

# Principi ed applicazioni dell'elaborazione digitale delle immagini

*Elaborare un'immagine significa operare una qualche «trasformazione» sull'immagine stessa con l'intento di produrre l'effetto desiderato. Come abbiamo già avuto modo di vedere, il filtraggio, cioè la convoluzione dell'immagine con un operatore filtro, è una delle tecniche più usate.*

*Finora l'immagine è stata sempre definita come un insieme di intensità luminosa o un valore a questa proporzionale, ordinati su di un array bidimensionale per gli indici di  $x$  e  $y$ . Tuttavia per alcuni dei filtri che comunemente vengono adottati, si preferisce per ragioni pratiche eseguire una trasformazione del dominio dell'immagine, in uno dove è più semplice eseguire il filtraggio*

*di Giuseppe Cardinale Ciccotti*

## La trasformata di Fourier

La più nota fra le trasformazioni, perché indubbiamente la più utile, è la trasformata di Fourier. Al di là di tutta la complessa teoria matematica che la definisce, quello che è importante per i nostri scopi è il calcolo della trasformata dal punto di vista pratico e come devono essere eseguite le operazioni nel nuovo dominio.

Per spirito di completezza, tuttavia, cercheremo di spiegare in modo generale ed intuitivo il significato della trasformata di Fourier. Consideriamo dunque un segnale acustico, per esempio, un suono, per dirla in breve; una ovvia rappresentazione è un grafico dove in ascissa è l'asse dei tempi ed in ordinata l'ampiezza, cioè nel caso di onda sonora il volume, del suono in quel momento. Il grafico sarà perciò un'onda che non è nient'altro che la variazione dell'ampiezza nel tempo.

Tutti però sappiamo bene che un suono può essere definito tramite le frequenze che lo compongono, per esempio un la di un qualsiasi strumento è la somma di una frequenza di 414 Hz più le sue eventuali armoniche, il numero e l'intensità di queste, è funzione dello strumento stesso.

Una rappresentazione alternativa di questo segnale può essere ancora un grafico dove però in ascissa troviamo la frequenza ed in ordinata ancora una grandezza che indica l'ampiezza. L'ampiezza, in questo caso, sarà il «volume» di quella particolare frequenza, se il nostro suono è definito come un la piuttosto «pulito», sul grafico troveremo un

impulso a 414 Hz di ampiezza elevata e alle frequenze delle varie armoniche degli impulsi via via più contenuti man mano che cresce l'ordine della armonica.

Il grafico così ottenuto è una rappresentazione dello spettro frequenziale del segnale dal quale si vede subito quali sono le frequenze significative del segnale.

È quasi superfluo dire che un segnale caratterizzato da un contenuto frequenziale esteso, cioè distribuito a varie frequenze, avrà un grafico nel dominio della frequenza come un'onda.

Se del nostro segnale ora volessimo sopprimere alcune armoniche indesiderate basterebbe far passare il segnale dentro un filtro adeguato che «ammazza» tali frequenze.

Ma come si comporta un filtro nel dominio della frequenza?

Vale a dire se per filtrare nel tempo eseguiamo la convoluzione, quale operazione effettueremo nella frequenza?

Dalle proprietà della trasformata di Fourier, si trova che la convoluzione nel tempo equivale ad una moltiplicazione nel dominio della frequenza.

Il filtro è descritto come una funzione della frequenza e quindi il filtraggio consisterà in una moltiplicazione di due funzioni; dal punto di vista grafico, basterà sovrapporre i profili del segnale e del filtro calcolando per ogni valore della frequenza il prodotto del segnale e del filtro in quel punto.

Naturalmente se il nostro scopo è quello di mantenere inalterato il segnale eliminando soltanto alcune frequenze indesiderate, il filtro varrà zero per tali

frequenze ed uno in tutto il resto del dominio.

Sagomando opportunamente questo filtro possiamo eseguire qualsivoglia filtraggio.

Anche se non l'abbiamo detto espressamente, abbiamo supposto finora i segnali continui, definiti cioè per tutti i valori del tempo e quindi della frequenza. Ciò è senz'altro vero per i dispositivi analogici come ad esempio i filtri del controllo dei toni dei preamplificatori hi-fi o i filtri di cross-over. Quando invece ci occupiamo di elaborazione digitale, manipoliamo dati numerici per esempio i campioni di un qualsiasi segnale sia esso un suono o un'immagine.

