

# DETECTOR DE CONTORNS BASAT EN EL DOMINI TRANSFORMAT

PERE MARTI-PUIG

Grup de Codisseny Hardware-Software, Departament de Tecnologies Digitals i de la Informació ,  
Universitat de Vic, Carrer de la Sagrada Família, 7- 08500 Vic, e-mail: pere.marti@uvic.cat

Data de recepció: 25/09/08

Data de publicació: 20/01/09

---

## RESUM

En aquest document es presenta un detector de contorns d'imatges basat en el domini transformat. A partir de la interpretació de la transformada de Fourier de la imatge i la seva formulació matricial en termes dels diferents modes, es realitza una selecció de les components passa baixes a partir de les quals es reconstrueix la component de baixa freqüència que es resta de la imatge original per tal d'obtenir el detector. Aquest detector de contorns no és esbiaixat. L'algorisme pot ser aplicat utilitzant diferents mides del bloc de processament, que pot anar de la imatge sencera a blocs de reduïdes dimensions: 36x36, 16x16 o 8x8, per fer un seguiment de les propietats locals de la imatge quan aquesta és presenta característiques espacials poc uniformes.

## ABSTRACT

En este documento se presenta un detector de contornos de imágenes basado en el dominio transformado. A partir de la interpretación de la transformada de Fourier de la imagen y su formulación matricial en términos de los diferentes modos, se realiza una selección de las componentes paso-bajas a partir de las cuales se reconstruye la componente de baja frecuencia que se restará a la imagen original para obtener el detector. Este detector de contornos no es sesgado. El algoritmo puede ser aplicado utilizando diferentes medidas del bloque de procesado, que puede ir de la imagen entera a bloques de reducidas dimensiones: 36x36, 16x16 o 8x8, que permiten hacer un seguimiento de las propiedades locales de la imagen cuando ésta presenta características sectoriales muy diversas.

## RESUMEN

In this document an image contour detector based on the transformed domain is presented. Following the interpretation of the image Fourier transform and its matrix formulation in terms of its different modes, we select the base-band ones from which we reconstruct the low frequency image component. This component is subtracted to the original image in order to obtain the contours. This contour detector is not biased. The algorithm can be implemented using different block processing sizes, which can range from the entire image to blocks of smaller dimensions: 36x36, 16x16 or 8x8. Small blocks improve the contour detector performance when the local properties of the image are not uniform.

## 1. INTRODUCCIÓ

Els canvis o discontinuïtats en la lluminositat d'una imatge tenen especial importància perquè sovint presenten els límits dels objectes que hi representen. Aquestes discontinuïtats locals dins una imatge de luminància s'anomenen contorns de luminància. La detecció de contorns és una eina bàsica en el processament d'imatge i existeixen nombrosos algorismes per a la seva realització. Molts d'aquests algorismes es basen en el desplaçament píxel a píxel d'unes plantilles que marquen les pautes de les operacions que es realitzen amb els píxels veïns. Aquestes plantilles poden ser representades a través de matrius. Els algorismes computacionalment menys costosos utilitzen plantilles de mida 3X3 tot i que fàcilment poden ser adaptats per a dimensions majors. A través d'aquestes plantilles es detecten els gradients, derivades de primer ordre, primer en la direcció de les files i posteriorment en la direcció de les columnes. Això és, caldrà realitzar un filtratge de la imatge amb una plantilla per calcular els gradients en la direcció de les files i posteriorment, amb una altra plantilla es calcula el gradient en la direcció de les columnes. Dins d'aquesta modalitat hi trobem els algorismes basats en la diferència de píxels i en la diferència de píxels separats, així com els algorismes de Roberts [1][3], de Prewitt [1][2], de Sobel [1] i de Frei-Chen [1][4]. Altres plantilles, que incorporen elements de filtratge gaussià, han estat dissenyades per Argyle [5], Macleod [6]. A [7], Canny's ha proposat un mètode heurístic per desenvolupar detectors de contorns. Un segon conjunt d'algorismes basa el disseny dels coeficients de les plantilles a partir de derivades de segon ordre. Entre aquests mètodes en destaquem els basats en el càlcul de la Laplaciana [8] i el de la detecció del seu pas per zero [9] o els basats directament en el càlcul en les derivades de segon ordre [10][11]. També es poden realitzar detectors de contorns a partir de la realització de filtres de resposta impulsional finita o partir de tècniques de morfologia [1].

Un dels problemes que presenten els algorismes de detecció de contorns és que, al final de l'algorisme, cal establir un llindar a partir del qual es considera que el píxel en qüestió pertany o no al contorn de la imatge [1]. L'establiment del valor òptim d'aquest llindar depèn també de la imatge analitzada i, en tots els casos, tenim una probabilitat d'error no nul·la, la qual cosa implica classificar malament el punt en relació a la seva pertinença a un contorn de la imatge. Es per aquest motiu que el problema de la detecció de contorns apareix com un tema obert susceptible a múltiples millores i noves interpretacions.

En aquest document es presenta un detector de contorns d'imatges basat en el domini transformat. A partir de la interpretació de la transformada de Fourier de la imatge i la seva formulació matricial en termes dels diferents modes, es realitza una reconstrucció de les components passa baixes de la imatge que es restarà de la imatge original per tal d'obtenir el detector.

La transformada de Fourier bidimensional té un algorisme ràpid que es coneix com la 2D-FFT (2-Dimensional Fast Fourier Transform), aplicada àmpliament en el processament d'imatges és una transformada ortogonal que presenta la propietat de poder ser calculada en dos passos a partir del càlcul de les transformades ràpides de Fourier unidimensional o FFT [12].

