

2.12

令人眼所能辨別的灰階強度變化表示為 ΔG .

$$\text{則 } \Delta G = \frac{2^k}{2^k} = 8$$

$$\Rightarrow 2^k = 32$$

$$\Rightarrow k = 5$$

也就是說32個灰階以下會出現假輪廓的現象

2.16

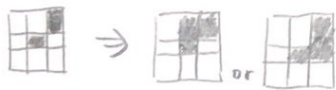
<演算法>

1. 令路徑的起點位置為 (m, n) , 且任何對角相鄰的點為 (p, q) .

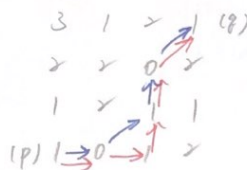
2. 觀察是否有 (m, n) 四相鄰的點與 (p, q) 四相鄰的點重疊且符合路徑條件, 若有則取該點為下個 (m, n) 點, 若無則取 (p, q) 為下個 (m, n) 點.

3. 重複步驟1~2從路徑起點直至終點, 所有被找到的 (m, n) 的集合即為新路徑.

此演算法的結果會讓原本「斜對角關係」的路徑變為「四相鄰關係」如下圖所示.



2.18

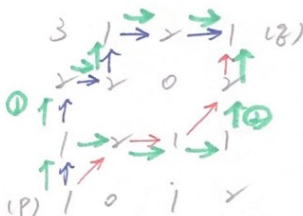


(a) $V = \{0, 1\}$

① 4-path: 找不到一符合 $V = \{0, 1\}$ 且四相鄰關係之路徑

② 8-path: 最短路徑為4步, 如上圖藍色路徑所示.

③ m-path: 最短路徑為5步, 如上圖紅色路徑所示.



(b) $V = \{1, 2\}$

① 4-path: 最短路徑為6步, 如上圖藍色路徑所示.

② 8-path: 最短路徑為4步, 如上圖紅色路徑所示.

③ m-path: 符合m-path的最短路徑有2條, 如上圖綠色路徑所示, 皆為6步.

2.37

(a) scaling transformation

$$A = \begin{bmatrix} C_x & 0 & 0 \\ 0 & C_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$A^{-1} = \frac{1}{\text{HeterA}} \begin{bmatrix} C_y & 0 & 0 \\ 0 & C_x & 0 \\ 0 & 0 & C_x C_y \end{bmatrix} = \begin{bmatrix} \frac{1}{C_x} & 0 & 0 \\ 0 & \frac{1}{C_y} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

(b) translation transformation

$$A = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

$$A^{-1} = \frac{1}{\text{HeterA}} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -t_x & -t_y & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -t_x & -t_y & 1 \end{bmatrix}$$

c) vertical/horizontal shearing transformation

⊙ vertical

$$A = \begin{bmatrix} 1 & S_v & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$A^{-1} = \frac{1}{\det A} \begin{bmatrix} 1 & 0 & 0 \\ -S_v & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ -S_v & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

⊙ horizontal

$$A = \begin{bmatrix} 1 & 0 & 0 \\ S_h & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$A^{-1} = \frac{1}{\det A} \begin{bmatrix} 1 & -S_h & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & -S_h & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

d) rotation transformation

$$A = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$A^{-1} = \frac{1}{\det A} \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

e) translation + rotation transformation

$$A = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta & t_x \\ \sin \theta & \cos \theta & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

$$A^{-1} = \frac{1}{\det A} \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ -t_y \sin \theta - t_x \cos \theta & -t_y \cos \theta + t_x \sin \theta & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ -t_y \sin \theta - t_x \cos \theta & -t_y \cos \theta + t_x \sin \theta & 1 \end{bmatrix}$$

3.18

$$P_1(r) = -r + r, P_2(z) = z$$

$$\oint_C S = T(r) = (L-1) \int_0^r P_1(u) du = (L-1)(-r^2 + r)$$

$$\oint_C S = G(z) = (L-1) \int_0^z P_2(w) dw = (L-1)z^2$$

$$\text{则 } \frac{S}{L-1} = -r^2 + r = z^2$$

$$\Rightarrow z = \pm \sqrt{-r^2 + r} \text{ (复数)}$$

$$\Rightarrow z = \sqrt{-r^2 + r}$$

3.18

(a)

$$f = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad W = \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

$$\begin{array}{c|c|c} 1 \times 0 & 2 \times 0 & 1 \times 0 \\ 2 \times 0 & 4 \times 1 & 2 \times 0 \\ 1 \times 0 & 2 \times 1 & 1 \times 0 \end{array} \Rightarrow 4 \times 1 + 2 \times 1 = 6$$