Anche in questo caso la trasformata di Fourier è definita e pari ancora ad un segnale campionato.

Il filtro è anch'esso definito da una serie di campioni e quindi eseguire il filtraggio è un semplice ciclo di moltiplicazioni.

Naturalmente il filtro è impostato da chi esegue il filtraggio direttamente nel dominio di Fourier, per cui non è necessario eseguirne la trasformata.

Chiaramente dopo il filtraggio bisognerà eseguire una antitrasformata per ritornare nel dominio di partenza e fruire del segnale.

Ricapitolando i passi da eseguire sono:

- 1) Eseguire la trasformata di Fourier del segnale sorgente.
- 2) Moltiplicare il segnale trasformato per il filtro.
- 3) Eseguire la trasformata inversa del segnale prodotto.

## La trasformata discreta di Fourier

Vediamo ora di capire come si può organizzare un algoritmo per il calcolo della trasformata discreta. Per il caso monodimensionale la trasformata è definita dalla formula:

$$F(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) e^{-j2\pi x u / N}$$

dove N sono i campioni del segnale di partenza e x ed u assumono valori interi tra 0 e N-1.

La trasformata inversa o antitrasformata come è anche detta, è invece definita da:

$$f(x) = \sum_{u=0}^{N-1} F(u) e^{j2\pi x u / N}$$

sempre con u ed x che vanno da 0 a N-1.

Come si vede dalle definizioni la trasformata restituisce valori complessi e

per tanto per estrarre lo spettro vanno calcolati i moduli per ciascun campione. Il modulo di un numero complesso è definito da:

$$|F(u)| = \sqrt{\text{Re}(F(u))^2 + \text{Im}(F(u))^2}$$

dove Re e Im restituiscono la parte reale ed immaginaria dell'argomento.

Il caso bidimensionale ci interessa molto di più in quanto un'immagine è stata definita proprio come una funzione f(x,y) e la sua digitalizzazione come un array bidimensionale di pixel fra 0..N-1, 0..M-1.

La trasformata bidimensionale è un'estensione della trasformata monodimensionale e quindi:

$$F(u,v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) e^{-j2\pi(ux/M + vy/N)}$$

con u=0,1,2..M-1 e v=0,1,2..N-1.

La trasformata inversa è parimenti definita da:

$$f(x,y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v) e^{j2\pi(ux/M + vy/N)}$$

Se come spesso capita o meglio si fa in modo che capiti, M=N si ottengono le seguenti formule:

$$F(u,v) = \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) e^{-j2\pi(ux+vy)/N}$$

$$f(x,y) = \frac{1}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F(u,v) e^{j2\pi(ux+vy)/N}$$

Queste ultime espressioni sono le più usate in pratica perché godono di proprietà particolari che ne semplificano il calcolo.

## Separabilità e traslazione

La proprietà più utile di cui godono la trasformata bidimensionale e la sua inversa definite su uno spazio quadrato di campioni è la separabilità cioè il poter esprimere le formule che le definiscono nel seguente modo:

$$F(u,v) = \frac{1}{N} \sum_{x=0}^{N-1} e^{-j2\pi x u / N} \sum_{y=0}^{N-1} f(x,y) e^{-j2\pi v y / N}$$

$$f(x,y) = \frac{1}{N} \sum_{u=0}^{N-1} e^{j2\pi x u / N} \sum_{v=0}^{N-1} F(u,v) e^{j2\pi v y / N}$$

Per i nostri scopi, il principale vantaggio della separabilità sta nel fatto che F(u,v) e f(x,y) possono essere ottenute in due passi attraverso successive applicazioni della trasformata di Fourier monodimensionale o della sua inversa.

Ciò è ancora più evidente se l'espressione precedente è messa nella forma:

$$F(u,v) = \frac{1}{N} \sum_{x=0}^{N-1} F(x,v) e^{-j2\pi x u / N}$$

dove

$$F(x,v) = N \left[ \frac{1}{N} \sum_{y=0}^{N-1} f(x,y) e^{-j2\pi v y / N} \right]$$

In quest'ultima ogni espressione fra le parentesi quadre è la trasformata monodimensionale per v=0,1,..N-1.

