

Pràctica 6 PDS: Detecció de Contorns

Juan Gabriel Gomila Salas

1. Introducció

En la pràctica anterior, vam tractar el problema de donada una imatge segmentar-la en una sèrie de bocins amb característiques de color o textura en comú. En aquesta, tractarem un tema que té relació amb l'anterior: determinar els contorns de les figures que apareixen en la nostra imatge.

Un contorn d'una imatge és un conjunt de píxels connectats (en 8 connectivitat considerarem) que separa una regió d'una altra. Aquest concepte és molt clar en la visió humana: si observam una imatge sabem distingir perfectament els objectes que hi apareixen, els fons, el cel, la terra,... però per a una màquina és una informació que, com veurem, no és gens trivial de detectar.

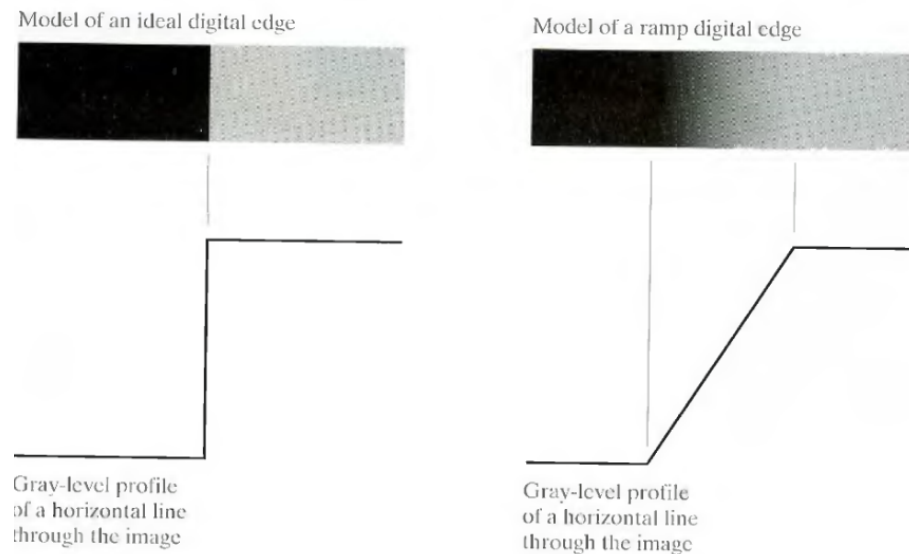


Figura 1: Noció de contorn d'una imatge digital

En la Figura 1, podem observar el model ideal de contorn d'una imatge: un canvi, una variació, més o manco brusca entre dos nivells de gris d'una imatge. En el primer cas, tenim el model ideal de contorn: una clara línia que separa dos nivells de gris diferents; però en el segon cas, hi ha un canvi gradual, per tant, el contorn serà un conjunt connex de píxels continguts dins la rampa de variació de color.

Aleshores, com podem fer una classificació que ens valgui per tots els casos i sigui, més o manco única (és a dir, no ens doni linees tan gruixudes com tota la rampa de variació de la segona figura)?

