

11. Detecția punctelor de muchie

11.1. Introducere

În această lucrare se va aborda problematica detecției punctelor de muchie din imagini digitale. Punctele de muchie sunt puncte în jurul cărora intensitatea imaginii suferă un salt brusc de-a lungul unei anumite direcții „x” (Fig. 11.1). Această variație de intensitate poate fi identificată prin detecția punctelor de maxim ale derivatei de ordin 1 a imaginii (gradientului: $\nabla f = f'$) sau prin detectarea trecerilor prin „0” ale derivatei de ordin 2 a imaginii (laplacianului: $\nabla^2 f = f''$).

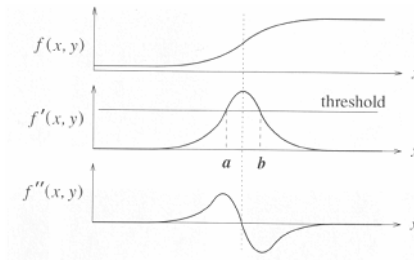


Fig. 11.1 Modalități de detecție a punctelor de muchie (zonelor în care intensitatea variază brusc).

În această lucrare se vor prezenta metode bazate pe detecția punctelor de maxim ale derivatei de ordin 1 (gradientul imaginii : $\nabla f(x,y)$).

11.2. Calculul gradientului imaginii

Gradientul într-un punct al imaginii este un vector care indică direcția de variație a intensității imaginii (Fig. 11.2) în jurul aceluși punct, modulul său fiind proporțional cu amplitudinea acestei variații raportată la unitatea de lungime (viteza acestei variații) (Fig. 11.1). Dacă punctele de muchie fac parte dintr-un contur (ca și în Fig. 11.2), gradientul într-un punct de muchie al acestui contur va fi perpendicular pe tangenta la acest contur în punctul respectiv.

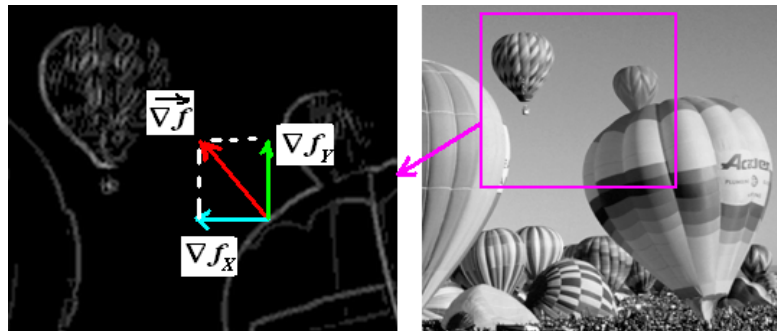


Fig. 11.2 Ilustrarea semnificației gradientului imaginii. În figura din stânga este reprezentată imaginea modulului gradientului.

Gradientul unei funcții continue f de 2 variabile se definește ca:

$$\nabla f(x, y) = \begin{bmatrix} \nabla f_x \\ \nabla f_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} = \begin{bmatrix} \lim_{\Delta x \rightarrow 0} \frac{f(x + \Delta x, y) - f(x, y)}{\Delta x} \\ \lim_{\Delta y \rightarrow 0} \frac{f(x, y + \Delta y) - f(x, y)}{\Delta y} \end{bmatrix} \quad (11.1)$$

Pentru imagini digitale gradientul se poate aproxima înlocuind Δx și Δy cu 1 în (11.1):

$$\nabla f(x, y) = \begin{bmatrix} \nabla f_x \\ \nabla f_y \end{bmatrix} = \begin{bmatrix} f[x+1, y] - f[x, y] \\ f[x, y+1] - f[x, y] \end{bmatrix} \quad (11.2)$$

Alte aproximări ale celor două componente ale gradientului se pot obține prin convoluția imaginii cu următoarele nuclee:

Prewitt:

$$\begin{aligned} \nabla f_x &= f(x, y) * \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \\ \nabla f_y &= f(x, y) * \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} \end{aligned} \quad (11.3)$$

Sobel:

$$\begin{aligned} \nabla f_x &= f(x, y) * \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \\ \nabla f_y &= f(x, y) * \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \end{aligned} \quad (11.4)$$

Roberts (cross):

$$\begin{aligned} \nabla f_x &= f(x, y) * \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \\ \nabla f_y &= f(x, y) * \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \end{aligned} \quad (11.5)$$