L'algorisme FFT fou descobert originàriament per Gauss [13], i redescobert per Cooley i Tukey [14] a la dècada del 1960. S'usa àmpliament en molts camps de l'enginyeria i en conseqüència s'ha desenvolupat una gran varietat d'algorismes d'ençà de la dècada dels 60 amb una extensa bibliografia existent. A grans trets, hi ha algorismes anomenats de ràdix superior [15][16], de ràdix mixt [17], de factors primers [18], de Winograd [19], de split-ràdix [20] [21], amb geometries idèntiques entre etapes [24], amb formulació recursiva [22], algorismes que barregen estratègies de delmació amb temps amb delmació en freqüència [23], entre molts d'altres. Les representacions matricials de la FFT es poden trobar a [26], [27], [28], [29] i noves tendències al camp les transformades discretes estan compilades a [30]. Tot i que a [25] es podem obtenir una

visió interessant sobre l'estat de art, volem citar especialment la referència [31] ja que constitueix un tractat modern i actualitzat de l'estat de l'art de l'algorísmica de la FFT, amb l'avantatge de proporcionar una comparativa molt completa sobre nombrosos algorismes que han anat sorgint al llarg de tots aquests anys reformulats usant una notació matemàtica comuna.

## 2. Notació utilitzada. Definició de la transformada de Fourier bidimensional.

A continuació presentem la notació utilitzada. Referenciem una matriu amb una lletra majúscula en negreta, les seves dimensions s'indiquen per mitjà d'un subíndex:  $\mathbf{X}_{M,N}$  representa una matriu de  $M$  files i  $N$  columnes amb elements  $x_{m,n}$  ( $m=0:M-1$ ,  $n=0:N-1$ ). Tots els vectors seran tipus columna i es referenciaran amb una lletra minúscula també en negreta, amb un subíndex per indicar el nombre d'elements; ex.  $\mathbf{x}_M = \{x_m\}$  representa un vector columna de  $M$  elements  $x_m$  ( $m=0:M-1$ ). En aquest cas el subíndex indica ordre o posició dins del vector. Per indicar la mida de les matrius no quadrades en relació a les operacions transposició o a l'operador hermític establim el següent conveni per a representar les mides:

$$\mathbf{X}_{M,N} = (\mathbf{X}_{M,N})^T = \mathbf{X}_{N,M}^T \quad (1)$$

Es defineix la matriu de la transformada de Fourier de dimensions  $M \times N$  en termes dels seus elements a partir de l'expressió:

$$\mathbf{F}_{M,N} = \left\{ e^{-j\frac{2\pi}{N}mn} \right\} \quad m = 0 : M - 1, n = 0 : N - 1. \quad (2)$$

Aquesta mateixa matriu,  $\mathbf{F}_{M,N}$ , també pot ser representada en termes dels seus  $N$  vectors columna, o modes, segons:

$$\mathbf{F}_{M,N} = [\mathbf{v}_0 \quad \mathbf{v}_1 \quad \dots \quad \mathbf{v}_{N-1}] \quad (3)$$

On el mode genèric  $\mathbf{v}_u$ , on  $u$  pot ser qualsevol valor entre 0 i  $N-1$ , es pot escriure en funció dels seus  $M$  elements com:

$$\mathbf{v}_u = \left\{ e^{-j\frac{2\pi}{N}mu} \right\} \quad m = 0 : M - 1. \quad (4)$$

Notem que la matriu de Fourier  $\mathbf{F}_{M,N}$  prèviament definida no té cap restricció dimensional essent  $M$  i  $N$  valors qualsevol, amb la qual cosa  $\mathbf{F}_{M,N}$  és en general una matriu rectangular. En aquest punt, per tal de simplificar la notació i per a major claredat, considerarem que la matriu és quadrada i per tant  $M=N$ . Els resultats sempre es poden transportar per a valors de  $N$  diferents a  $M$ . Per simplicitat, en endavant, considerem la matriu quadrada  $\mathbf{F}_{M,M}$ .

És fàcil observar com  $\mathbf{F}_{M,M}$  presenta la propietat de ser simètrica, això és, el resultat de canviar files per columnes deixa inalterada la matriu original.

$$\mathbf{F}_{M,M} = (\mathbf{F}_{M,M})^T \quad (5)$$

A (5) el superíndex  $T$  significa transposat. Aquesta mateixa propietat en termes dels modes

quedarà de la forma:

$$[\mathbf{v}_0 \quad \mathbf{v}_1 \quad \dots \quad \mathbf{v}_{M-1}] = \begin{bmatrix} \mathbf{v}_0^T \\ \mathbf{v}_1^T \\ \dots \\ \mathbf{v}_{M-1}^T \end{bmatrix}. \quad (6)$$

La transformada discreta de Fourier d'un vector columna  $\mathbf{x}_M$ , serà un altre vector columna  $\mathbf{y}_M$  que es pot expressar a partir del producte  $\mathbf{F}_{M,M} \mathbf{x}_M$ . En aquest punt és interessant la interpretació del vector transformat com una combinació lineal dels modes de  $\mathbf{F}_{M,M}$  segons:

$$\mathbf{y}_M = \mathbf{F}_{M,M} \mathbf{x}_M = \mathbf{v}_0 x_0 + \mathbf{v}_1 x_1 + \dots + \mathbf{v}_{M-1} x_{M-1} \quad (7)$$

Si organitzem els elements de  $\mathbf{x}_M$  en una fila, igualment interessant resulta la interpretació següent:

$$\mathbf{y}_M^T = \mathbf{x}_M^T \mathbf{F}_{M,M} \quad (8)$$

Si considerem la representació d'una imatge com una matriu, on cada punt representa un nivell de gris, el càlcul de la seva transformada, atesa la propietat de separabilitat [12], es pot portar a terme partint de la realització d'una FFT a cada una de les files de la imatge i a continuació una altra FFT sobre cada una de les columnes. Llavors, si  $\mathbf{X}_{M,M}$  és una imatge quadrada, organitzada en una matriu, la transformada de Fourier de la imatge, quedarà recollida en la matriu  $\mathbf{Y}_{M,M}$ , segons l'expressió:

$$\mathbf{Y}_{M,M} = (\mathbf{F}_{M,M})^T \mathbf{X}_{M,M} \mathbf{F}_{M,M} \quad (9)$$

Fem notar en aquest punt que  $\mathbf{Y}_{M,M}$  té valors complexos. Atesa la propietat (5), l'expressió (9) es simplifica de la forma següent:

$$\mathbf{Y}_{M,M} = \mathbf{F}_{M,M} \mathbf{X}_{M,M} \mathbf{F}_{M,M} \quad (10)$$

### 3. Desenvolupament de l'algorisme.

Partim de la interpretació vista a l'apartat anterior segons la qual una transformada discreta de Fourier pot ser vista com una combinació lineal dels modes que formen la matriu, això és, dels seus vectors columna. Si tenim en compte que cada un d'aquests modes representa una freqüència ordenada de menor a major pulsació segons s'incrementa l'índex de les columnes, podem interpretar el vector transformat com una combinació lineal de les freqüències. Partint d'aquesta interpretació, doncs, podem realitzar una selecció dels modes que volem utilitzar per a la representació de la transformada.