Quindi la funzione bidimensionale F(x,v) è ottenuta attraverso la trasformata lungo ciascuna riga di f(x,y) e multipli-

Figura 1 - Listato della routine in Fortran per il calcolo della FFT di un vettore a 1024 campioni. La stessa routine calcola anche la antitrasformata.

```

SUBROUTINE FFT(F, LN)
COMPLEX F(1024), U, W, T, CMLPX
PI=3.141593
N=2**LN
NV2=N/2
NM1=N-1
J=1
DO 3 I=1, NM1
  IF(I.GE.J) GO TO 1
  T=F(J)
  F(J)=F(I)
  F(I)=T
1  K=NV2
2  IF(K.GE.J) GO TO 3
  J=J-K
  K=K/2
  GO TO 2
3  J=J+K
DO 5 L=1, LN
  LE=2**L
  LE1=LE/2
  U=(1.0, 0.0)
  W=CMLPX(COS(PI/LE1), -SIN(PI/LE1))
  DO 5 J=1, LE1
    DO 4 I=J, N, LE
      IP=I+LE1
      T=F(IP)*U
      F(IP)=F(I)-T
4      F(I)=F(I)+T
5      U=U*W
DO 6 I=1, N
6  F(I)=F(I)/FLOAT(N)
RETURN
END

```

cando il risultato per N. La  $F(u,v)$  è infine il risultato della trasformata di  $F(x,y)$  lungo ogni colonna dell'immagine.

Lo stesso risultato si ottiene se il calcolo è eseguito prima lungo le colonne e poi le righe; nelle formule precedenti ciò significa invertire l'ordine delle sommatorie.

Il medesimo procedimento viene poi applicato alla  $F(u,v)$  per estrarre l'antitrasformata.

Un'altra utile proprietà che torna comoda nell'impostazione dell'algoritmo è quella della traslazione:

$$f(x,y)e^{j2\pi(u_0x+v_0y)/N} \Leftrightarrow F(u-u_0, v-v_0)$$

se poniamo  $u_0$  e  $v_0$  pari a  $N/2$  cioè trasliamo l'origine nel centro dell'immagine otteniamo:

$$f(x-x_0, y-y_0) \Leftrightarrow F(u,v)e^{-j2\pi(ux_0+vy_0)/N}$$

Si vede quindi che l'origine della trasformata di Fourier può essere spostata al centro del corrispondente quadrato  $N \times N$  delle frequenze semplicemente moltiplicando  $f(x,y)$  per  $(-1)^{x+y}$ .

Bisogna inoltre tener presente come si potrebbe facilmente dimostrare che il modulo non viene modificato da una traslazione perciò la visualizzazione dello spettro può essere facilmente posizionata come si vuole.

La proprietà della traslazione è assai importante, considerata insieme al fatto che la trasformata di Fourier è periodica con periodo N, cioè:

$$F(u,v) = F(u+N, v+N)$$

Osservando ora che molti segnali, tra cui in particolar modo le immagini, contengono quasi sempre la maggior parte dell'energia compresa in una banda contigua all'origine, ne segue che se lo spettro viene visualizzato come un'immagine, cioè scrivendo i valori  $F(u,v)$  nei pixel indicizzati da riga e colonna della memoria video del nostro elaboratore, ci possiamo aspettare due macchie attorno all'origine e attorno all'angolo opposto. Una rappresentazione siffatta

dello spettro non è che sia molto d'aiuto; conviene molto di più traslare di un semiperiodo la trasformata in modo da posizionare lo spettro nel centro dell'immagine e coglierne il significato.

Esistono altre proprietà della Trasformata ma per non confondere troppo le idee, preferiamo rimandare la spiegazione al momento in cui ce ne serviremo.

### Fast Fourier Transform

Probabilmente più d'uno dei lettori avrà già sentito parlare di quest'algoritmo noto con il suo acronimo FFT, perché è uno dei capisaldi dell'elaborazione numerica tanto che spesso è utilizzato in ambito scientifico come benchmark di riferimento per le prestazioni dei sistemi di calcolo.

Di seguito illustriamo l'algoritmo tramite la sua formulazione matematica per il caso monodimensionale visto che il caso bidimensionale può essere ricavato da quest'ultimo sfruttando la proprietà di separabilità.