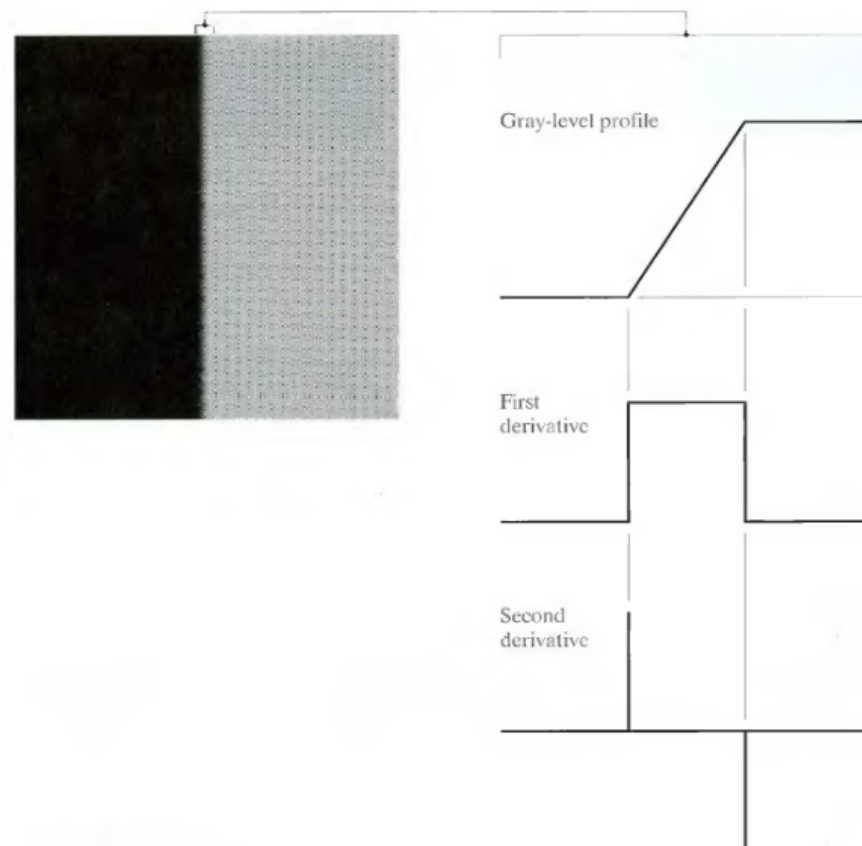


Figura 2: A la dreta, imatge amb dos nivells de gris diferents, a la dreta el perfil de la imatge, la seva primera i segona derivada.

En la Figura 2, podem observar una imatge amb dos nivells de grisos diferents, separats per tant per un contorn. A la seva dreta podem observar tres figures dife-

rents. En la primera d'elles, podem veure el perfil del nivell de gris entre les dues regions. Les altres dues representen la primera i la segona derivada del perfil inicial. La primera derivada és positiva en els punts de transició a la rampa si ens movem d'esquerra a dreta en el perfil inicial, és constant en tot punt de la rampa i zero en les regions de nivell de gris constant. Així mateix, la segona derivada és positiva en els punts de transició associats als costats obscurs de la imatge, negativa en els punts de transició associats als costats luminosos de la imatge i zero en els punts de la rampa i on el nivell de gris és constant.

Així doncs, per poder determinar vores, podem emprar les facultats de la primera i/o segona derivada: determinar els punts que pertanyen a la rampa, veure quins píxels pertanyen a les zones d'ombra-il·luminació del contorn. Per fer-ho emprarem diverses tècniques que hem englobat en dos tipus diferents: aquelles que es poden implementar aplicant una màscara a la imatge inicial, i d'altres una mica més complexes que necessiten més eines de programació.

Totes les proves les farem amb la mateixa imatge, per tal de poder comparar els resultats obtinguts: la de la nina que hem emprat ja al llarg de diferents pràctiques anteriors.

2. Tècniques de Mask Applying

La informació que en 1D ens dóna la derivada, en 2D ens la dóna el vector gradient:

$$\nabla I = (I_x, I_y)$$

que ens descriu la direcció de màxima variació, en el nostre cas, la direcció de màxima variació del nivell de gris de la imatge. Per tant, una manera de saber on hi ha els canvis més bruscos, és cercar aquells punts on el mòdul del gradient és màxim. Bàsicament les tres tècniques que veurem aquí es basen en tres maneres diferents de discretitzar el mòdul gradient d'una imatge, de manera lineal, aplicant un filtre 2 o 3D com el de la figura

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

Vegem cadascuna d'elles una mica més a fons.

2.1. Sobel Edge Detection

Seguint la notació de la matriu 3×3 anterior, podem discretitzar les derivades direccionals, I_x i I_y de la funció amb les dues màscares següents:

-1	-2	-1	Sobel
0	0	0	
1	2	1	

$$G_x = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)$$

-1	0	1
-2	0	2
-1	0	1

$$G_y = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)$$

Figura 3: Màscares corresponent a les derivades direccionals I_x i I_y del mètode de Sobel.