A grans trets, els contorns representen variacions brusques del contingut del senyal original, això és, informació d'alta freqüència o altrament dit, són representables a través de modes alts, mentre que les zones de la imatge que no presenten discontinuïtats porten un contingut de baixa freqüència i els modes amb índex baix seran més rellevants per a la seva representació. Seguint aquesta representació podem obtenir la transformada parcial del senyal només en un conjunt de modes, una selecció de la totalitat de modes presents en la transformada completa. Aquesta

selecció es pot realitzar a partir de criteris freqüencials.

A continuació seleccionem només els  $U$  primers modes de la matriu, de manera que obtenim una matriu rectangular de dimensions  $M \times U$ ,  $U < M$ , que permetria la descomposició del senyal original en les freqüències més baixes.

Sigui la matriu  $\tilde{\mathbf{F}}_{M,U}$  :

$$\tilde{\mathbf{F}}_{M,U} = \left\{ e^{-j\frac{2\pi}{N}mn} \right\} \quad m = 0 : M - 1, n = 0 : U - 1. \quad (11)$$

Llavors la descomposició de la imatge en els modes seleccionats prendrà la forma:

$$\tilde{\mathbf{Y}}_{U,U} = (\tilde{\mathbf{F}}_{M,U})^T \mathbf{X}_{M,M} \tilde{\mathbf{F}}_{M,U} \quad (12)$$

Notem que la matriu resultat de la transformada presentarà dimensions inferiors, en aquest cas  $U \times U$ . Notem també la relació:

$$(\tilde{\mathbf{F}}_{M,U})^H \tilde{\mathbf{F}}_{M,U} = \tilde{\mathbf{F}}_{U,M}^H \tilde{\mathbf{F}}_{M,U} = M \mathbf{I}_{U,U} \quad (13)$$

On el superíndex  $H$  representa l'operador hermític (complexe-conjugat) i la matriu  $\mathbf{I}_{U,U}$  és la matriu identitat de mida  $U \times U$ .

La reconstrucció de les components passa-baixes, és a dir, de baix contingut freqüencial de la imatge s'obtidran de l'operació:

$$\mathbf{B}_{M,M} = \alpha (\mathbf{F}_{M,M})^H \begin{bmatrix} \tilde{\mathbf{Y}}_{U,U} & \mathbf{O}_{U,M-U} \\ \mathbf{O}_{M-U,U} & \mathbf{O}_{M-U,M-U} \end{bmatrix} (\mathbf{F}_{M,M}) \quad (14)$$

On el paràmetre  $\alpha$  presenta el valor  $M^2$  i les matrius  $\mathbf{O}_{U,M-U}$ ,  $\mathbf{O}_{M-U,U}$  i  $\mathbf{O}_{M-U,M-U}$  són matrius formades únicament per zeros.

Atès que moltes operacions són productes per zero, podem simplificar l'equació (14). Després d'un estudi de l'expressió (14) en termes de les matrius que hi intervenen en termes de diferents sub-blocs, s'arriba a la següent expressió que es pot reescriure en termes de la matriu  $\tilde{\mathbf{F}}_{M,U}$ .

$$\mathbf{B}_{M,M} = \alpha \tilde{\mathbf{F}}_{M,U}^* \tilde{\mathbf{F}}_{U,M}^T \mathbf{X}_{M,M} \tilde{\mathbf{F}}_{M,U} \tilde{\mathbf{F}}_{U,M}^H \quad (15)$$

On el símbol  $*$  significa conjugat. La imatge  $\mathbf{B}_{M,M}$  conté les baixes freqüències de la imatge original  $\mathbf{X}_{M,M}$ . Obtindrem la imatge dels contorns de la següent diferència:

$$\mathbf{C}_{M,M} = |\mathbf{X}_{M,M} - \mathbf{B}_{M,M}| \quad (16)$$

On l'operador  $|\cdot|$  significa valor absolut. Això és:

$$\mathbf{C}_{M,M} = \left| \mathbf{X}_{M,M} - \alpha \tilde{\mathbf{F}}_{M,U}^* \tilde{\mathbf{F}}_{U,M}^T \mathbf{X}_{M,M} \tilde{\mathbf{F}}_{M,U} \tilde{\mathbf{F}}_{U,M}^H \right| \quad (17)$$

On prenem el valor absolut per tal d'evitar valors negatius. Arribats en aquest punt, per tal de definir l'extracció de contorns com una operació matricial definim les matrius  $\mathbf{U}_{M,M}$  i  $\mathbf{V}_{M,M}$ :

$$\mathbf{U}_{M,M} = \tilde{\mathbf{F}}_{M,U} \tilde{\mathbf{F}}_{U,M}^H \quad (18)$$

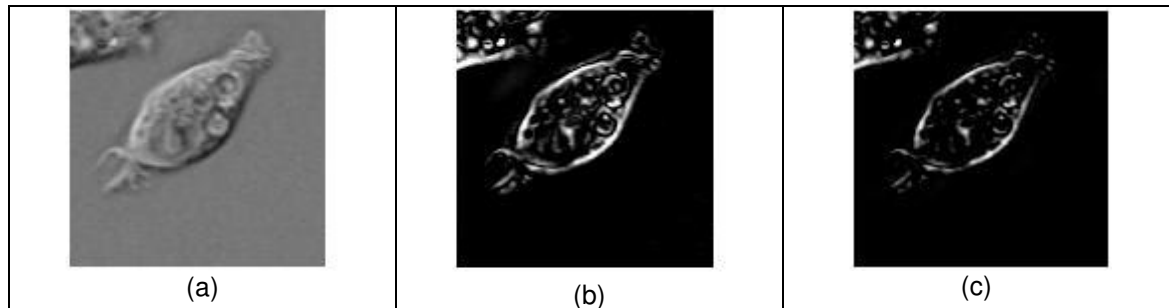
$$\mathbf{V}_{M,M} = \alpha \tilde{\mathbf{F}}_{M,U} \tilde{\mathbf{F}}_{U,M}^H \quad (19)$$