L'algoritmo si basa sul metodo dei «raddoppi successivi» e ci conviene per compattezza riscrivere l'equazione che definisce la trasformata discreta monodimensionale, nel seguente modo:

$$F(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x)W_N^{ux}$$

dove

$$W_N = e^{-j2\pi/N}$$

assumendo che  $N=2^n=2 \cdot M$  dove n ed M sono interi positivi.

Sostituendo nell'equazione precedente otteniamo:

$$F(u) = \frac{1}{2M} \sum_{x=0}^{2M-1} f(x)W_{2M}^{ux} = \frac{1}{2} \left( \frac{1}{M} \sum_{x=0}^{M-1} f(2x)W_{2M}^{(2x)u} + \frac{1}{M} \sum_{x=0}^{M-1} f(2x+1)W_{2M}^{(2x+1)u} \right)$$

dato che  $W_{2M}^{2x} = W_M^{x}$  allora la trasformata è data da:

$$F(u) = \frac{1}{2} \left( \frac{1}{M} \sum_{x=0}^{M-1} f(2x)W_M^{ux} + \frac{1}{M} \sum_{x=0}^{M-1} f(2x+1)W_M^{ux} W_{2M}^{x} \right)$$

Se ora definiamo:

$$F_{\text{pari}}(u) = \frac{1}{M} \sum_{x=0}^{M-1} f(2x)W_M^{ux}$$

per  $u=0,1,2,\dots,M-1$  e

$$F_{\text{dispari}}(u) = \frac{1}{M} \sum_{x=0}^{M-1} f(2x+1)W_M^{ux}$$

la trasformata si riscrive così:

$$F(u) = \frac{1}{2} (F_{\text{pari}}(u) + F_{\text{dispari}}(u)W_{2M}^u)$$

Infine poiché

$$W_M^{u+M} = W_M^u \text{ e } W_{2M}^{u+M} = -W_{2M}^u$$

ne segue che

$$F(u+M) = \frac{1}{2} (F_{\text{pari}}(u) - F_{\text{dispari}}(u)W_{2M}^u)$$

L'implementazione è la traduzione delle formule sopra esposte ed in realtà si basano sull'applicazione successiva della divisione in parte pari e dispari.

Facciamo un esempio per semplificare le idee ed eseguiamo la FFT di un vettore a 8 campioni  $[f(0), \dots, f(7)]$ .

Secondo le equazioni che abbiamo scritto la parte pari della trasformata usa i campioni  $[f(0), f(2), f(4), f(6)]$ , mentre la dispari i campioni  $[f(1), f(3), f(5), f(7)]$ .

Chiaramente la sommatoria gode della proprietà additiva e quindi posso calcolare la somma dei quattro termini come somma di termini a due a due. In tal modo riconduco il calcolo sempre ad una somma di due termini.

Quindi i due sottovettori pari e dispari vengono divisi a loro volta in due sottovettori pari e dispari cioè per il primo otterrò:

$$[f(0), f(4)] \text{ e } [f(2), f(6)]$$

mentre per il secondo

$$[f(1), f(5)] \text{ e } [f(3), f(7)]$$

Non occorre che divida ulteriormente i vettori perché sono già di due termini ciascuno.

Si procede quindi alle moltiplicazioni per i fattori W ed alle somme successive.

Come si può evincere, la cosa importante è l'ordinamento del vettore perché se i campioni fossero nell'ordine

$$[f(0), f(4), f(2), f(6), f(1), f(5), f(3), f(7)],$$

il calcolo sarebbe diretto.

Fortunatamente l'ordinamento può essere risolto con un semplice rovesciamento dei termini della parola binaria che mantiene l'indice.

Più precisamente bisogna invertire i bit della parola secondo una simmetria speculare rispetto ad uno degli estremi; così per esempio  $4d=1002$  diventa  $0012=1d$  quindi il campione di indice 4 nell'array è scambiato di posto con quello che ha indice 1 e così via.

### Bibliografia

R. C. Gonzalez, P. Wintz, «Digital Image Processing», Addison-Wesley, 1987

P. Melli, «L'elaborazione digitale delle immagini», Franco Angeli, 1990

Naturalmente i campioni trasformati sono nell'ordine corretto.

Tutto quello che abbiamo precisato finora vale anche per la trasformata inversa, infatti basta cambiare nell'algoritmo la  $f(x)$  con la  $F(u)$  per calcolare l'antitrasformata.