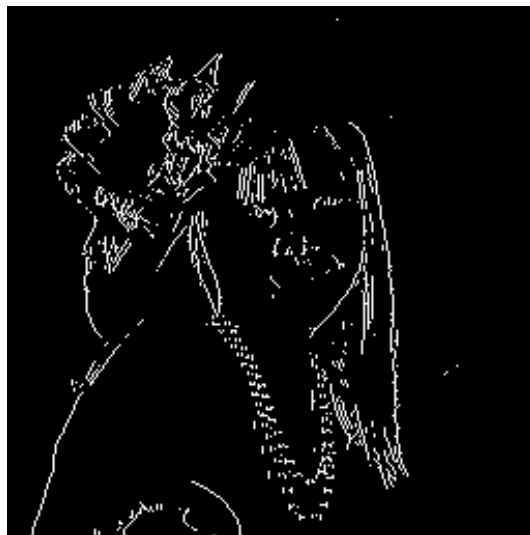
de manera que el mòdul del gradient de la següent forma

$$|\nabla I| = \sqrt{I_x^2 + I_y^2} = \sqrt{[(z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)]^2 + [(z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)]^2}$$

Així doncs, donat un píxel (x, y) direm que és un píxel de contorn si $|\nabla I| > T$ on T és un umbral donat.



a)



b)



a)



b)

Figura 4: a) Fotografia de la nina. Detecció de contorns emprant l'algoritme de Sobel en la direcció b) vertical, c) horitzontal i d) ambdues.

La Figura 4 mostra el resultat, emprant sols la primera màscara, i.e. fent $|\nabla G| \simeq |G_x| > T$, a continuació sols la segona, i.e. fent $|\nabla G| \simeq |G_y| > T$, i el darrer cas la combinació d'ambdós que comentàvem, i.e. fent $|\nabla G| > T$ en comparació amb la imatge original de la nina.

En la primera, obtenim sobretot contorns verticals, ja que amb la màscara emprada, només ens fixam en tires horitzontals a l'hora de fer el gradient, mentre que en la segona, obtenim sobretot els detalls horitzontals, ja que sols tenim en compte tires verticals pel càlcul del gradient. Així doncs, en la primera podem observar tots els detalls del pel o l'esquena de la nina (corbes verticals) que no apareixen en la segona mentre que en la segona veim, per exemple, el detall del capell, els llavis, la barra de darrera que no apreciàvem en l'anterior. Així doncs, en la tercera d'elles, podem observar detalls tant verticals com horitzontals, que, tot i que no ens donen un nivell de detall tan precís com cadascun per separat (hi ha detalls com les celles que si que es veien en la segona imatge, ara no s'aprecien), si que ens donen una informació global més acurada que no pas cadascun d'elles per separat.

2.2. Prewitt Edge Detection

Seguint la notació de la matriu 3×3 anterior, una altra forma de discretitzar les derivades direccionals, I_x i I_y de la funció són les dues màscares següents:

-1	-1	-1
0	0	0
1	1	1

-1	0	1
-1	0	1
-1	0	1

Prewitt

$$G_x = \frac{(z_7 + z_8 + z_9) - (z_1 + z_2 + z_3)}{(z_1 + z_2 + z_3)}$$

$$G_y = \frac{(z_3 + z_6 + z_9) - (z_1 + z_4 + z_7)}{(z_1 + z_4 + z_7)}$$

Figura 5: Màscares corresponent a les derivades direccionals I_x i I_y del mètode de Prewitt.

de manera que el mòdul del gradient de la següent forma

$$|\nabla I| = \sqrt{I_x^2 + I_y^2} = \sqrt{[(z_7 + z_8 + z_9) - (z_1 + z_2 + z_3)]^2 + [(z_3 + z_6 + z_9) - (z_1 + z_4 + z_7)]^2}$$

Així doncs, donat un píxel (x, y) direm que és un píxel de contorn si $|\nabla I| > T$ on T és un umbral donat. La Figura 6 mostra el resultat, emprant sols la primera màscara, sols la segona, i el darrer cas la combinació d'ambdós.



a)



b)



c)



d)

Figura 6: a) Fotografia de la nina. Detecció de contorns emprant l'algoritme de Prewitt en la direcció b) vertical, c) horitzontal i d) ambdues.

