R11945072 生醫電資一 張栢彥

Homework 4 (Due: 5/24)

(1) Write a Matlab or Python program to measure the structural similarity (SSIM) of two images A and B. The sizes of A and B are equivalent.

SSIM(A, B, c1, c2)

where c1 and c2 are some adjust constants.

The Matlab or Python code should be handed out by NTUCool. (20 scores)

(2) (a) How do we use three real multiplications to implement a complex multiplication? (10 scores)

(b) Suppose that

$$\begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix} = \begin{bmatrix} b_1 & -b_2 & -b_3 & -b_4 \\ b_2 & b_1 & b_4 & -b_3 \\ b_3 & b_4 & b_1 & -b_2 \\ -b_4 & b_3 & b_2 & b_1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix}$$

How do we implement above matrix operation with the least number of real multiplications? (10 scores)

(3) Determining the numbers of real multiplications for the (a) 125-point DFT, (b) the 147-point DFT, and (c) the 385-point DFT. (15 scores)

(4) What is the complexity of the 3D DFT as follows? Express the solution in terms of the big order. (10 scores)

$$Y[p,q,r] = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1}\sum_{k=0}^{K-1} e^{-j2\pi\frac{pm}{M}-j2\pi\frac{qn}{N}-j2\pi\frac{rk}{K}} x[m,n,k]$$

3DDFT (NlogN)

(5) Suppose that there are 1200 cars in a dataset and an algorithm detects 1000 cars. However, among the detected cars, 100 of them are in fact other objects. Determine the precision, the recall, and the F-score of the algorithm. (10 scores)

(6) Suppose that length($x[n]$) = 1100. What is the best way to implement the convolution of $x[n]$ and $y[n]$ if

$x[n] * y[n]$ 會用FT?

(a) length($y[n]$) = 500, (b) length($y[n]$) = 40,

(c) length($y[n]$) = 6, and (d) length($y[n]$) = 2 ? (25 scores)

Please show (i) the convolution method (direct, sectioned convolution, or non-sectioned convolution), (ii) the number of points of the FFT, (iii) and the number of real multiplications for the best implementation method. Also, consider the general case where $x[n]$ and $y[n]$ are complex sequences and the FFT of $y[n]$ can be computed in prior.

6600 最少 → 分析 time 最少的計算量

(Extra): Answer the questions according to your student ID number.

(ended with (2, 7), (3, 8), (4, 9), (0, 5))

8點DCT. 2,4,6 乘法量是多少?

(2)

(a) 假設要做2個複數運算,通常要4個 MUL

$$\left.\begin{aligned} A &= a+bi \\ B &= c+di \end{aligned}\right\} \text{故 } ac,\ ad,\ bc,\ bd \Rightarrow ac+(ad)i+(bc)i-bd$$

若要改為 3 real multiplications. 則替換代數,如:

$$e = a \times c$$
$$f = b \times d$$
$$g = (a+b) \times (c+d)$$

$$\left.\begin{aligned} &\text{實數部分}: e-f \\ &\text{虛數部分}: g-e-f \end{aligned}\right\} \text{naive method}$$

⇒ 成功以 3 multiplications 合成2個複數運算!

(b) 對於矩陣運算乘法,我們不能將乘法次數減少到標準方法之下,因每個運算都需施行該點積(但可用其它方法降低總運算量,含"+","×")

$$\begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix} = \begin{bmatrix} b_1 & -b_2 & -b_3 & -b_4 \\ b_2 & b_1 & b_4 & -b_3 \\ b_3 & b_4 & b_1 & -b_2 \\ -b_4 & b_3 & b_2 & b_1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix}$$

⇒ 故根據矩陣運算,仍需進行 $4\times4 = \boxed{16}$ 次的 real multiplication.

⇒ 也可簡化為2個2x2乘法運算 $2\times(2\times2) = \boxed{8\text{次}}$

(3) 是直接使用DFT(不用FFT)進行運算之方法,為 $O(N^2)$

不確定

(a) 125-point DFT = 25×5

$$\Rightarrow MUL_{25}\times5 + 25\times MUL_5 = 148\times5 + 25\times10 = \boxed{990}$$

(b) 147-point DFT $\quad 147 = 3\times7^2$

$$\Rightarrow 49\times \underset{1/2}{MUL3} + 3\times(7\times16 + 7\times16 + 3\times6\times6) = \boxed{1094}$$

(c) 385-point DFT $\quad 385 = 5\times7\times11$

$$\Rightarrow \frac{385}{5}MUL_5 + \frac{385}{7}MUL_7 + \frac{385}{11}MUL_{11} = 77\times10 + 55\times16 + 35\times40 = \boxed{3050}$$

(4)

若使用DFT的直接計算，以N表為每個維度大小，複雜度為 $O(N^6)$

(即每維度執行 N^2，共3個維度：$O(N^2)^3 = \boxed{O(N^6)}$

倘若使用FFT的 radix-2 Cooley Tukey 算法，每個 $O(N^2)$ 降到 $O(N\log N)$ [N個點]

而若有3維，即有 N^3 個點，則利用FFT後 complexity 變為：$\boxed{O(N^3 \log N)}$ #

(5)

總量：1200 輛 (正面)

檢測：1000輛 FP：100輛 TP：900輛

Precision：$TP/(TP+FP) = 900/1000 = 0.9$

Recall：$TP/(TP+FN) = 900/1200 = 0.75$

F-Score：$\dfrac{2 \times Precision \times Recall}{Precision + Recall} = \dfrac{2 \times 0.9 \times 0.75}{0.9 + 0.75} \fallingdotseq 0.818181 \approx 0.82$

(6) $length(x[n]) = 1100$

(a) $length(y[n]) = 500$

(i) 適用於 $M \approx N$，故 可用IFFT (non-Sectioned Conv.) 方法

(ii) 下一個大於 $1100+500-1 = 1599$ 的最小乘法量 $P \geq 1599$ ⇒ $P = 1680$

(iii) $MUL = 2 \times MUL_{1680} + 3 \times 1680 = 25880$

(b) $length(y[n]) = 40$

(i) 適用於 $M \ll N$，故 原可用 Sectioned Convolution (但題目無給分段最適L，且 Sectioned 的 complexity 會過大。) → 改用 IFFT 方法

(ii) 下一個大於 $40 + 1100 - 1 = 1139$ 的 min乘法量 ⇒ $P \geq 1139$ → $P = 1152$

(iii) $MUL = 2 \times MUL_{1152} + 3 \times 1152$
$= 2 \times 7088 + 3 \times 1152 = 17632$ #

(c) length (y[n]) = 6

(i) 適用於 M 極小，故
可用 direct computing / or IFFT

比較 (ii) 根據(iii)的比較
採用大於 1105 的最小乘法量點：1024 = P

(iii) ① 3MN
$= 3 \times 6 \times 1100 = 19800$

② IFFT： $1100 + 6 - 1 = 1105$ ， $P \geq 1105$
$P = 1024$

$MUL = 2 \times 7436 + 3 \times 1024 = 17944$

較小，故用此法 (IFFT)

(d) length (y[n]) = 2

(i) 適用於 M 極小，故
可用 direct computing

(ii) 不適用於
direct computing

(iii) 3MN
$= 3 \times 2 \times 1100 = 6600$

<extra point> 學號尾數 2

Q. Part 1 簡化後用多少 multiplication

$y_0 = 0.7010\,(z_0 + z_1 + z_2 + z_3)$

$y_4 = 0.7010\,(z_0 - z_1 - z_2 + z_3)$

$$\begin{bmatrix} y_2 \\ y_6 \end{bmatrix} = \begin{bmatrix} 0.9239 & 0.3827 \\ 0.3827 & -0.9239 \end{bmatrix} \begin{bmatrix} z_0 - z_3 \\ z_1 - z_3 \end{bmatrix}$$

$\Rightarrow$ $\boxed{4 \text{ 個 MULs}}$ #