บทที่ 3

การออกแบบและพัฒนาซอฟต์แวร์

3.1 โครงสร้างข้อมูล

ข้อมูลนำเข้าระบบเป็นข้อมูลภาพดิจิทัล 2 ขนาด คือ ภาพสีในแบบจำลองสีอาร์จีบีในแฟ้มข้อมูลภาพแบบบิตแมพขนาด 256x256 จุดภาพ มีขนาดเท่ากับ (256x256x3)+54 หรือ 196,662 ไบต์ เลข 3 หมายถึง จำนวนแม่สีในจุดภาพ คือ สีแดง สีเขียว และสีน้ำเงิน เลข 54 หมายถึง ขนาดเฮดเดอร์ของแฟ้มข้อมูลภาพแบบบิตแมพ และภาพสีในแบบจำลองสีอาร์จีบีในแฟ้มข้อมูลภาพแบบบิตแมพขนาด 512x512 จุดภาพ มีขนาดเท่ากับ (512x512x3)+54 หรือ 786,486 ไบต์

การนำข้อมูลเข้าระบบในแต่ละครั้งใช้ภาพ 2 ภาพ คือ ภาพต้นฉบับและภาพที่ผ่านการบีบซึ่งเป็นภาพขนาดเดียวกันเพื่อนำมาเปรียบเทียบกัน ส่วนข้อมูลออกจากระบบจะเป็นค่ามาตราส่วนคุณภาพภาพและภาพที่แสดงความผิดเพี้ยนจากปัจจัยความผิดเพี้ยนทั้ง 5 จำนวน 5 ภาพ

3.2 ส่วนประกอบที่สำคัญของโปรแกรม

การเปรียบเทียบการบีบภาพระหว่างเทคนิคเจเพ็กและเวฟเลตใช้โปรแกรมประมวลผล 2 โปรแกรม คือ โปรแกรมการเปลี่ยนภาพสีเป็นภาพเกรย์สเกล และโปรแกรมมาตราส่วนคุณภาพภาพ

3.2.1 โปรแกรมการเปลี่ยนภาพสีเป็นภาพเกรย์สเกล

โปรแกรมการเปลี่ยนภาพสีจะเปลี่ยนแฟ้มข้อมูลบิตแมพโดยเลือกขอบเขตภาพสี 24 บิตเท่านั้น ให้เป็นภาพเกรย์สเกลซึ่งเป็นแฟ้มข้อมูลแบบรอ 8 บิต แฟ้มข้อมูลภาพแบบบิตแมพจะมีส่วนหัวของแฟ้มข้อมูลหรือบิตแมพไฟล์เฮดเดอร์ ขนาด 14 ไบต์ และส่วนหัวของข้อมูลภาพหรือบิตแมพอินโฟเฮดเดอร์ ขนาด 40 ไบต์ ซึ่งมีโครงสร้างดังต่อไปนี้

```
struct bitmapfileheader
 {
        char bftype[2];
        unsigned long bfsize;
        char bfreserved1[2];
        char vfreserved2[2];
        unsigned long bfoffbits;
 } bmfh;
struct bitmapinfoheader
{
        unsigned long bisize;
        unsigned long biwidth;
        unsigned long biheight;
        unsigned int biplanes;
        unsigned int bibitcount;
        unsigned long bicompression;
        unsigned long bisizeimage;
        unsigned long bixpelspermeter;
        unsigned long biypelspermeter;
        unsigned long biclrused;
        unsigned long biclrimportant;
} bmih;
```

ส่วนต่อไปเป็นข้อมูลภาพซึ่งจะเรียงลำดับค่าสีจากสีน้ำเงิน สีเขียว และสีแดงตามลำดับ เมื่อจุดภาพมีขนาด 24 ไบต์ต่อหนึ่งจุดภาพ

```
struct rgbquad
{
        unsigned char rgbblue;
        unsigned char rgbgreen;
        unsigned char rgbred;
} rgb;
```

สูตรในการคำนวณค่าจุดภาพเป็นดังต่อไปนี้

ค่าจุดภาพใดๆ ในภาพ = (0.299*rgb.rgbred) + (0.587*rgb.rgbgreen) +
(0.114*rgb.rgbblue);