Ca orice mărime vectorială, gradientul este caracterizat de un modul și o direcție. Modulul gradientului se calculează cu formula (11.6) iar direcția cu formula (11.7):

$$|\nabla f(x, y)| = \sqrt{(\nabla f_x(x, y))^2 + (\nabla f_y(x, y))^2} \quad (11.6)$$

$$\theta(x, y) = \arctg\left(\frac{\nabla f_y(x, y)}{\nabla f_x(x, y)}\right) \quad (11.7)$$

11.3. Metoda Canny de detecție a muchiilor

Metoda propusă de Canny se bazează pe calculul gradientului imaginii dar în plus permite:

- maximizarea raportului semnal zgomot pentru o detecție corectă
- o localizare bună a punctelor de muchie
- minimizarea numărului de răspunsuri pozitive la o singură muchie singulară (eliminarea non-muchiilor)

Pașii metodei propuse de Canny sunt dați mai jos:

1. Filtrarea imaginii cu un filtru Gaussian pentru eliminarea zgomotelor.
2. Calculul modulului și direcției gradientului.
3. Suprimarea non-maximelor modulului gradientului.
4. Binarizarea adaptivă a punctelor de muchie și prelungirea muchiilor prin histereză.

11.3.1. Filtrarea imaginii cu un filtru Gaussian

Zgomotul din imagine este o informație de “frecvență înaltă” care se suprapune peste imaginea originală. Aceasta induce apariția unor puncte de muchie false. Zgomotul inherent procesului de achiziție al imaginilor are un model Gaussian și se poate elimina cu un filtru omonim. Procedura de filtrare a zgomotului Gaussian a fost prezentată în detaliu în lucrarea 10.

11.3.2. Calculul modulului și direcției gradientului

Calcularea modulului și direcției gradientului presupune alocarea a câte unui buffer temporar de dimensiunea imaginii și inițializarea elementelor lor în conformitate cu (11.6) și respectiv (11.7), unde componentele orizontale $\nabla f_x(x,y)$ și respectiv verticale $\nabla f_y(x,y)$ se pot calcula prin convoluția imaginii în conformitate cu (11.3) sau (11.4).

11.3.3. Suprimarea non-maximelor modulului gradientului

Are drept scop subțierea muchiilor prin păstrarea doar a punctelor de muchie care au modulul maxim al gradientului de-a lungul direcției de variație a intensității (de-a lungul direcției gradientului).

Primul pas constă în cuantificarea direcțiilor gradientului (calculate folosind (11.7)) în 4 intervale de câte 45° în conformitate cu Fig. 11.3:

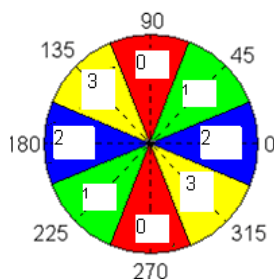


Fig. 11.3 Cuantificarea direcțiilor gradientului în procesul de eliminare a non-maximelor.

Presupunând, de exemplu, că direcția gradientului într-un anumit punct al imaginii este „1” (Fig. 11.4) modulul gradientului în punctul P este un maxim local dacă: $|\nabla P| > |\nabla I_6|$ și $|\nabla P| > |\nabla I_2|$. Dacă aceste condiții sunt îndeplinite, punctul P se reține. Altfel, se elimină.

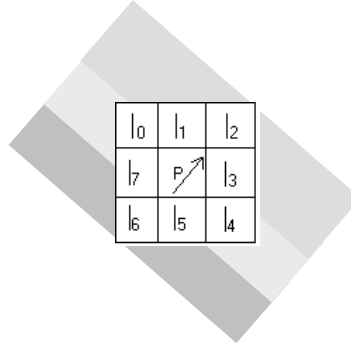


Fig. 11.4 Ilustrarea unui exemplu de calcul al punctului de muchie cu modulul maxim al gradientului.

11.3.4. Binarizarea adaptivă a punctelor de muchie și prelungirea muchiilor prin histereză

După efectuarea cu succes a calculului gradientului și a operației de suprimare a non-maximelor modulului gradientului, **se obține o imagine în care intensitatea pixelilor este egală cu valoarea modulului gradientului în acel punct**. De asemenea, grosimea punctelor de muchie din această imagine (a căror modul este nenul) este (ideal) de un pixel. În continuare se descriu pașii pentru obținerea muchiilor finale.

11.3.4.1. Binarizarea adaptivă

Binarizarea adaptivă încearcă să extragă un număr relativ constant de puncte de muchie pentru o dimensiune dată a imaginii. În acest fel se compensează iluminarea și contrastul diferit între imagini (un prag fix sau va scoate prea multe muchii, sau nu va scoate muchii deloc).