Llavors podem escriure la matriu de contorns  $\mathbf{C}_{M,M}$  com:

$$\mathbf{C}_{M,M} = \left| \mathbf{X}_{M,M} - \mathbf{V}_{M,M}^H \mathbf{X}_{M,M} \mathbf{U}_{M,M} \right| \quad (20)$$

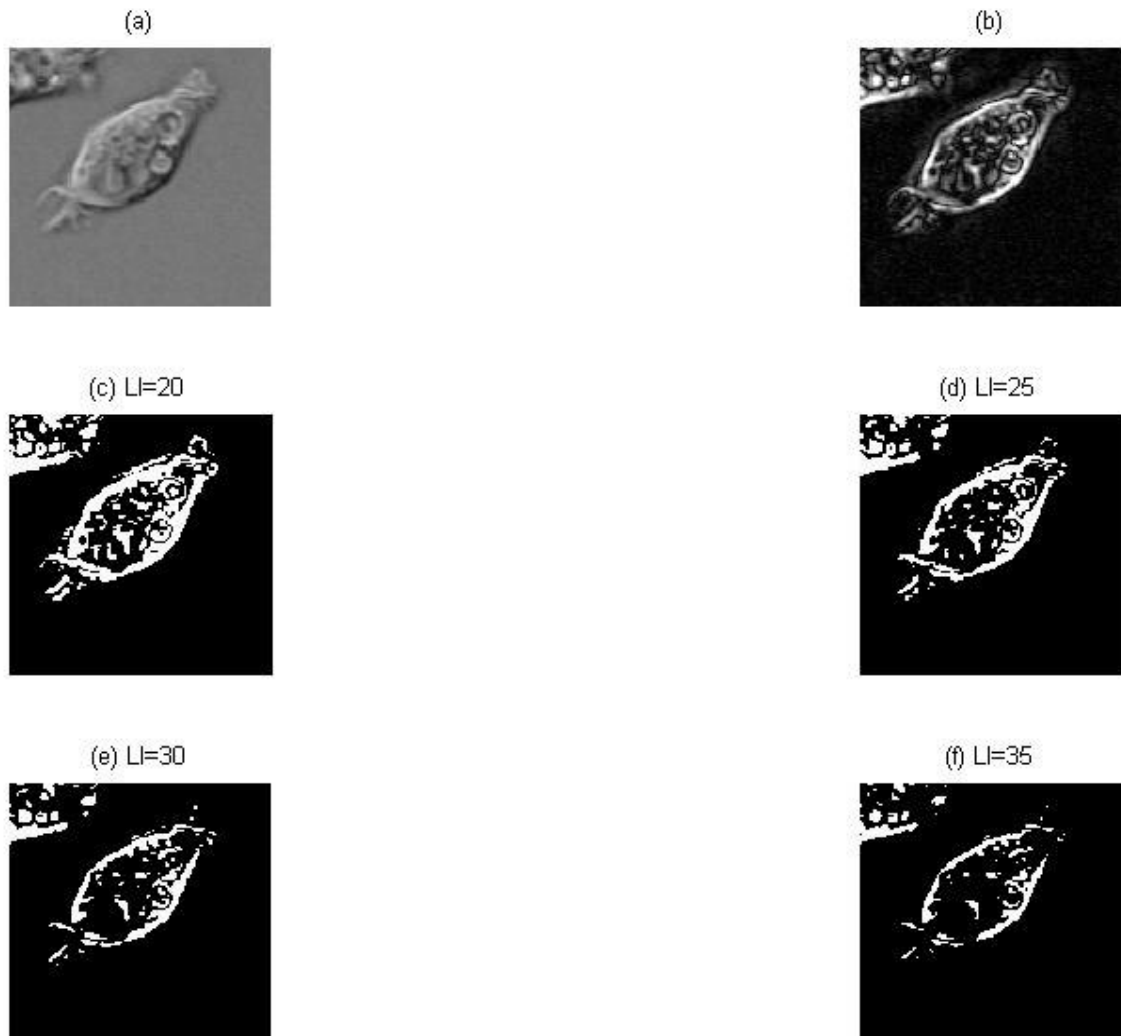
#### 4. Presentació de resultats

A continuació mostrem un conjunt de resultats. A la figura 1; a la imatge 1.a es representa un bacteri. A 1.b s'han extret els contorns a partir de l'algorisme exposat amb els paràmetres  $N=265$  (mida del bloc de processament) i  $U=4$  (modes utilitzats per extreure la component passa-baixes). A 1.c s'ha aplicat novament l'algorisme, amb els mateixos paràmetres, sobre la imatge 1.b amb un efecte de reducció de l'amplada dels contorns. Sovint a les imatges que contenen els contorns s'hi aplica un procés de binarització per tal de simplificar-les. Aquest procés consisteix en determinar un valor per sobre del qual els valors dels píxels de les imatges agafaran el valor màxim (color blanc) i per sota del qual agafaran el valor mínim (color negre).



**Figura 1** Detecció de contorns amb l'algorisme proposat. A, 1.a, es representa la imatge original, a 1.b, els contorns amb els paràmetres de l'algorisme  $N=256$  i  $U=4$ , i a 1.c, el processament de 1.b amb els mateixos valors de  $N$  i  $U$ .

