

Proiect TPI

Tehnici de prelucrare a imaginilor în domeniul transformatei Fourier

Elaborat:

Melnic Anastasia

1.Tema lucrării

Lucrarea urmărește înțelegerea fundamentelor teoretice a unor tehnici de prelucrare a imaginilor în domeniul transformatei Fourier și implementarea acestora pe procesorul Analog Devices Blackfin BF533. M-am concentrat pe calculul Transformatei Fourier Discrete (DFT), analiza spectrului de amplitudine și fază, aplicarea filtrelor Butterworth trece-sus și trece-jos și reconstituirea imaginii, folosind inversa Transformatei Fourier Discrete.

2.Fundamentare teoretică

Filtrarea în domeniul Fourier presupune :

- 1.Aplicarea transformatei Fourier Discrete și translatarea spectrului
- 2.Multiplicarea transformatei imaginii cu un filtru în frecvență
- 3.Aplicarea inversei transformatei Fourier Discrete

2.1.Transformata Fourier Discretă

Transformata Fourier discretă bidimensională (1) este esențială în procesarea imaginilor digitale, permițând analiza componentelor frecvențiale ale imaginii. Pentru o imagine digitalizată $U(x,y)$, transformata Fourier este o funcție complexă $F(u,v)$ care reflectă distribuția frecvențelor spațiale. Aceasta este utilizată pentru obținerea spectrului de amplitudine și spectrului de fază, facilitând aplicarea tehnicilor de filtrare și restaurare.

$$F(u,v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} U(x,y) e^{-2\pi j \left(\frac{ux}{M} + \frac{vy}{N} \right)} \quad (1)$$

Pentru a calcula spectrul de amplitudine și de fază, se folosesc formulele (2) și (3):

$$|F(u,v)| = \sqrt{[\operatorname{Re} F(u,v)]^2 + [\operatorname{Im} F(u,v)]^2} \quad (2)$$

$$F(u,v) = |F(u,v)| e^{-j\varphi(u,v)} \quad (3)$$

Spectrul de amplitudine ne arată cu cât contribuie o anumită frecvență la structura generală a imaginii, iar spectrul de fază conține informația de poziționare a detaliilor din imagine.

2.2.Filtrare în domeniul Fourier

Filtru Butterworth trece-jos este definit de relația:

$$H(u, v) = \frac{1}{1 + \left(\frac{W(u, v)}{W_0}\right)^{2n}}, \quad (4)$$

unde W este distanța față de origine în planul (u, v) , iar n este parametru ce controlează panta de descreștere a filtrului.

Filtru Butterworth trece-sus este definit de relația:

$$H(u, v) = \frac{1}{1 + \left(\frac{W_0}{W(u, v)}\right)^{2n}} \quad (5)$$

2.3. Inversa transformatei Fourier

Trecerea din domeniul spectral în domeniul spațial se face pe baza transformatei Fourier discrete inverse (IDFT):

$$f(x, y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) e^{2\pi j \left(\frac{ux}{M} + \frac{vy}{N} \right)} \quad (6)$$

Exemplu:

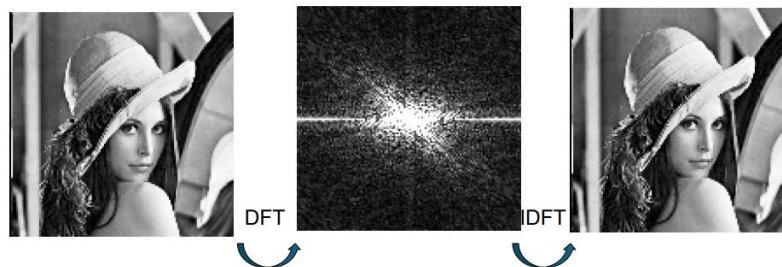


Figura 1 Trecerea din domeniul spațial în domeniul spectral și invers

3.Implementarea soluției adoptate

Schema bloc a proiectului este:

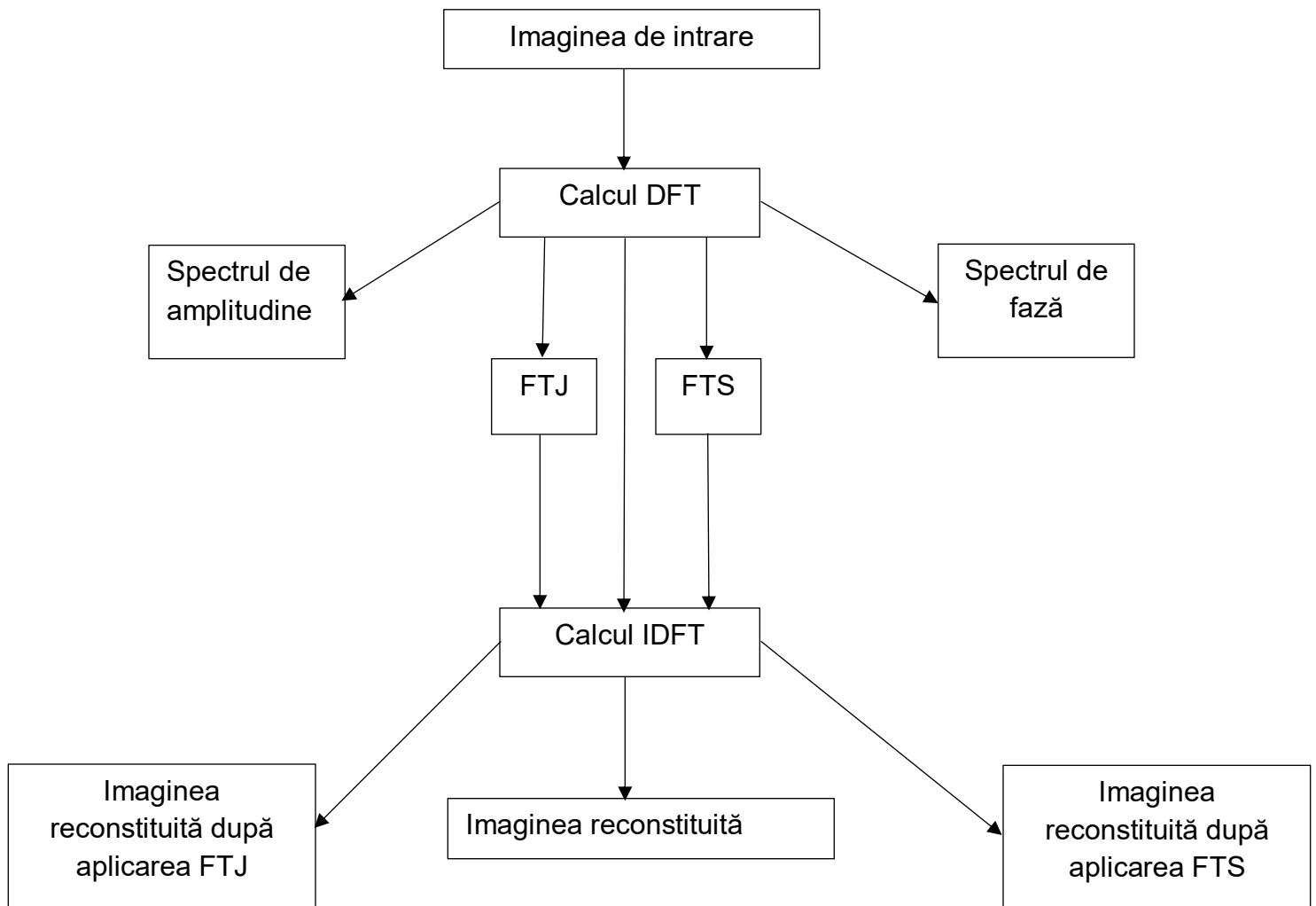


Figura 2 Schema bloc

1. Am ales o imagine de intrare grayscale 8x8, deoarece, dacă ar fi avut o dimensiune mai mare, simularea ar fi rulat extrem de lent (câteva ore).
2. Am translatat imaginea prin multiplicare cu $(-1)^{x+y}$ pentru a centra spectrul, apoi am calculat DFT-ul, conform formulei matematice (1). Am ales inițial să implementez DFT în detrimentul FFT pentru simplitate.
3. Am afișat spetrele de amplitudine și fază obținute.
4. Am implementat funcția de calcul a IDFT, pentru a trece din domeniul spectral în domeniul spațial și a reconstitui imaginea. IDFT-ul aplică aceeași translatare ca și DFT pentru corectarea poziționării.