Parametrul care se dă procedurii de detecție a pragului este raportul dintre numărul de puncte de muchie și numărul de puncte cu modulul gradientului diferit de zero din imagine.

$$NrPuncteMuchie = p * (NumarPixeli - NumarPixeliCuModulGradientNul) \quad (11.8)$$

Parametrul p ia de obicei valori între 0.01 și 0.1.

Algoritmul este următorul:

1. Se calculează histograma de intensități a imaginii (a modulelor gradientului după suprimarea non-maximelor). Se vor normaliza aceste valori în intervalul $[0..255]$ (prin împărțire la $4\sqrt{2}$ în cazul în care gradientul a fost calculat utilizând operatorul Sobel). Rezultă o histogramă $Hist[0..255]$,

$$Hist[i] = \text{Numărul pixelilor având gradientul normalizat de modul } i \quad (11.9)$$

2. Se calculează numărul de pixeli diferiți de zero care nu vor fi puncte de muchie:

$$NrNonMuchie = (1-p) * (Height * Width - Hist[0]) \quad (11.10)$$

3. Se pornește de la poziția 1 din histogramă, și se însumează valorile histogramei pe măsură ce avansăm spre 255. Dacă la un moment dat suma este mai mare decât $NrNonMuchie$, atunci

ne oprim și poziția la care am ajuns este pragul pe care l-am găsit. Această tehnică, intuitiv, ne găsește valoarea intensității (*PragAdaptiv*) sub care se găsesc *NrNonMuchie* pixeli.

Atenție la marginile imaginii (pentru care nu s-a calculat gradientul)! Ele trebuie să fie zero sau să nu fie luate în considerare, deoarece valorile nedefinite în aceste zone pot influența valoarea pragului găsit.

11.3.4.2. Extinderea muchiilor prin histereză

Binarizarea cu pragul adaptiv nu garantează completitudinea muchiilor (părți umbrite ale obiectului sau prezența zgomotelor pot afecta procesul de detecție). Rezultatul va fi o imagine cu foarte multe muchii fragmentate.

De aceea se impune o tehnică de prelungire a muchiilor. Muchiile obținute prin binarizarea inițială cu pragul găsit la punctul anterior, care sunt muchii definitive, se caută a se prelungi cu muchii mai puțin clare, care nu trec testul binarizării cu acest prag, dar ar putea să apară ca muchii la un prag mai scăzut.

Formal, se definesc două praguri:

$$Prag_înalt = PragAdaptiv \quad (11.11)$$

$$Prag_coborât = k * Prag_înalt \quad (11.12)$$

Unde $k < 1$ (de exemplu, $k = 0.4$).

Imaginea modulului gradientului se parcurge pixel cu pixel. În imaginea rezultat se marchează pixelii cu intensitatea mai mare decât *Prag_înalt* ca MUCHIE_TARE (ex. 255), iar pixelii cu intensitatea mai mare decât *Prag_coborât* dar mai mică decât *Prag_înalt* se marchează ca MUCHIE_SLABĂ (ex. 128). Pixelii cu intensitatea mai mică decât *Prag_coborât* sunt considerați NON_MUCHII și sunt eliminați. Rezultatul acestei marcări (negativat) se vede în Fig. 11.5 stânga.

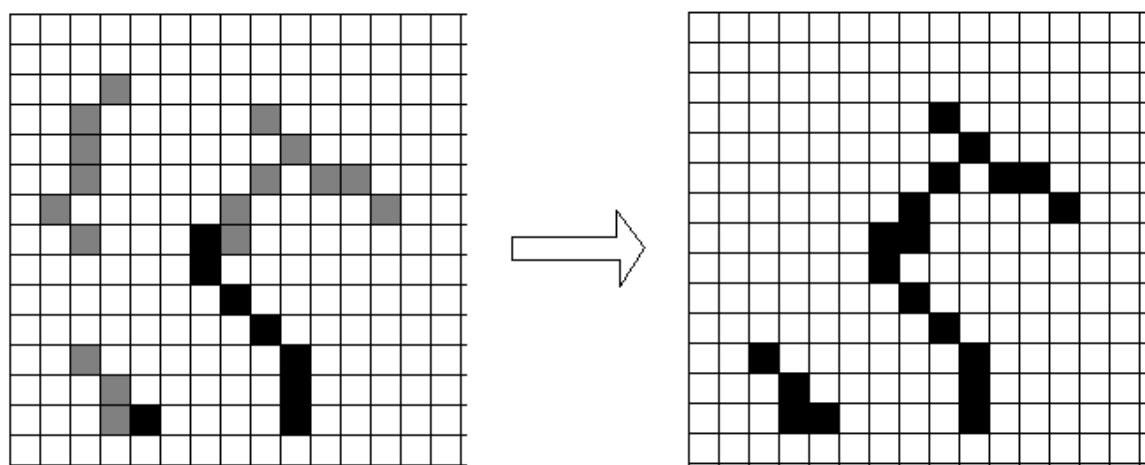


Fig. 11.5 Stânga: imaginea cu muchii tari și slabe; Dreapta: rezultatul prelungirii muchiilor tari și eliminarea celor slabe rămase.