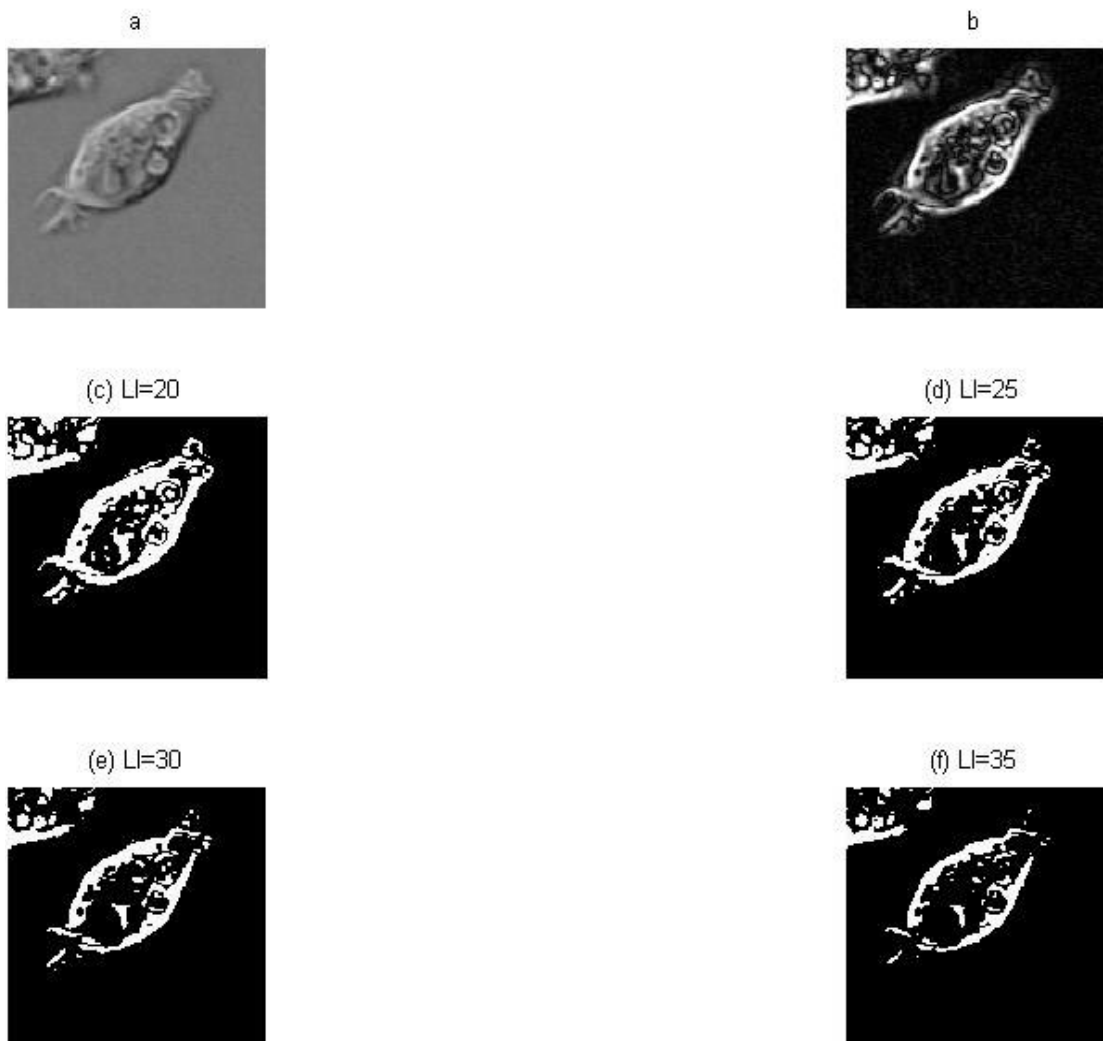
A la figura 2, es mostra l'efecte del llindar de binarització dels contorns sobre la imatge del bacteri. La imatge (a) és la imatge original, la imatge (b) s'obté d'aplicar l'algorisme presentat per  $U=4$ , la resta d'imatges, (c), (d), (e) i (f) són el resultat d'aplicar els llindars 20, 25, 30 i 35 en el procés de binarització. D'aquesta manera il·lustrem la tendència dels algorismes de detecció de contorns respecte d'un valor (llindar), el valor òptim del qual depèn sovint de la imatge.



**Figura 2** Efectes del valor del llindar sobre el procés de binarització de la imatge (b) obtinguda a partir de l'algorisme proposat amb els paràmetres de l'algorisme  $N=256$  i  $U=4$  sobre la imatge (a). S'han provat els valors de llindar (LI) 20, 25, 30 i 35.

A la figura 3, es repeteix l'experiment anterior amb la diferència que l'algorisme proposat s'executa amb el valor de  $U=1$ . Així mateix, (a) és la imatge original, la imatge (b) s'obté d'aplicar l'algorisme del detector de contorns per  $U=1$  i la resta d'imatges, (c), (d), (e) i (f), són el resultat d'aplicar els llindars 20, 25, 30 i 35 necessaris pel procés de binarització.

L'interès d'aquests resultats, que no difereixen en excés als obtinguts amb  $U=4$ , radiquen en què, si només utilitzem el mode de més baixa freqüència, l'algorisme es pot simplificar moltíssim i podem desenvolupar un algorisme ràpid de molt baixa complexitat.