2.3. Roberts Edge Detection

Seguint la notació de la matriu 3×3 anterior, una altra forma de discretitzar les derivades direccionals, I_x i I_y de la funció és amb màscares 2×2 corresponents a la meitat inferior dreta de la matriu inicial:

$$\begin{array}{cc}
 \begin{array}{|c|c|} \hline -1 & 0 \\ \hline 0 & 1 \\ \hline \end{array} & \begin{array}{|c|c|} \hline 0 & -1 \\ \hline 1 & 0 \\ \hline \end{array} \\
 G_x = z_9 - z_5 & G_y = z_8 - z_6
 \end{array}
 \quad \text{Roberts}$$

Figura 7: Màscares corresponent a les derivades direccionals I_x i I_y del mètode de Prewitt.

de manera que el mòdul del gradient de la següent forma

$$|\nabla I| = \sqrt{I_x^2 + I_y^2} = \sqrt{[z_9 - z_5]^2 + [z_8 - z_6]^2}$$

Així doncs, donat un píxel (x, y) direm que és un píxel de contorn si $|\nabla I| > T$ on T és un umbral donat. La Figura 8 mostra el resultat, emprant sols la primera màscara, sols la segona, i el darrer cas la combinació d'ambdós.

2.4. Resultats

Ara que ja hem pogut observar les tres per separat, si les comparem, notarem que aquesta darrera tècnica es queda amb els detalls més grans, no centrant-se tant en els detalls. D'altra banda, tant Sobel, com Prewitt són dues tècniques molt semblants ja que el nivell de detall d'ambdues és molt semblant: la barra trasseran, els cabells del front de la nina, els detalls de la flor del capell, el collar de perles,... més o manco són iguals de respectats per un com per l'altre model.



a)



b)



c)



d)

Figura 8: a) Fotografia de la nina. Detecció de contorns emprant l'algoritme de Roberts en la direcció b) vertical, c) horitzontal i d) ambdues.

3. Altres tècniques

En aquest segon apartat, tractarem dues tècniques basades en la informació que ens proporciona la segona derivada d'una funció: a través el laplacà o bé cercant els zeros de la segona derivada; i d'altre banda, una tècnica basada en el càlcul dels gradients direccionals majors, la tècnica que, fins ara és considerada com la que dóna els millors resultats visuals: el mètode de Canny.

3.1. Laplacà d'una Gaussiana

El laplacà d'una funció f de classe C^1 vé donat per

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

Aquest es pot discretitzar de dues formes segons les dues mascare següents

0	-1	0	-1	-1	-1
-1	4	-1	-1	8	-1
0	-1	0	-1	-1	-1

la primera, més classica, la segona, aprofitant la informació de les diagonals. Ara bé, el laplacà no es pot aplicar directament a la imatge sense més per diversos motius: les segones derivades solen incrementar el renou, com veiem en la Figura 2, es produeixen dobles contorns i és incapaç de detectar la direcció del contorn.

Per això, una primera tècnica per procedir a detectar els zero de la segona derivada, evitant tots aquests problemes, emprarem una difuminació prèvia al càlcul dels zeros del laplacà. Considerem la funció Gaussiana

$$h(r) = -e^{-\frac{r^2}{2\sigma^2}}$$

on $r^2 = x^2 + y^2$ i σ és la desviació típica. Aquesta és una funció la qual, si es convoluciona amb la imatge inicial la difumina. El grau de difuminació ve determinat segons el valor de σ . El seu laplacà (segona derivada respecte de r) es

$$\Delta h(r) = -\frac{r^2 - \sigma^2}{\sigma^4} e^{-\frac{r^2}{2\sigma^2}}$$

Com que la segona derivada és una operació lineal, convolucionar una imatge amb $\Delta h(r)$ és el mateix que convolucionar la imatge amb la gaussiana primer i després computar el laplacà del resultat. En el nostre cas, en termes d'eficiència computacional, empram el primer mètode.

Així, un cop aplicada la màscara resultant al laplacà de la gaussiana, obviem els valors, el laplacà dels quals està per sota d'un valor donat T i ens quedam els que estan per sobre, que són precisament, els que conformaran el contorn de la nostra imatge.



Figura 9: Algoritme del laplacà d'una gaussiana amb $\sigma = 2$.

En la Figura 9, podem observar l'exemple de la nina, amb una desviació típica $\sigma = 2$, i un valor llindar $T = 0.2$. Els resultats no deixen res que desitjar a la resta de mètodes presentats, centrant-se en trets generals de la figura i ignorant els detalls menors de la mateixa.