(b)

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \xrightarrow{\text{padding}} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$W * f \Rightarrow \begin{bmatrix} 0 & 1 & 2 & 1 & 0 \\ 0 & 3 & 6 & 3 & 0 \\ 0 & 4 & 8 & 4 & 0 \\ 0 & 3 & 6 & 3 & 0 \\ 0 & 1 & 2 & 1 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \xrightarrow{\text{padding}} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$W * f \Rightarrow \begin{bmatrix} 0 & 1 & 2 & 1 & 0 \\ 0 & 3 & 6 & 3 & 0 \\ 0 & 4 & 8 & 4 & 0 \\ 0 & 3 & 6 & 3 & 0 \\ 0 & 1 & 2 & 1 & 0 \end{bmatrix}$$

Part 2:

此次作業目標是設計一介面，供使用者對輸入的 jpeg 彩色影像操作灰階轉換（圖一）、比較、Threshold function 與調整其灰度（圖二）、亮度、對比度、解析度等。

一、程式介紹

本次作業共有七個小題，但內容大致為對輸入的彩色影像進行影像處理，並作出經處理灰階圖的 histogram，因此選擇將七個小題合併於一個介面內完成。此外，也加入了一些限制與功能讓程式更為完善，例如未選擇輸入影像即點按任一影像處理之功能鍵，會跳出“Please select an input image”警語，避免程式未讀檔即執行影像處理之迴圈（圖三）。

```
QImage OutputImage(1920,1080,QImage::Format_RGB32);

for(int j = 0; j < 1080; j++)
{
    for(int i = 0; i < 1920; i++)
    {
        QRgb pixcolors = propimage.pixel(i,j);
        r = qRed(pixcolors);
        g = qGreen(pixcolors);
        b = qBlue(pixcolors);

        grayvalueB= (int)(0.299*r+0.587*g+0.114*b);

        OutputImage.setPixel(i, j, qRgb(grayvalueB,grayvalueB,grayvalueB));

        countgrayvalue[grayvalueB]++;

    }
}

QPixmap OuputImage_pixmap = QPixmap::fromImage(OutputImage);
ui->output->setPixmap(OuputImage_pixmap.scaled(ui->output->width(),ui->output->height(),Qt::KeepAspectRatio));

MainWindow::MainWindow::showhistogram(countgrayvalue);
for(int i=0; i<256; i++ ) countgrayvalue[i] = 0;
```

圖一、將彩色輸入影像之 r、g、b 值依題目係屬組成灰階值輸出灰階影像

```
for(int j = 0; j < 1080; j++)
{
    for(int i = 0; i < 1920; i++)
    {
        QRgb pixcolors = propimage.pixel(i,j);
        r = qRed(pixcolors);
        g = qGreen(pixcolors);
        b = qBlue(pixcolors);

        grayvalueB= (int)(0.299*r+0.587*g+0.114*b);

        int newvalue= (round((grayvalueB*value/255.0f)))*255/value;

        OutputImage.setPixel(i, j, qRgb(newvalue,newvalue,newvalue));

        countgrayvalue[newvalue]++;

    }
}

QPixmap OuputImage_pixmap = QPixmap::fromImage(OutputImage);
ui->output->setPixmap(OuputImage_pixmap.scaled(ui->output->width(),ui->output->height(),Qt::KeepAspectRatio));

MainWindow::MainWindow::showhistogram(countgrayvalue);
for(int i=0; i<256; i++ ) countgrayvalue[i] = 0;
```

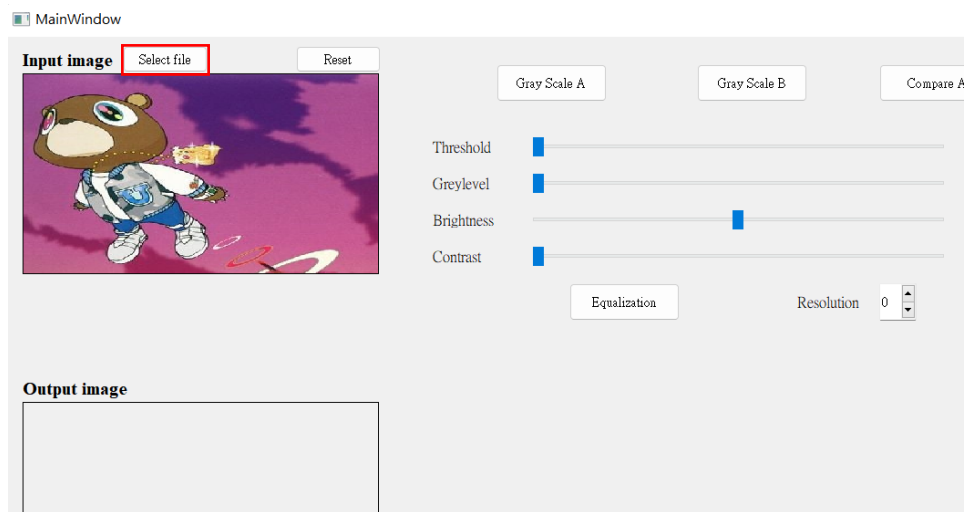
圖二、調整灰階影像之灰度（grayscale level）

```
應用程式輸出
HW2 x
QImage::pixel: coordinate (1367,61) out of range
QImage::pixel: coordinate (1368,61) out of range
QImage::pixel: coordinate (1369,61) out of range
QImage::pixel: coordinate (1370,61) out of range
QImage::pixel: coordinate (1371,61) out of range
QImage::pixel: coordinate (1372,61) out of range
QImage::pixel: coordinate (1373,61) out of range
QImage::pixel: coordinate (1374,61) out of range
QImage::pixel: coordinate (1375,61) out of range
QImage::pixel: coordinate (1376,61) out of range
QImage::pixel: coordinate (1377,61) out of range
QImage::pixel: coordinate (1378,61) out of range
QImage::pixel: coordinate (1379,61) out of range
QImage::pixel: coordinate (1380,61) out of range
QImage::pixel: coordinate (1381,61) out of range
QImage::pixel: coordinate (1382,61) out of range
QImage::pixel: coordinate (1383,61) out of range
QImage::pixel: coordinate (1384,61) out of range
QImage::pixel: coordinate (1385,61) out of range
QImage::pixel: coordinate (1386,61) out of range
QImage::pixel: coordinate (1387,61) out of range
QImage::pixel: coordinate (1388,61) out of range
QImage::pixel: coordinate (1389,61) out of range
QImage::pixel: coordinate (1390,61) out of range
QImage::pixel: coordinate (1391,61) out of range
```

圖三、未加入限制前，若未選擇輸入檔案即執行功能，程式將空跑迴圈。

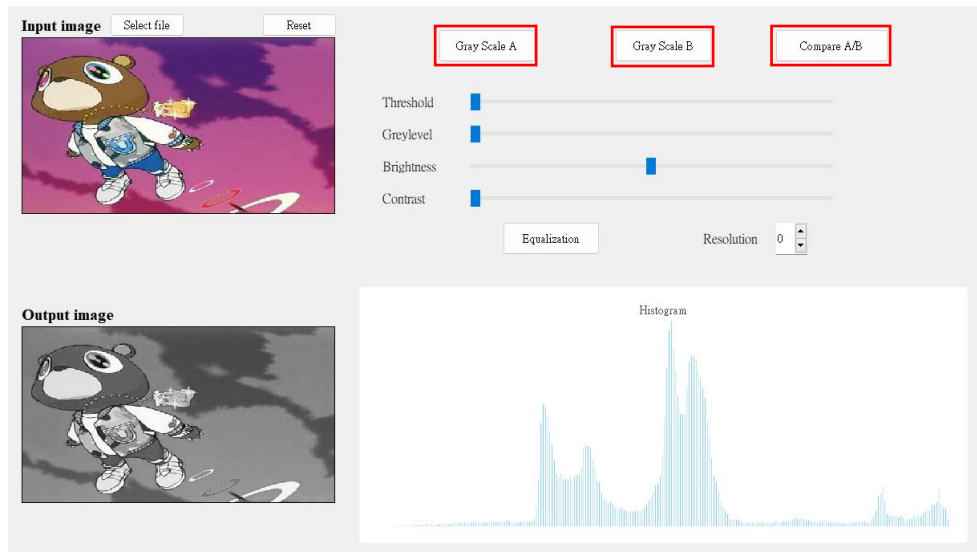
二、UI 介紹

1. 點選介面左上角的“Select file”按鈕即可選擇欲輸入之 jpeg 影像，選擇後輸入影像即呈現於“Input image”顯示窗（圖四）。



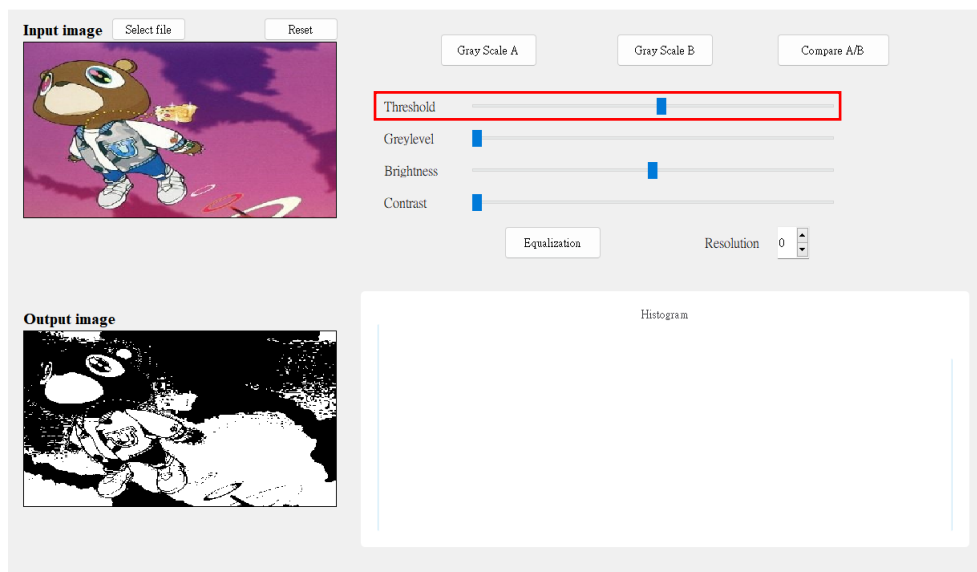
圖四

2. 選擇介面右上角之“GrayScale A”或“GrayScale B”按鈕即可依題意分別將輸入彩圖轉換成兩種不同的灰階影像，呈現於“Output image”顯示窗，並在介面右下角顯示其 histogram（圖五）。點選一旁之“Compare A/B”按鈕即可將兩灰階影像相減，以觀察兩灰階影像之差異。



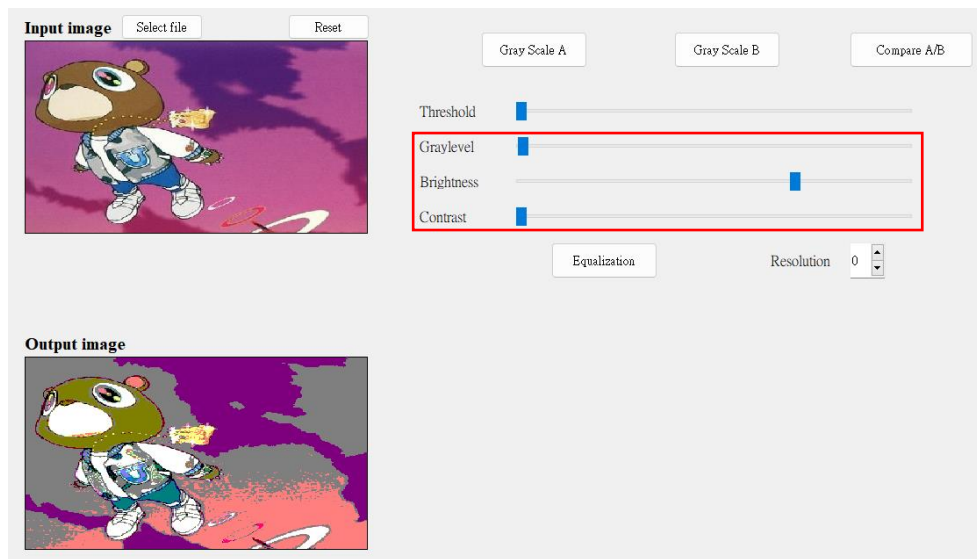
圖五

3. 拉動“Threshold”拉桿可以調整鄰界數值，將 Grayscale image A、Grayscale image B 或者兩圖之差（Compare A/B）等灰階影像轉成二元影像（圖六）。



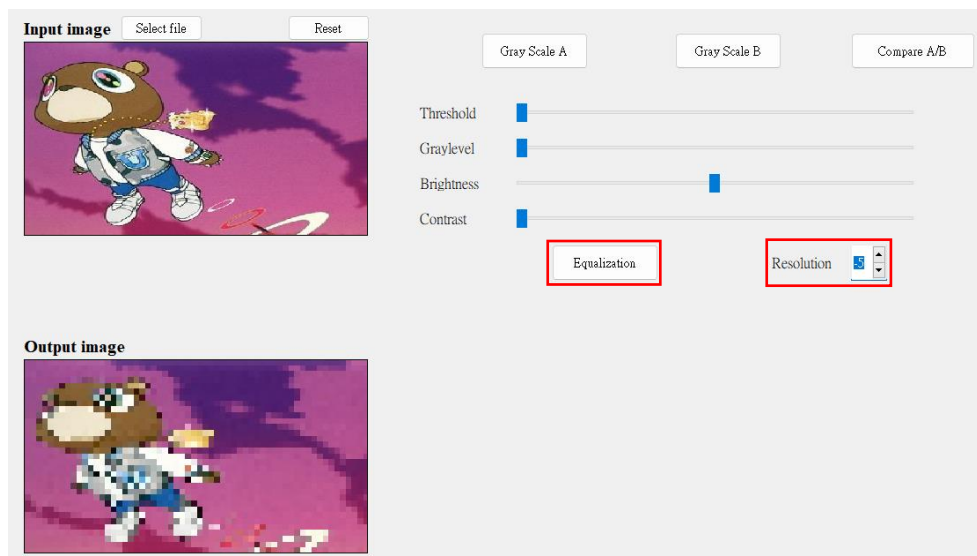
圖六

4. 拉動“Graylevel”、“Brightness”與“Contrast”等拉桿能對輸入彩圖、Grayscale image A、Grayscale image B 分別調整灰度（圖七）、亮度與對比度。



圖七

5. 點按“Equalization”按鈕能對輸入彩圖、Grayscale image A、Grayscale image B 進行等化處理。變動“Resolution”之數值能調整輸入彩圖之空間解析度（圖八）。可調動數值範圍為-5~3，0 表原圖，數值愈高解析度愈高，數值愈低則解析度愈低。



圖八

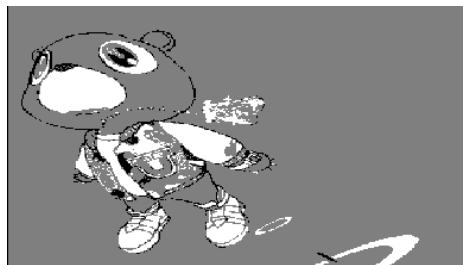
三、結果與討論

1. 若 Grayscale image A、Grayscale image B 兩張灰階影像相近，執行“Compare A/B”時將不易看出兩者差距。但若通過 histogram 先觀察兩圖相減後最多的灰階值為何，再透過“Threshold”拉桿將臨界值調整為該數值，便能輕易看出兩灰階影像差異之處（圖九）。



圖九

2. 灰度在數值低時，每單位的變化都能輕易看出輸出圖像的差距。然而，隨著灰度越來越高，每單位灰度的變化將對輸出圖像影響愈來愈小（圖十）。說明人的眼睛只能分辨某一程度以上的灰階落差，灰階落差愈小，人眼愈難辨識其差異。



$n=3$



$n=4$



$n=128$



$n=129$

圖十

3. 若顯示窗維持一定大小，調高輸入圖像之空間解析度並無法看出差異；調低空間解析度則如同將原圖經馬賽克處理（圖十一）。



$r=3$



$r=0$



$r=-5$

圖十一