สมการดังกล่าวเป็นสมการการเปลี่ยนจากภาพสีเป็นภาพเกรย์สเกล มีรูปแบบจากการเปลี่ยนแบบจำลองสีอาร์จีบีเป็นแบบจำลองสีวายไอคิว โดย Y คือ ลูมิแนนซ์ หรือค่าความส่องสว่าง ซึ่งจะเกิดจากการผสมค่าสีแดง, เขียว และน้ำเงินในสัดส่วนต่างๆ กัน

3.2.2 โปรแกรมมาตราส่วนคุณภาพภาพ

โปรแกรมมาตราส่วนคุณภาพภาพจะนำผลลัพธ์จากโปรแกรมการเปลี่ยนภาพสีให้เป็นภาพเกรย์สเกลมาประมวลผล ซึ่งผลลัพธ์ที่ได้จากโปรแกรมการเปลี่ยนภาพสีให้เป็นภาพเกรย์สเกลจะเป็นภาพเกรย์สเกลที่แต่ละจุดภาพจะใช้ 8 บิต ไม่มีเฮดเดอร์ของแฟ้มข้อมูล ดังนั้นจะเป็นข้อมูลภาพเท่านั้น เมื่อผ่านการประมวลผลโดยโปรแกรมมาตราส่วนคุณภาพภาพ จะให้ผลลัพธ์เป็นภาพที่แสดงความผิดเพี้ยนของภาพ 5 ภาพซึ่งเป็นภาพแบบรอ (Raw) กล่าวคือ ไม่มีเฮดเดอร์ของแฟ้มข้อมูล เป็นข้อมูลภาพเพียงอย่างเดียว และค่าที่ได้จากการคำนวณค่ามาตราส่วนคุณภาพภาพคือค่ามาตราส่วนคุณภาพภาพ

ปัจจัยความผิดเพี้ยนที่ 1

/* คำนวณค่าผิดพลาดระหว่างภาพต้นฉบับและภาพที่บีบ $e_i(x,y)$ และผลรวมของค่าจุดภาพในภาพต้นฉบับ $\sum i^2(x,y)$ */

for (y = 0; y < ความสูงของภาพ; y++) {
for (x = 0; x < ความกว้างของภาพ; x++)

$$e_i(x,y) = i(x,y) - \hat{i}(x,y);$$

$$\sum i^2(x,y) = i(x,y) * i(x,y);$$

}

/* คำนวณค่าความผิดเพี้ยนในแต่ละจุดภาพของภาพ $f_1(x,y)$ และผลรวมของค่าความผิดเพี้ยนในแต่ละจุดภาพ $\sum f_1(x,y)$ */

for (y = 0; y < ความสูงของภาพ; y++) {

for (x = 0; x < ความกว้างของภาพ; x++)

$f_1(x,y) = [e_i(x,y) * W_{TV}(x,y)]^2$;

$\sum f_1(x,y) = f_1(x,y) + f_1(x,y)$;

}

ดังนั้นค่าแสดงความผิดพลาดของปัจจัยความผิดเพี้ยนที่ 1 $Factor[1] = \frac{\sum f_1(x,y)}{\sum i^2(x,y)}$

สำหรับการคำนวณ W_{TV} ที่จุดภาพใดๆ มีดังนี้

ให้ $f_c = 5.56$;

for (y = 0; y < ความสูงของภาพ; y++) {

if (y > ความสูงของภาพ/2) v = ความสูงของภาพ – y;

else v = y;

for (x = 0; x < ความกว้างของภาพ; x++)

if (x > ความกว้างของภาพ/2) u = ความกว้างของภาพ – x;

else u = x;

$f = \sqrt{u^2 + v^2}$;

$W_{TV}(x,y) = \frac{1}{1 + \left(\frac{f}{f_c}\right)^2}$;

}

ปัจจัยความผิดเพี้ยนที่ 2

/* คำนวณค่าผิดพลาดที่ผ่านการปรับคอนแทรสท์ระหว่างภาพต้นฉบับและภาพที่บีบ $e_w(x,y)$ และผลรวมของค่าจุดภาพที่ผ่านการปรับคอนแทรสท์ในภาพที่บีบ $\sum \hat{i}^2(x,y)$ */

for (y = 0; y < ความสูงของภาพ; y++) {

for (x = 0; x < ความกว้างของภาพ; x++)