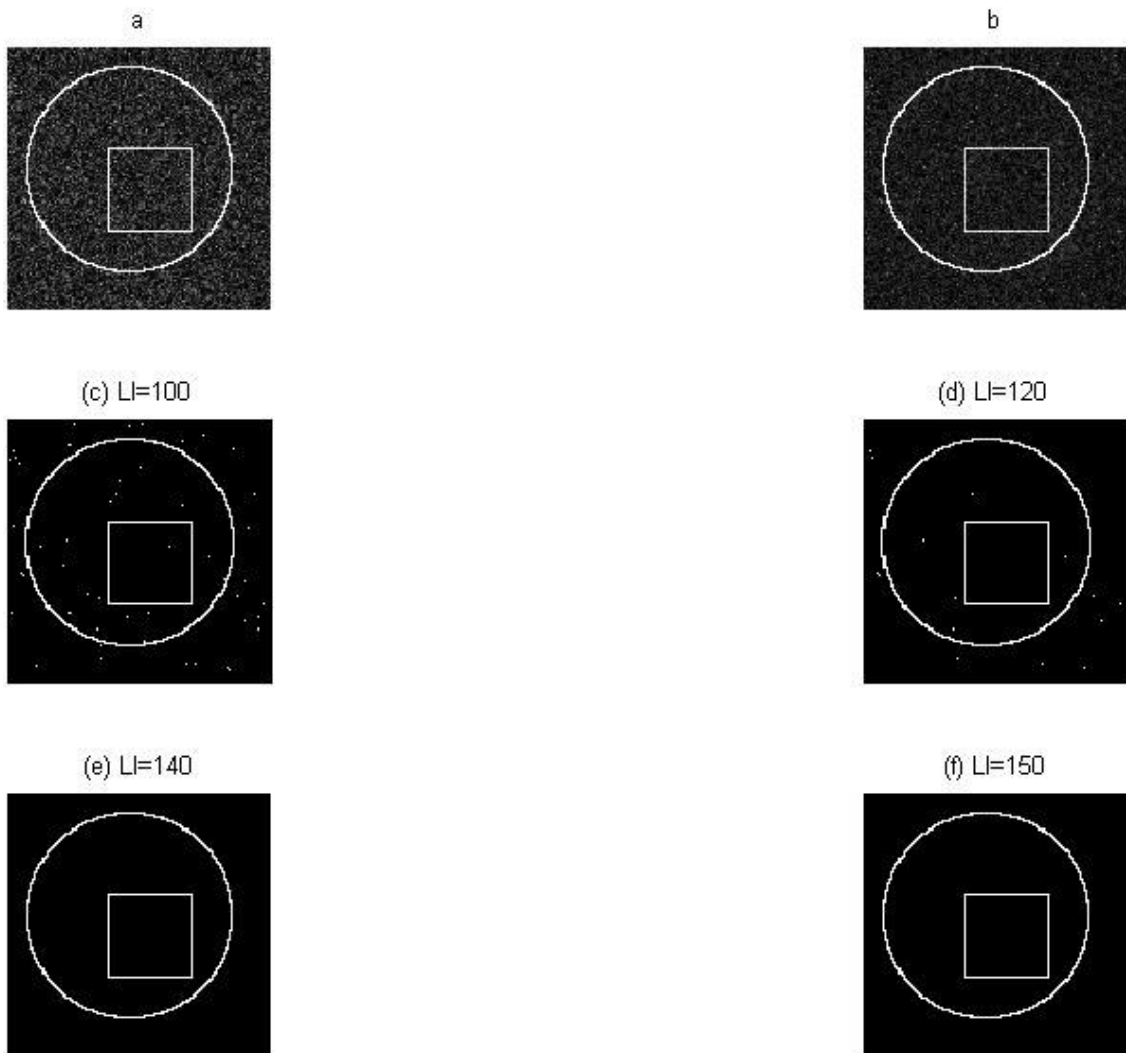
**Figura 3** *Detecció de contorns amb l'algorisme proposat amb paràmetres  $N=256$  i  $U=1$ . En les diferents iteracions del mateix procés es constata una reducció del soroll impulsional i la preservació dels contorns d'interès.*

La següent prova, a la figura 4, s'ha realitzat sobre una imatge sintètica. En aquest cas s'ha representat un cercle i un quadrat i s'hi ha afegit un elevat soroll impulsional, (figura 2.a). Els detectors de contorns basats en plantilles, derivades, gradients o Laplacianes solen ser molt sensibles a aquest tipus de soroll.

La imatge (a) de la figura 4 és la imatge de partida. Es pot observar que, a la imatge (b), resultat d'aplicar l'algorisme proposat per  $N=128$  i  $U=4$ , es preserven els contorns de les figures i es redueix de forma notable el soroll impulsional. La resta de figures, (c), (d), (e) i (f) mostren l'efecte del llindar sobre la discretització binària dels contorns.

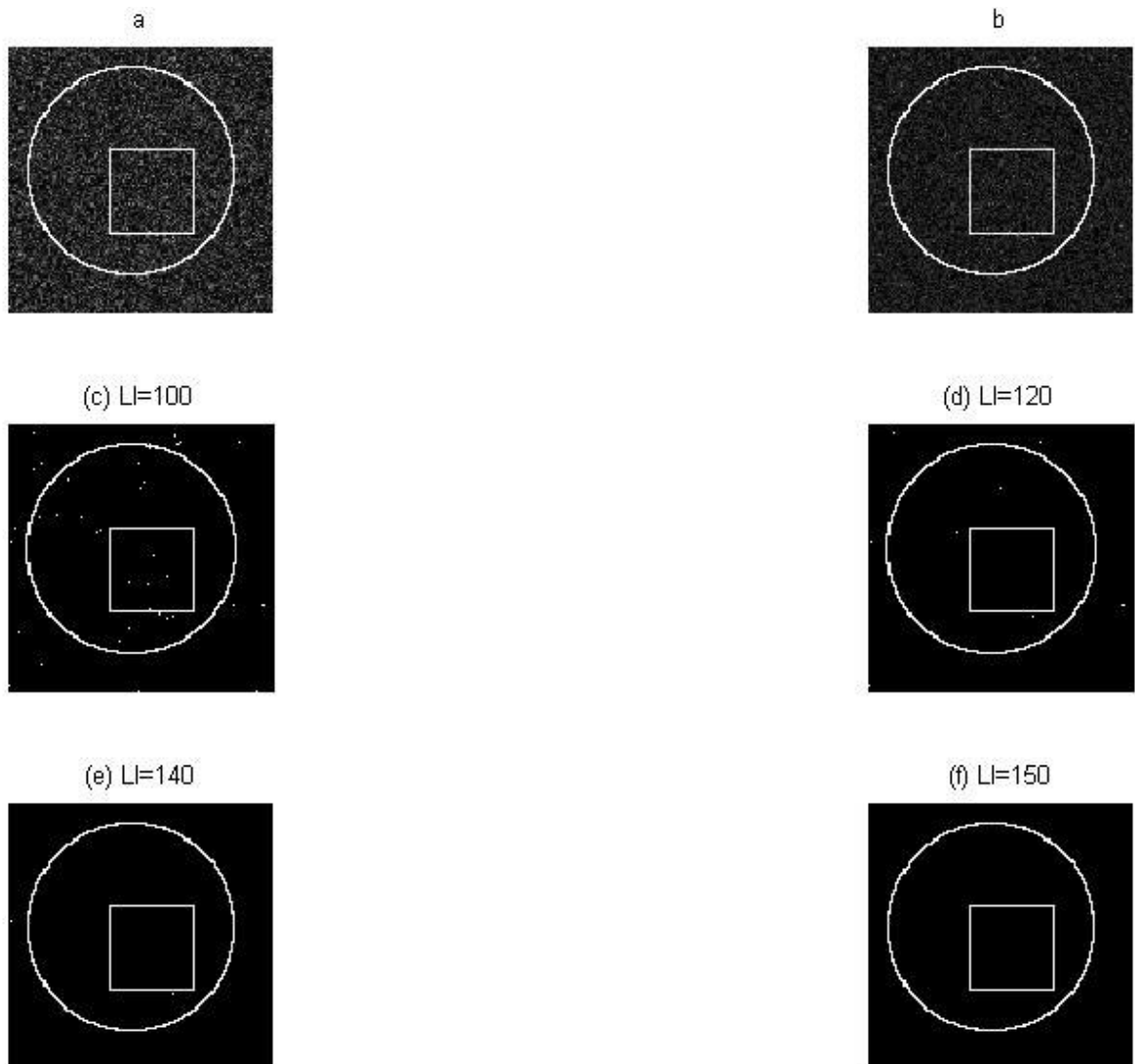
A nivell de comentari volem fer notar que l'aplicació iterativa del l'algorisme proposat sobre la seva sortida produeix un efecte de reducció progressiva del soroll impulsional.