In figura 1 trovate un'implementazione Fortran della subroutine che calcola la FFT di un vettore  $F$  di lunghezza  $2^N$  passato in ingresso. In uscita lo stesso vettore contiene i campioni trasformati.

Il primo loop, il DO 3, si occupa di riordinare i dati.

Il secondo loop, il DO 5, porta a termine i calcoli in cascata e l'ultimo loop, il DO 6, divide i risultati per  $N$ .

Naturalmente questo programma è suscettibile di numerosi miglioramenti, in primis la memorizzazione tabellare dei coefficienti  $W$ . Tuttavia secondo quanto è affermato in bibliografia questa routine così semplice è inferiore soltanto del 12% in efficienza rispetto ad un programma Fortran ottimizzato.

Inoltre se  $F$  è un vettore che contiene i campioni nel dominio della frequenza

tale programma calcolerà la trasformata inversa.

Anche se a prima vista non può sembrare, quest'algoritmo è assai efficiente; rispetto al calcolo diretto della trasformata su un vettore di 1024 campioni, l'algoritmo risulta essere 102.4 volte più veloce, su 8K campioni addirittura 630.15 più rapido.

### Conclusioni

Appuntamento molto teorico, forse un po' pesante ma sicuramente indispensabile, per chiunque voglia cimentarsi nell'elaborazione digitale delle informazioni. Lo spazio è tiranno e ci costringe a chiudere senza poter fare un esempio di filtraggio nel dominio della frequenza.

Una domanda sorge spontanea direbbe qualcuno... Ma è veramente conveniente fare tutto questo giro di trasformate per eseguire un filtraggio? La risposta è generica: dipende dalla situazione, ci sono casi in cui il filtro è ben definito nella frequenza come vedremo

mentre il suo alter-ego nel tempo è assai complicato. In altri casi se il filtro nel tempo è ben sagomato e si dispone di un algoritmo di convoluzione efficace non conviene trasformare.

Qualcun altro si chiederà come mai abbiamo mostrato un algoritmo implementato in Fortran, ebbene per applicazioni di questo tipo il Fortran è ancora largamente usato visto che si dispone di un esteso patrimonio di librerie proprie in Fortran. Inoltre i sacri testi, sono di qualche anno addietro quando il Fortran era ancora in auge.

Ecco dunque un modo per coinvolgere i lettori: aspettiamo le vostre implementazioni dell'algoritmo di FFT in C, in Pascal o in qualsiasi altro linguaggio vorrete inviarcelo.

Potrebbe essere interessante fare una classifica di prestazioni, se qualcuno volesse poi cimentarsi in una versione parallela, magari in Occam su Transputer, respireremmo aria di Supercomputing sui nostri Personal!

ME

CON INFOMATE POWER GUARDIAN LA VOSTRA RETE NON SARA' MAI COSI'

MESI **15** GARANZIA

I gruppi di continuità Infomate

Power Guardian UA-1000 e UA-

3000 proteggono il server della vostra rete locale contro i blackout, le microinterruzioni, i disturbi impulsivi, le sovratensioni istantanee ed il rumore nella linea di alimentazione elettrica. Con power Guardian i vostri dati e l'hardware sono al sicuro da quei problemi che i raddrizzatori ed i soppressori non sono progettati per affrontare.

Potete scegliere fra le versioni da 1000 VA e da 3000 VA. Entrambe forniscono un tempo più che sufficiente per salvare le informazioni e chiudere ordinatamente il vostro sistema, prevenendo perdite di informazioni potenzialmente devastanti. Ed entrambe possono essere predisposte per la chiusura e la ripartenza automatica in remoto, fornendo così la soluzione definitiva alla protezione automatica con la massima semplicità di uso.

Per una protezione affidabile dell'alimentazione la scelta è chiaramente Infomate Power Guardian. Progettato per tenere la vostra rete fuori dal buio - completamente certificato TUV, garanzia estesa di 15 mesi.

**infomate**

Per qualsiasi informazione sul Power Guardian Infomate rivolgetevi a Infomate Corp. 12F, 77 Keelung Rd., Sec. 2, Taipei, Taiwan R.O.C. Tel: 886-2-7362933 Fax: 886-2-7334403

Infomate è un marchio registrato e Power Guardian è un marchio di fabbrica della Infomate Corp.



UA-1000 UA-3000