5. Inițial, pentru verificarea algoritmilor de calcul a DFT și IDFT, am vizualizat imaginea reconstituită direct, fără a efectua alte operații în domeniul frecvență.
6. Am aplicat asupra spectrului imaginii inițiale 2 filtre Butterworth (trece-jos și trece-sus).
7. Am folosit IDFT pentru a trece iar din domeniul spectral în domeniul spațial și a vizualiza imaginile rezultate.

void DFT_2D(uint8 *imgInput, float *parte_reala_DFT, float* parte_imag_DFT) – calculează transformata Fourier Discretă

void Spectru_modul(float *parte_reala_DFT, float* parte_imag_DFT, float* modul_DFT) – determină spectrul de amplitudine

void Spectru_faza(float *parte_reala_DFT, float* parte_imag_DFT, float* faza_DFT) – determină spectrul de fază

void IDFT_2D(float *parte_reala_DFT, float *parte_imag_DFT, uint8 *imgOutput) – calculează inversa transformatei Fourier discrete

void Butterworth_LPF(uint8 *imgOutput, float* parte_reala_DFT, float* parte_imag_DFT, float W0) – aplică filtrul trece-jos

void Butterworth_HPF(uint8 *imgOutput, float* parte_reala_DFT, float* parte_imag_DFT, float W0) – aplică filtrul trece-sus

4.Rezultate experimentale

În urma rulării codului, am vizualizat, cu ajutorul Image Viewer din VisualDSP++, imaginile:

- `imgInput[8*8]`

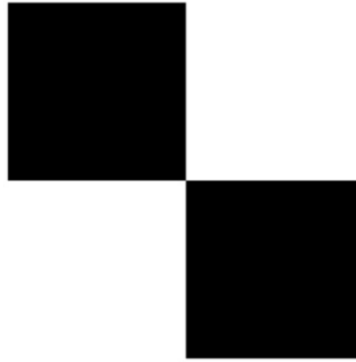


Figura 3 Imaginea inițială

- `Magnitude_spectrum[8*8]`

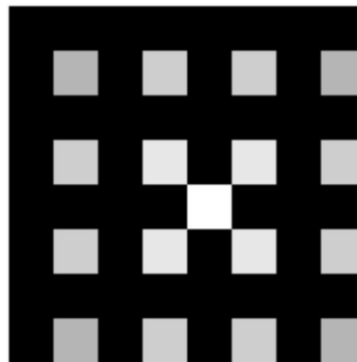


Figura 4 Spectrul de amplitudine

- `Phase_spectrum[8*8]`



Figura 5 Spectrul de fază

- `imgReconstituitaDirect[8*8]`

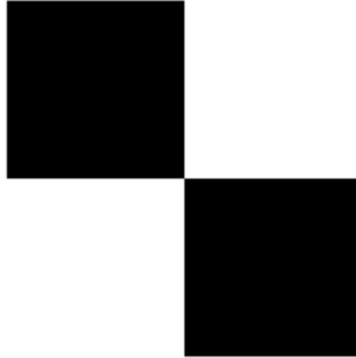


Figura 6 Imaginea reconstituită direct

- `imgOut_Butterworth_LPF[8*8]`



Figura 7 Imaginea rezultată în urma aplicării FTJ

- `imgOut_Butterworth_HPF[8*8]`

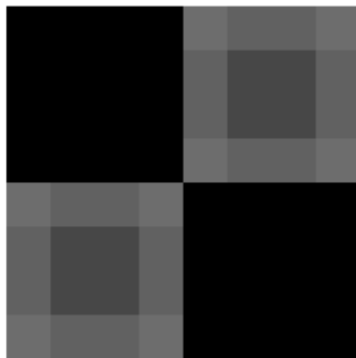


Figura 8 Imaginea rezultată în urma aplicării FTS

5. Concluzii

- Algoritmul DFT este ineficient pentru imagini/semnale mari și rulează extrem de lent în mediul de simulare pentru Blackfin BFF533. Ca metoda de îmbunătățire a proiectului aș propune implementarea algoritmului FFT(Fast Fourier Transform), care este mult mai rapid decât DFT.
- Am observat că implementările funcțiilor de calcul a DFT și IDFT sunt corecte, având în vedere că imaginea reconstituită (Figura 6) este identică cu cea inițială (Figura 3).
- Spectrul de amplitudine (Figura 4) indică cât de “intense” sunt frecvențele prezente în imagine, iar spectrul de fază (Figura 5) conține informații despre poziția spațială a detaliilor din imagine.
- Filtrul trece-jos elimină detaliile de înaltă frecvență (contururile clare, zgomotul) și păstrează componentele de frecvență joasă, ceea ce produce un efect de blur artistic.
- Filtrul trece-sus evidențiază marginile și detaliile fine, eliminând zonele uniforme și variațiile lente.

6.Bibliografie

[1] A. V. Miclea, “Tehnologii de prelucrare a informației”, Curs 8

[2] The Medium website, [Digital Image Processing: Fourier Transform | by Jeffrey | NTUST-AIVC | Medium](#)