**Figura 4** Efectes del valor del llindar sobre el procés de binarització de la imatge (b) obtinguda a partir de l'algorisme proposat amb els paràmetres de l'algorisme  $N=128$  i  $U=4$  sobre la imatge (a). S'han provat els valors de llindar (LI) 100, 120, 140 i 150.

A la figura 5 es repeteix la prova amb només un mode ( $U=1$ ), a fi de constatar que l'algorisme de baixa complexitat també funciona de manera òptima.



**Figura 5** Efectes del valor del lindingar sobre el procés de binarització de la imatge (b) obtinguda a partir de l'algorisme proposat amb els paràmetres de l'algorisme  $N=128$  i  $U=4$  sobre la imatge (a). S'han provat els valors de lindingar (LI) 100, 120, 140 i 150.

## 5. Conclusions, consideracions sobre el cost computacional i línies de recerca futures.

Primer de tot cal dir que s'ha desenvolupat un detector de contorns a partir de la informació en el domini freqüencial que funciona molt correctament ja per valors de  $U$  petits, la qual cosa equival a dir que es necessiten molt pocs modes per obtenir una aproximació de baixa freqüència del senyal. A l'extrem les components de baixa freqüència a la matriu  $\mathbf{C}_{M,M}$  del senyal original  $\mathbf{X}_{M,M}$  queden les components d'alta freqüència associades als contorns.

Per a l'obtenció de  $\mathbf{C}_{M,M}$ , per valors de  $U$ ,  $U < M$  de qualsevol mida, des del punt de vista del cost computacional resulta molt interessant l'ús de l'algorisme ràpid de la transformada de Fourier o FFT (Fast Fourier Transform). El cost del càlcul directe de la transformada discreta sobre un vector de mida  $N$  és proporcional a  $N^2$ . Amb l'algorisme FFT (1D) el cost es redueix a una quantitat proporcional a  $\text{Mlog}_2 N$ . Per a una fila o columna amb 1024 píxels (o punts) la quantitat d'operacions passa a ser de l'ordre de  $10^6$  a  $10^4$ , és a dir, hi ha una reducció d'un factor 100.

Abans però, per valors de  $U$ ,  $U < M$ , però majors que 1, caldrà que realitzem un seguit de consideracions per tal d'obtenir el màxim d'avantatges en el càlcul dels contorns:

- És important disposar d'un algorisme FFT que adquireixi les dades de forma ordenada i les deixi igualment ordenades. Amb això s'evita l'operació de permutacions bit-reveral que caldria aplicar a la sortida (o opcionalment a l'entrada) de cada fila i de cada columna de la imatge. En aquest punt recordem que la FFT-2D aplicada a una imatge equival a la realització de  $N$  FFTs-1D sobre cada una de les columnes i  $M$  FFTs sobre cada una de les files que componen la imatge a transformar. Un algorisme òptim en aquest sentit seria el desenvolupat a [32].
- Atès que només estem interessats en els primers  $U$  modes de la transformada de Fourier no és necessari el càlcul de tot l'algorisme de la FFT. És més, per una transformada de 1024 punts es calculen els coeficients de 1024 modes. El nombre d'operacions es pot reduir considerablement si tenim en compte que l'algorisme ja proporciona molt bons resultats pels quatre primers modes, això és, per  $U=4$ .
- Pensant en una implementació hardware o programada en un processador de punt fix, es poden simplificar les operacions involucrades en el càlcul dels  $U$  modes necessaris si tenim en compte que les dades a transformar són reals (els píxels tenen un valor enter positiu). L'algorisme FFT, per defecte, contempla dades d'entrada complexes.

Des del punt de vista de la continuació d'aquest treball i l'obtenció de línies de recerca futures, presentem diferents possibilitats interessants per a ser explorades i que enumerem a continuació:

- Una possible recerca en el camp de l'algorísmica consistirà en el desenvolupament de la factorització *sparse* de la matriu  $\tilde{\mathbf{F}}_{M,U}$  de la que sortirà l'algorisme ràpid que permeti l'obtenció dels  $U$  modes desitjats, de forma ordenada, sense haver de calcular tots els coeficients les FFTs-1D. Aquest objectiu sembla abastable, a priori, si  $U$  és potència de 2.
- La segona línia a explorar faria referència al desenvolupament de filtres digitals dissenyats en el domini transformat des del punt de vista d'operacions d'àlgebra matricial, a l'estil de les operacions realitzades en el càlcul de la component passa baixes del senyal  $\mathbf{B}_{M,M}$  a partir de la pre-multiplicació i la post-multiplicació matricial de la matriu original.
- També és interessant estudiar les prestacions de l'algorisme simplificat on  $U=1$  i determinar les condicions que han de complir les imatges pel seu bon funcionament així com establir els seus límits. En aquest sentit seria interessant un estudi del valor de  $U$  necessari en funció de les característiques de la imatge. Recordem que el primer mode de la transformada de Fourier ens dona la component contínua del senyal, que es pot calcular d'un simple promig dels valors de la imatge.
- També és interessant l'adaptació de l'algorisme a un processament de la imatge per blocs. Les avantatges que proporcionaria serien, a nivell de càlcul, una major facilitat en

el càlcul de la transformada i a nivell de resultats una adaptació progressiva a les característiques de la imatge quan aquestes variïn geogràficament.