3.2. Zero-Crossing

La tècnica és similar a l'anterior, sols que en aquest cas, no hi ha convolució prèvia amb cap filtre que elimini el renou, amb la qual cosa el resultat serà més groller que en el cas anterior. La tècnica es basa en cercar els zeros del laplacià directament. El resultat el podem observar en la Figura 10



Figura 10: Algoritme de Zero-Crossing

Com es pot apreciar, hi ha moltíssims més detalls, els quals poden no interessar, sobretot si aquests detalls no formassin part de la nostra figura original, si no que fos renou addicional que embruta la nostra imatge.

3.3. Mètode de Canny

El mètode de Canny (1986) és un dels detectors de contorns més potent que existeixen fins a la data. El seu funcionament és el següent:

- La imatge és emborronada emprant un filtre Gaussià amb una desviació típica donada, σ per tal de reduir el renou.
- Per a cada punt $(x, y) \in \Omega$, es calcula el mòdul del gradient local $\Delta I(x, y) = \sqrt{I_x^2 + I_y^2}$ així com la seva direcció $\alpha(x, y) = \tan^{-1}(I_y/I_x)$, on les components I_x i I_y es poden calcular amb qualssevol de les tècniques de discretització vistes en els primers tipus d'algoritmes. Aquest angle és arrodonit al valor més proper a 0, 45, 90, 135, 180, 225, 270 o 315. A continuació, per a cada punt es calcula quina és la variació del nivell de gris en la direcció del gradient computat anteriorment. Es defineix, sota aquest contexte, un **punt de contorn** com aquell que té la major variació en la direcció del gradient.
- Donats dos umbrals $0 < T_1 < T_2$, es marquen amb valor zero aquells píxels amb la variació inferior a T_1 , no són punts de contorn; els que la variació està entre T_1 i T_2 s'anomenen punts de contorn dèbil (marcats amb un 1), i els que tenen una variació superior a T_2 s'anomenen punts de contorn fort (marcats amb un 2).
- Finalment, un algoritme d'unió de punts parteix d'un contorn format pels punts de contorn fort, i incorpora al contornaquells punts de contorn dèbil que estan connectats en 8-connectivitat a ells.

El resultat es pot observar en la Figura 11, on hem fet distints experiments prenent diferents umbrals de tall i diversos valors de σ . Com es pot observar, jugant una mica amb els paràmetres podem aconseguir gairebé tantes corbes de contorn com volguem, des de unes poques, fins a un profund detall de la flor del capell, les ones del pel de la nina o cadascuna de les perles del seu collar.



Figura 11: Detecció de contorns emprant l'algoritme de Canny amb paràmetres $[T_1, T_2]$ i σ donats per: a) $[0.1, 0.4], \sigma = 2$, b) $[0.001, 0.1], \sigma = 2.5$, c) $[0.0375, 0.09375], \sigma = 2.5$ i d) $[0.0375, 0.09375], \sigma = 2$.

4. Més Resultats

En aquesta secció veurem quatre exemples amb contorns de més clars a més confusos, i proves que hem fet amb tots els algoritmes que s'han explicat al llarg del document. Cada joc de prova consta de vuit fotografies ordenades de la següent forma:

1	2
3	4
5	6
7	8

1. Imatge original.
2. Sobel Edge Detection en dues direccions (T adaptatiu).
3. Prewitt Edge Detection en dues direccions (T adaptatiu).
4. Roberts Edge Detection en dues direccions (T adaptatiu).
5. Laplaciana d'una Gaussiana amb $\sigma = 2$, $T = 0.01$.
6. Laplaciana d'una Gaussiana amb $\sigma = 2$, $T = 0.001$.
7. Zero Crossing.
8. Canny Edge Detection (T adaptatiu).

Aquells algoritmes que empren un T adaptatiu, han emprat el valor que, estadísticament, el propi algoritme detecta com 'el que millor funcionarà' basant-se en un anàlisi a gran escala de la distribució de colors en la imatge.

Com a trets generals, destacar que els quatre primers exemples (2 a 5) ens donen contorns molt precisos, sense detalls interns (que segons per quines coses, com per exemple la comparació d'imatges, la detecció de certs objectes per part d'un robot,... pot ser del tot inviable). Els tres darrers (6 a 8) presenten ja bastant més contorns (sobretot Canny i el laplaciana de la Gaussiana amb $T = 10^{-3}$), fins el punt que es fa difícil en alguns casos entendre l'esbós de la imatge persé, sobretot en el cas de la fruitera i les verdures.