$e_w(x,y) = contrast\ (i(x,y)) - contrast\ (\hat{i}(x,y))$;

$\sum \hat{i}^2(x,y) = contrast\ (\hat{i}(x,y)) * contrast\ (\hat{i}(x,y))$;

}

/* คำนวณค่าความผิดเพี้ยนในแต่ละจุดภาพของภาพ $f_2(x,y)$ และผลรวมของค่าความผิดเพี้ยนในแต่ละจุดภาพ $\sum f_2(x,y)$ */

for (y = 0; y < ความสูงของภาพ; y++) {

for (x = 0; x < ความกว้างของภาพ; x++)

$$f_2(x,y) = I_T(x,y)[e_w(x,y) * Sa(x,y)];$$

$$\sum f_2(x,y) = f_2(x,y) + f_2(x,y);$$

}

ดังนั้นค่าแสดงความผิดพลาดของปัจจัยความผิดเพี้ยนที่ 2 $Factor[2] = \dfrac{\sum f_2(x,y)}{\sum \hat{i}^2(x,y)}$

สำหรับการคำนวณ Sa ที่จุดภาพใดๆ มีดังนี้

ให้ $\alpha = 2, \beta = 8, f_0 = 11.1294$; และ $w_0 = \dfrac{2\pi f_0}{60}$;

for (y = 0; y < ความสูงของภาพ; y++) {

if (y > ความสูงของภาพ/2) v = ความสูงของภาพ – y;

else v = y;

for (x = 0; x < ความกว้างของภาพ; x++)

if (x > ความกว้างของภาพ/2) u = ความกว้างของภาพ – x;

else u = x;

$$f = \sqrt{u^2 + v^2};$$

$$w = \frac{2\pi f}{60};$$

$$Sa(x,y) = \left(1.5e^{\frac{-\alpha^2 w^2}{2}} - e^{-2\alpha^2 w^2}\right) * \left(\frac{1 + e^{\beta(w-w_0)}\cos^4 2\theta}{1 + e^{\beta(w-w_0)}}\right);$$

}

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

$contrast(i(x,y))$ และ $contrast(\hat{i}(x,y))$ คือ การปรับคอนแทรสท์ในแต่ละค่าจุดภาพ ด้วยสมการ (ค่าจุดภาพ * $772.4105847^{\frac{1}{2.2}}$)

ปัจจัยความผิดเพี้ยนที่ 3

/* ความแตกต่างในแกนแนวนอนในขอบเขตของบล็อกตามแนวตั้ง */

for (y = 0; y < ความสูงของภาพ; y++) {

for (x = ขนาดของบล็อกซึ่งเท่ากับ 8; x < ความกว้างของภาพ; x+=ขนาดของบล็อก)

$$\Delta_h(x,y) = e_w(x,y) - e_w(x,y+1);$$
$$f_{3h}(x,y) = I_h(x,y) * \Delta_h(x,y) * \Delta_h(x,y);$$
$$\sum f_{3h}(x,y) = f_{3h}(x,y) + f_{3h}(x,y);$$
$$N_h = N_h + 1;$$

}

$$F_{3h} = \frac{\sum f_{3h}(x,y)}{N_h};$$

/* ความแตกต่างในแกนแนวตั้งในขอบเขตของบล็อกตามแนวนอน */

for (y = ขนาดของบล็อก; y < ความสูงของภาพ; y+=ขนาดของบล็อก) {

for (x = 0; x < ความกว้างของภาพ; x++)

$$\Delta_v(x,y) = e_w(x,y) - e_w(x+1,y);$$
$$f_{3v}(x,y) = I_v(x,y) * \Delta_v(x,y) * \Delta_v(x,y);$$
$$\sum f_{3v}(x,y) = f_{3v}(x,y) + f_{3v}(x,y);$$
$$N_v = N_v + 1;$$

}

$$F_{3v} = \frac{\sum f_{3v}(x,y)}{N_v};$$

ดังนั้นค่าแสดงความผิดพลาดของปัจจัยความผิดเพี้ยนที่ 3 $Factor[3] = \sqrt{F_{3h}^2 + F_{3v}^2}$