## REFERÈNCIES

- [1] W. K. Pratt, *Digital Image Processing*, John Wiley & Sons, New York, 1991.
- [2] J. M. S. Prewitt, "Object Enhancement and Extraction", in *Picture Processing and Psychopictorics*, Academic Press, New York, 1970.
- [3] L. G. Roberts, "Machine Perception of Tree-Dimensional Solids", in *Optical and Electro-Optical Information Processing*, MIT Press, Cambridge, MA, 1965.
- [4] W. Frei, C. Chen, "Fast Boundary Detection: A Generalization and a New Algorithm", *IEEE Trans. Computers*, C-26, p.p. 988-998, October 1973.
- [5] E. Argyle, "Techniques for Edge Detection", *Proc. IEEE*, Vol 59, p.p. 285-287, February 1971.
- [6] I. D. G. Macleod, "Commenst on Techniques for Edge Detection", *Proc. IEEE*, Vol 60, p.p. 344, March 1972.
- [7] J. Canny, "A Computational Approach to Edge Detection", *IEEE Trans. Pattern Analysis and Machine Intelligence*, PAMI-8, p.p. 679-698, November 1986.
- [8] D. Marr, E. Hildrith, "Theory of Edge Detection", *Proc. Royal Soc. London*, Vol. 207, p.p. 187-217, 1980.
- [9] A. Huertas, G. Medioni, "Detection of Intensity Changes Using Laplacian-Gaussian Masks", *IEEE Trans. Pattern Analysis and Machine Intelligence PAMI*, Vol 8, p.p. 651-664, September 1986.
- [10] R. M. Haralick, "Digital Step Edges from zero Crossing of Secod Directional Derivatives", *IEEE Trans. Pattern Analysis and Machine Intelligence PAMI*, Vol 6, p.p. 58-68, January 1984.
- [11] M. Hueckel, "An Operator which Locates Edges in Digital Pictures", *JACM*, Vol 18, p.p. 113-125, January 1971.
- [12] A. K. Jain, *Fundamentals of Digital Image Processing*, Prentice Hall, New Jersey, 1989.
- [13] M. T. Heideman, D. H. Johnson, and C. S. Burrus, "Gauss and the History of the FFT", *IEEE Acoustics, Speech, and Signal Processing Magazine*, vol. 1, pp. 14-21, October 1984.
- [14] J. W. Cooley, J. W. Tukey, "An Algorithm for the Machine Calculation of Complex Fourier Series", *Math. of Computations* Vol. 19, p.p. 297-301, April. 1965.
- [15] G. D. Bergland, "A Radix-Eight Fast-Fourier Transform Subroutine for Real-Valued Series", *IEEE Trans. Audio Electroacoust.* vol. 17, no. 2, pp. 138-144, June 1969.
- [16] D. Takahashi, "A Radix-16 FFT Algorithm Suitable for Multiply-Add Instruction based on Goedecker Method", *Intern. Conference on Acoustics, Speech, and Signal Processing, ICASSP-2003*, Page(s):II - 665-8 vol.2 6-10, April 2003.
- [17] R. C. Singleton, "An Algorithm for Computing the Mixed Radix Fast Fourier Transform", *IEEE Trans. Audio Electroacoust.* vol. 1, no. 2, pp. 93-103, June 1969.
- [18] D. Pi. Kolba and T. W. Parks, "A Prime Factor FFT Algorithm Using High-Speed Convolution", *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 25, no. 4, pp. 281-294, August 1977.
- [19] S. Winograd, "On Computing the Discrete Fourier Transform", *Math. Comput.*, vol. 32, no. 141, pp. 175-199, January 1978.
- [20] H. V. Sorensen and C. S. Burrus, "A New Efficient Algorithm for Computing a Few DFT Points", *IEEE Trans. Acoust., Speech, Signal Proc.*, vol. 35, no. 6, pp. 849-863, June 1987.
- [21] D. Takahashi, "An Extended Split-Radix FFT Algorithm", *IEEE Signal Processing Letters*, vol. 8, no. 5, pp. 145-147, May 2001.
- [22] A. R. Varkonyi-Koczy, "A Recursive Fast Fourier Transform Algorithm", *IEEE Trans. Circuits*

- and Systems, II, vol. 42, pp. 614-616, September 1995.
- [23] A. Saidi, "Decimation-in-Time-Frequency FFT Algorithm", Proc. IEEE International Conf. on Acoustics, Speech, and Signal Processing, pp. III: 453-456, April 19-22 1994.
  - [24] M. C. Pease, "An adaptation of the fast Fourier transform for parallel processing". *J. Assoc. Comput.* Vol. 15, p.p. 252-324, April 1968.
  - [25] P. Duhamel and M. Vetterli, "Fast Fourier transforms: A tutorial review and a state of the art", *Signal Process.*, vol. 19, pp. 259-299, 1990.
  - [26] J. A. Glassman, "A generalization of the fast Fourier transform", *IEEE Trans. Comput.*, vol. C-19, pp. 105-116, February 1970.
  - [27] M. Drubin, "Kronecker product factorization of the FFT matrix", *IEEE Trans. Comput.*, vol. C-20, pp. 590-593, May 1971.
  - [28] H. Sloate, "Matrix Representations for Sorting and the Fast Fourier Transform", *IEEE Trans.on Circuits and Systems*, vol., cas-21, No. 1, pp. 109-116 January 1974.
  - [29] J. Granata, M. Conner, R. Tolimieri, "Recursive Fast Algorithms and the Role of the Tensor Product", *IEEE Trans.on Signal Proc.*, vol.40., No. 12, pp. 2921-2930 December 1992.
  - [30] S. Egner, M. Püschel, "Automatic Generation of Fast Discrete Signal Transforms", *IEEE Trans.on Signal Proc.*, vol.49., No. 9, pp. 1992-2002 December 2001.
  - [31] C. Van Loan, *Computational Framworks for the Fast Fourier Transform*, SIAM, Philadelphia, 1992.
  - [32] P. Marti-Puig, "Two families of radix-2 FFT algorithms with ordered input and output data", *IEEE Signal Proc. Lett.* (Acceptat, pendent de publicació)