \leq 15) \quad (1.13)$$

(c) Componenta de albastru trebuie s'a fie cea mai pu'tin semnificativ'a (Avatars OUT :D)

$$C_7 = (R > B) \& (G > B) \quad (1.14)$$

In general un pixel, caracterizat de tripletul (R, G, B) este considerat a fi pixel de piele dac'a se respect'a condi'tia:

$$(C_1 \& C_2 \& C_3 \& C_4) | (C_4 \& C_5 \& C_6) == 1 \quad (1.15)$$

ACTIVITATE: Aplicatii algoritmul propus de Peer et al. pe setul de imagini existent, masurand acuratetea rezultatelor obtinute.

ACTIVITATE: Incercati sa scrieti algoritmul intr-un mod optimizat fãra sa mai parcurgeti pixelii imaginii. Exista vreo diferența semnificativa de timp in execuția programului clasic fatã de cel optimizat?

Andrei Racoviteanu
Ultima modificare: 26 septembrie 2019