ปัจจัยความผิดเพี้ยนที่ 4

/* คำนวณ local error correlations ซึ่งเป็นหน้าต่างขนาด 5x5 จุดภาพ และมีจุดศูนย์กลางที่จุดภาพ (m,n) และคำนวณค่าความผิดเพี้ยนในแต่ละจุดภาพของภาพ $f_4(x,y)$ ให้ขนาดครึ่งหน้าต่างเท่ากับ 2, $xLag(k), yLag(l) <= 2$ และ $Max_Lag = 2$ */

for (y = 0; y < ความสูงของภาพ; y++) {
  for (x = 0; x < ความกว้างของภาพ; x++)
    for (yLag = -Max_Lag; yLag <= Max_Lag; yLag++)
      for (xLag = 0; xLag <= Max_Lag; xLag++)
        if (xLag > 0 หรือ yLag > 0) {
          xMin = max (x – ขนาดครึ่งหน้าต่าง, 0) – min (0, xLag);
          xMax = min (x + ขนาดครึ่งหน้าต่าง, ความกว้าง - 1) – max (0, xLag);
          yMin = max (y – ขนาดครึ่งหน้าต่าง, 0) – min (0, yLag);
          yMax = min (y + ขนาดครึ่งหน้าต่าง, ความสูง - 1) – max (0, yLag);
          for (y1 = yMin, y2 = yMin+yLag; y1 <= yMax; y1++, y2++)
            for (x1 = xMin, x2 = xMin+xLag; x1 <= xMax; x1++, x2++) {
              $n++;$
              $$r(m,n,k,l) = \frac{\left[e_w(x1,y1) * e_w(x2,y2) - \frac{e_w(x1,y1) * e_w(x2,y2)}{n} \right]}{n-1};$$
            }
          $f_4(x,y) = r(m,n,k,l)^{0.25};$
        }
}

/* คำนวณผลรวมของค่าความผิดเพี้ยนในแต่ละจุดภาพ $\sum f_4(x,y)$ */

for (y = 0; y < ความสูงของภาพ; y++) {
  for (x = 0; x < ความกว้างของภาพ; x++)
    $$\sum f_4(x,y) = |f_4(x,y)| + |f_4(x,y)|;$$
}

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ดังนั้นค่าแสดงความผิดพลาดของปัจจัยความผิดเพี้ยนที่ 4 $Factor[4] = \frac{\sum f_4(x,y)}{MN}$

โดยที่ M คือ ความสูงของภาพเป็นจุดภาพ และ N คือ ความกว้างของภาพเป็นจุดภาพ

ปัจจัยความผิดเพี้ยนที่ 5

/* คำนวณค่าความผิดเพี้ยนในแต่ละจุดภาพของภาพ $f_5(x,y)$ และผลรวมของค่าความผิดเพี้ยนในแต่ละจุดภาพ $\sum f_5(x,y)$ */

for (y = 0; y < ความสูงของภาพ; y++) {

for (x = 0; x < ความกว้างของภาพ; x++)

if $KirschEdge(x,y)$ >= ค่าขีดแบ่งขอบ (Edge threshold) ซึ่งเท่ากับ 400

$N_k = N_k + 1;$

$f_5(x,y) = I_m(x,y) * |e_w(x,y)| * e^{\frac{-0.04|i(x,y-1)-i(x,y+1)|}{2}} * e^{\frac{-0.04|i(x-1,y)-i(x+1,y)|}{2}};$

$\sum f_5(x,y) = f_5(x,y) + f_5(x,y);$

}

ดังนั้นค่าแสดงความผิดพลาดของปัจจัยความผิดเพี้ยนที่ 5 $Factor[5] = \frac{\sum f_5(x,y)}{N_k}$

สำหรับการคำนวณ *KirschEdge* ที่จุดภาพใดๆ มีดังนี้

for (y = 1; y < ความสูงของภาพ; y++) {

for (x = 1; x < ความกว้างของภาพ; x++)