Următorul pas este de a se extinde muchiile tari cu pixelii muchiilor slabe învecinate, ca în Fig. 11.5. Dacă un punct de muchie tare se învecinează cu un punct de muchie slabă, punctul de muchie slabă este marcat ca muchie tare. Acest punct la rândul său devine sursă de extindere.

Procesul se repetă până nu mai există puncte de muchie tare învecinate cu puncte de muchie slabă.

O implementare eficientă folosește o coadă pentru a ține coordonatele punctelor de muchie tare. Algoritmul rulează astfel:

1. Parcurge imaginea de la stânga la dreapta și de sus în jos și găsește primul punct de MUCHIE_TARE și pune coordonatele sale în coadă.
2. Cât timp coada nu este vidă
 - a) Extrage primul punct din coadă
 - b) Găsește toți vecinii de tip MUCHIE_SLABĂ ai acestui punct
 - c) Marchează în imagine vecinii acestui punct ca puncte de MUCHIE_TARE
 - d) Pune coordonatele acestor puncte în coadă
 - e) Continuă cu următorul punct din coadă
3. Continuă de la pasul 1 cu următorul punct de MUCHIE_TARE.
4. Elimină toate punctele de MUCHIE_SLABĂ din imagine, dându-le valoare NON_MUCHIE (0).

O considerație finală se impune la definiția “învecinării” unui punct de muchie tare cu un punct de muchie slabă. Se poate considera învecinarea directă (pixeli adiacenți), în vecinătate de 4 sau de 8, sau se poate tolera o distanță oarecare (1, 2 pixeli). Un argument pentru aceasta ar putea fi că, din cauza zgomotului, muchiile pot să fie întrerupte pe distanțe mici.

11.4. Activități practice

Activitățile practice ale aceste lucrări se vor desfășura de-a lungul a două ședințe de laborator:

11.4.1. Laborator 1

1. Se vor calcula componentele orizontale ∇f_x și verticale ∇f_y ale gradientului prin convoluție cu măștile prezentate în (11.3) ... (11.5) și se vor vizualiza rezultatele (operația de convoluție a fost implementată în lucrarea 9).
2. Se vor calcula modulul (11.6) și direcția (11.7) gradientului în fiecare pixel al imaginii pentru cele trei aproximări ale gradientului (Sobel, Prewitt și Roberts) și se va vizualiza modulul gradientului într-o fereastră destinație.
3. Se va realiza binarizarea imaginii rezultat de la punctul 2 cu un prag fix, arbitrar ales.
4. Se vor implementa pașii 1-3 ai algoritmului Canny (pasul 1 – filtrarea gaussiană - a fost implementat în lucrarea 10, la pasul 2 se vor folosi rezultatele de la punctul 2 pentru operatorul Sobel, iar la pasul 3 se va implementa operația de subțiere a muchiilor). Se va vizualiza într-o fereastră destinație rezultatul obținut după aplicarea pasului 3. Se va compara cu rezultatul obținut prin aplicare simplă a operatorului Sobel (punctul 2).
5. **Salvați-vă ceea ce ați lucrat. Utilizați aceeași aplicație în laboratoarele viitoare. La sfârșitul laboratorului de procesare a imaginilor va trebui să prezentați propria aplicație cu algoritmi implementați!!**

11.4.2. Laborator 2

1. Se va implementa algoritmul de binarizare cu prag adaptiv și algoritmul de prelungire a muchiilor prin histereză. Se vor vizualiza rezultate intermediare pentru muchii tari, muchii slabe, și rezultatul final. Se va experimenta cu diferite criterii de învecinare, și cu diferiți parametrii p .

2. Se vor testa și vizualiza rezultatele finale ale aplicării detectorului de muchii Canny pe diverse tipuri de imagini.
3. **Salvați-vă ceea ce ați lucrat. Utilizați aceeași aplicație în laboratoarele viitoare. La sfârșitul laboratorului de procesare a imaginilor va trebui să prezentați propria aplicație cu algoritmi implementați!!**

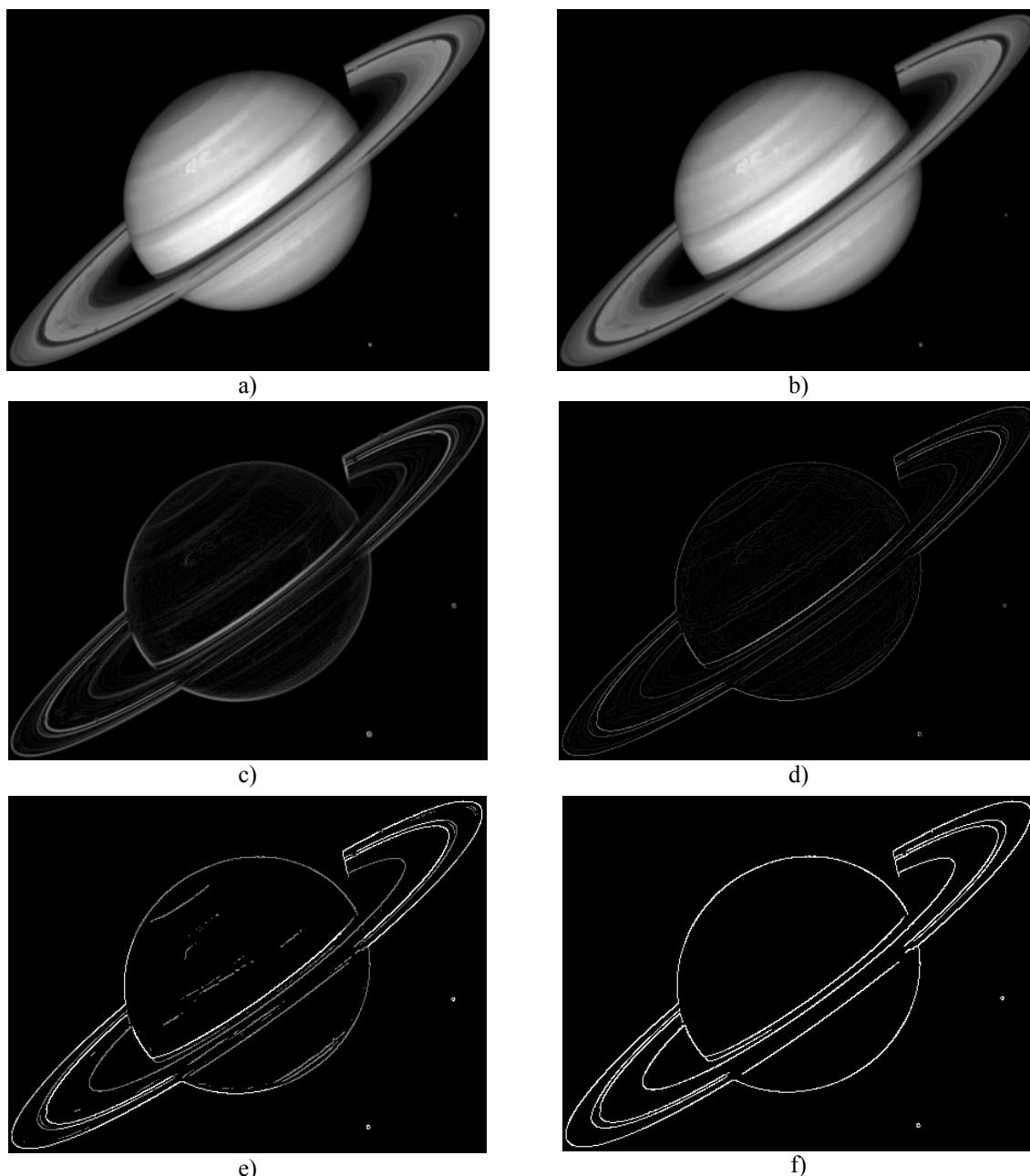


Fig. 11.6 Detecția muchiilor prin metoda Canny – exemplu de rezultat: a) imaginea inițială; b) după filtrarea Gaussiană ($\sigma = 0.5$); c) magnitudinea normalizată a gradientului (utilizând operatorii Sobel); d) după suprimarea non-maximelor; e) după binarizarea adaptivă ($p = 0.1$); f) muchiile finale după extinderea prin histereză și eliminarea muchiilor slabe

Bibliografie

- [1] E.Trucco, A.Verri, *Introductory Techniques for 3-D Computer Vision*, Prentice Hall, 2001
- [2] R.C.Gonzales, R.E.Woods, *Digital Image Processing, 2-nd Edition*, Prentice Hall, 2002