$u[0] = i(x-1, y-1);$

$u[1] = i(x, y-1);$

$u[2] = i(x+1, y-1);$

$u[3] = i(x+1, y);$

$u[4] = i(x+1, y+1);$

$u[5] = i(x, y+1);$

$u[6] = i(x-1, y+1);$

$u[7] = i(x-1, y);$

for (j = 1, $MaxValue = 0$; j < 9; j++) {

$$value = u[j-1] + u[j\%8] + (u[j+1]\%8);$$

if $value > MaxValue$

$$MaxValue = value;$$

}

for (k = 0; k < 8; k++)

$$KirschEdge(x,y) = KirschEdge(x,y) + u[k];$$

$$KirschEdge(x,y) = -3 * KirschEdge(x,y) + 8 * MaxValue;$$

}

3.3 การทำงานของโปรแกรมเพื่อการเปรียบเทียบการบีบอัดภาพ

การทำงานของโปรแกรมเพื่อการเปรียบเทียบการบีบอัดภาพระหว่างเทคนิคเจเพ็กและเวฟเลตใช้โปรแกรมประมวลผล 2 โปรแกรม คือ โปรแกรมการเปลี่ยนภาพสีเป็นภาพเกรย์สเกล และโปรแกรมมาตราส่วนคุณภาพภาพ

3.3.1 การทำงานของโปรแกรมการเปลี่ยนภาพสีเป็นภาพเกรย์สเกล มีพารามิเตอร์ 2 ตัว ดังต่อไปนี้

c:\rgb2g source_image gray_image

ตัวอย่าง c:\rgb2g i.bmp i.raw

rgb2g **คือ** ชื่อโปรแกรม source_image **คือ** ภาพต้นฉบับ gray_image **คือ** ภาพผลลัพธ์ของโปรแกรมซึ่งเป็นภาพเกรย์สเกล ดังนั้นจากตัวอย่าง การทำงานของโปรแกรม คือ การเปลี่ยนภาพสี 24 บิตแฟ้มข้อมูลแบบบีเอ็มพี i.bmp เป็นภาพเกรย์สเกล 8 บิตแฟ้มข้อมูลแบบรอ i.raw

3.3.2 การทำงานของโปรแกรมมาตราส่วนคุณภาพภาพ มีพารามิเตอร์ 2 ชุด ใช้คำสั่งต่อไปนี้

c:\compare_jw [-s size_of_image] [-f prefix_output_image source_image compressed_image]

ตัวอย่าง compare_jw -s 512 -f distorted_i i.raw compressed_i.raw

compare_jw คือ ชื่อโปรแกรม [-s size_of_image] คือ พารามิเตอร์ชุดแรก size_of_image คือ ขนาดของจุดภาพต่อหนึ่งแฟ้มข้อมูล จากการวิจัยมี 2 ขนาด คือ 256x256 และ 512x512 จุดภาพ มีเงื่อนไขว่า ภาพต้นฉบับและภาพที่บีบอัดจะต้องมีขนาดเท่ากัน [-f prefix_output_image source_image compressed_image] คือ พารามิเตอร์ชุดสอง prefix_output_image คือ คำนำหน้าที่จะนำไปใส่ในภาพผลลัพธ์ source_image คือ ภาพต้นฉบับ compressed_image คือ ภาพที่บีบอัดและคลายแล้วเพื่อเปรียบเทียบกับภาพต้นฉบับ จากตัวอย่างการทำงานของโปรแกรม คือ การเปรียบเทียบคุณภาพของภาพที่บีบอัดและคลายแล้ว (compressed_i.raw) กับภาพต้นฉบับ (i.raw) ขนาด 512x512 จุด ผลลัพธ์ของการทำงานเป็นแฟ้มข้อมูลภาพที่แสดงความผิดเพี้ยน 5 ภาพ (Distorted images) เช่น distorted_i.factor1, distorted_i. factor 2, distorted_i. factor 3, distorted_i. factor4, distorted_i. factor5 เป็นต้น และค่าแสดงคุณภาพของภาพ (PQS) เช่น

factor1: [0.000000,785.158235]

factor1: 0.0006339536, weighted: 0.00002219

factor2: [0.000000,1380.204359]

factor2: 0.0002842917, weighted: 0.00001251

factor3: [0.000000,528.056888]

factor3: 14.5667908909, weighted: 0.14566791

factor4: [3.328094,39.820730]

factor4: 11.1716595172, weighted: -1.47465906

factor5: [0.000000,66.229918]

factor5: 29.7721120366, weighted: -4.01923512

PQS = 3.09616712

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย