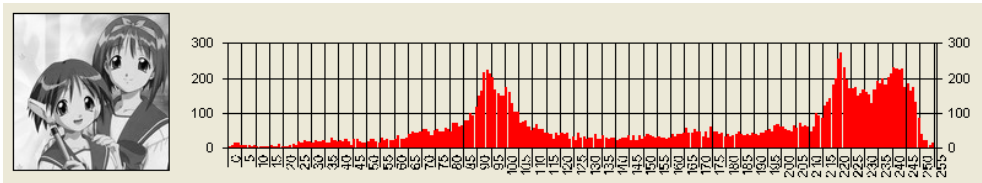


I. TUJUAN :

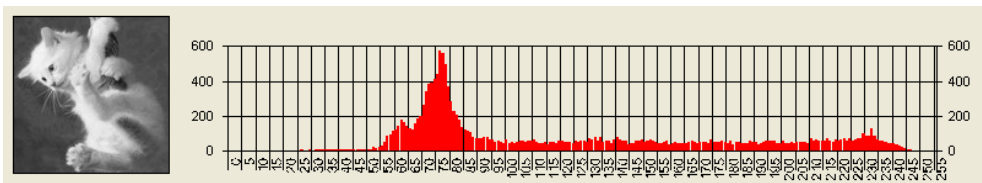
1. Mahasiswa dapat memahami prinsip-prinsip color histogram pada suatu image
2. Mahasiswa dapat menggunakan proses histogram untuk mengetahui suatu gambar yang memiliki warna – warna tertentu
3. Mahasiswa dapat mendeteksi suatu warna dengan melihat color histogram pada setiap citra.

II. DASAR TEORI :**HISTOGRAM dalam gambar Gray Scale**

- Histogram di dalam gambar gray-scale menyatakan distribusi dari derajat keabuan (terang/gelap) pada suatu gambar.
- Dari histogram ini dapat dilihat apakah gambar tersebut lebih banyak warna gelap atau lebih banyak warna terang
- Teknik histogram ini dapat dikembangkan untuk memperbaiki kualitas gambar (image enhancement) dengan apa yang dinamakan dengan **Histogram**



Gambar ini didominasi warna terang, karena grafik di sebelah kanan terlihat lebih banyak.

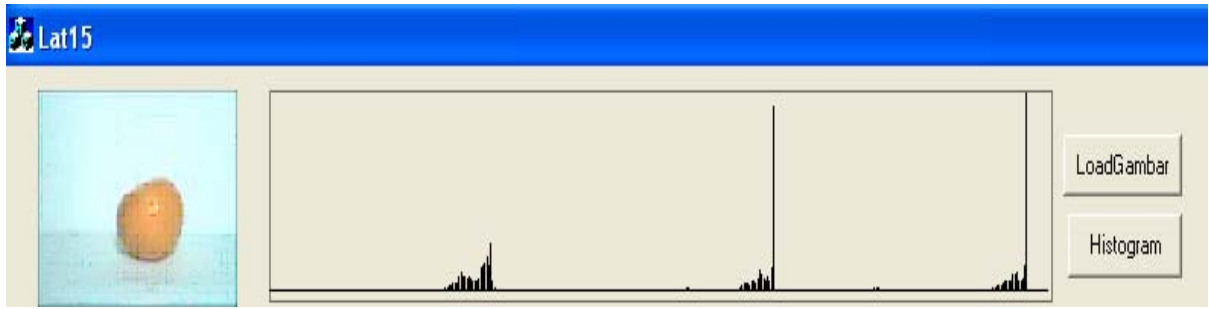


Gambar ini didominasi warna gelap, karena grafik di sebelah kiri terlihat lebih banyak.

Color Histogram

- Color Histogram adalah suatu proses untuk meratakan histogram agar derajat keabuan dari yang paling rendah (0) sampai dengan yang paling tinggi (255) mempunyai kemunculan yang rata.

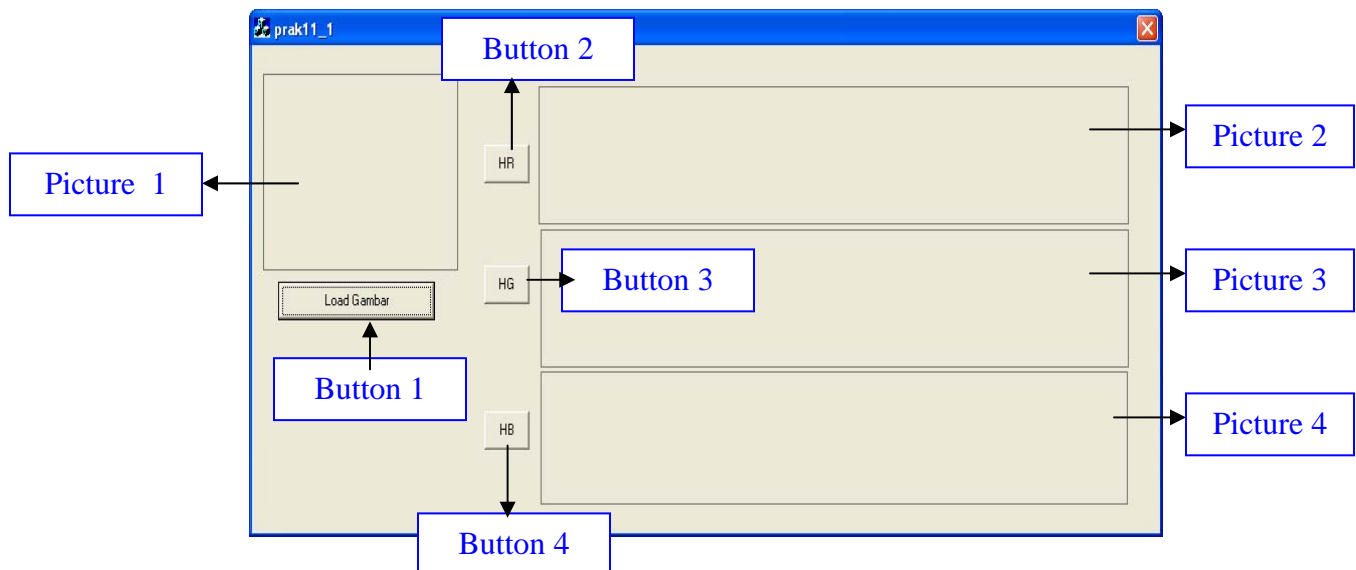
- Dengan histogram ini hasil gambar yang memiliki histogram yang tidak merata yang banyak loncatan gradiasinya akan menjadi gambar yang lebih jelas karena derajat keabuannya tidak dominan gelap atau dominan terang.



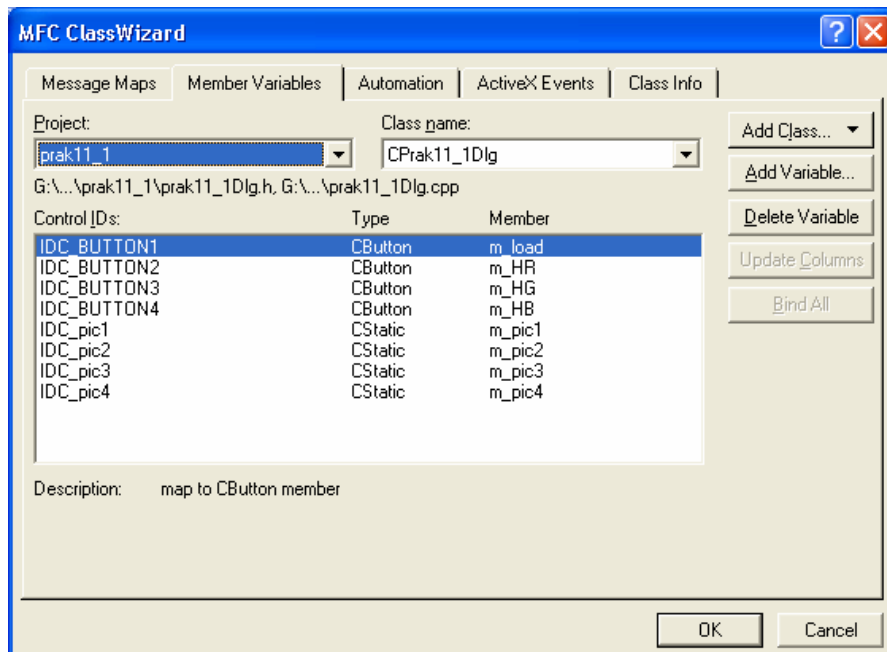
Contoh Gambar Color Histogram pada Buah Jeruk

III.HASIL DAN ANALISA PERCOBAAN :

1. Membuat suatu project color histogram dengan mendeteksi warna RGB pada suatu citra yaitu secara terpisah setiap picture pada histogram Red,Green, dan Blue.
Langkah – langkah membuat project diatas adalah sebagai berikut :
 - a. Buat aplikasi AppWizard seperti pada praktikum sebelumnya dan beri nama project dengan prak11_1
 - b. Buat disain dialog dengan menggunakan 4 buah control picture, dan 4 buah button.



- c. Merubah properti control dan tambahkan class variabel control tipe dan membernya seperti berikut :



- d. Tambahkan program pada saat button1 click seperti di bawah ini, akan melakukan load gambar dalam kotak control picture dengan cara melakukan browse gambar, pilih gambar yang akan diproses.

```
void CPrak11_1Dlg::OnButton1()
{
    // TODO: Add your control notification handler code here
    static char BASED_CODE szFilter[]="Bitmap Files (*.bmp)|*.bmp| |";

    CFileDialog m_ldFile(TRUE, "*.bmp", name, OFN_HIDEREADONLY|OFN_OVERWRITEPROMPT, szFilter);
    if(m_ldFile.DoModal()==IDOK)
    {
        name=m_ldFile.GetPathName();
    }

    CDC* pDC = m_pic1.GetDC();// mengarah kontrol picture
    CDC dcMem; //mengarahkan obyek yg sdh dipilih pd bitmap
    CRect rect;//kotak di picture
    BITMAP bm;//mendefinisikan fungsi2 yg ada d bitmap
    HBITMAP
    hBitmap=(HBITMAP)::LoadImage(AfxGetInstanceHandle(), name, IMAGE_BITMAP, 0, 0, LR_LOADFROMFILE|LR_CREATEDIBSECTION);
    if(hBitmap)
    {
        if(m_bmpBitmap.DeleteObject())
            m_bmpBitmap.Detach();
        m_bmpBitmap.Attach(hBitmap);
    }
    m_pic1.GetClientRect(rect);//
    m_bmpBitmap.GetBitmap(&bm);//
    dcMem.CreateCompatibleDC(pDC);
    //pDC->MoveTo(10,190);
    //pDC->LineTo(300,190);
    dcMem.SelectObject(&m_bmpBitmap);
    pDC->StretchBlt(0,0,rect.Width(),rect.Height(),&dcMem, 0,0,bm.bmWidth,bm.bmHeight,SRCCOPY);
}
}
```

- e. Menambah header file

- Buka file prak11_1Dlg.h
- Tambahkan program seperti dibawah ini

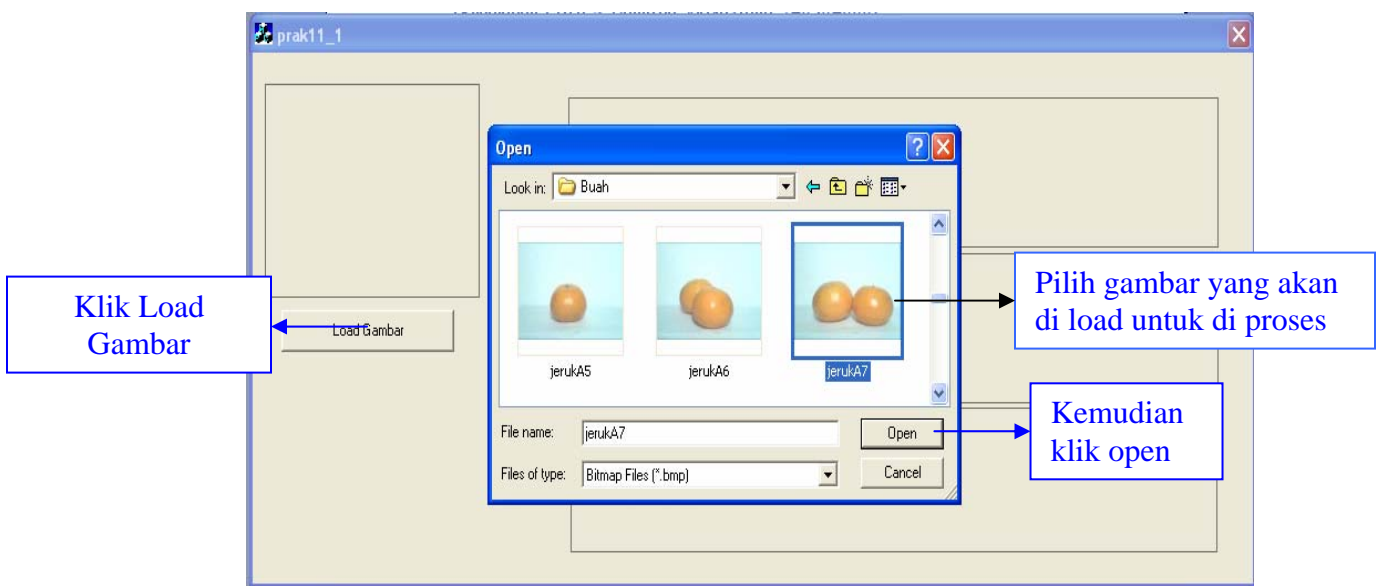
public:

```
CPrak10_1Dlg(CWnd* pParent = NULL); // standard constructor
```

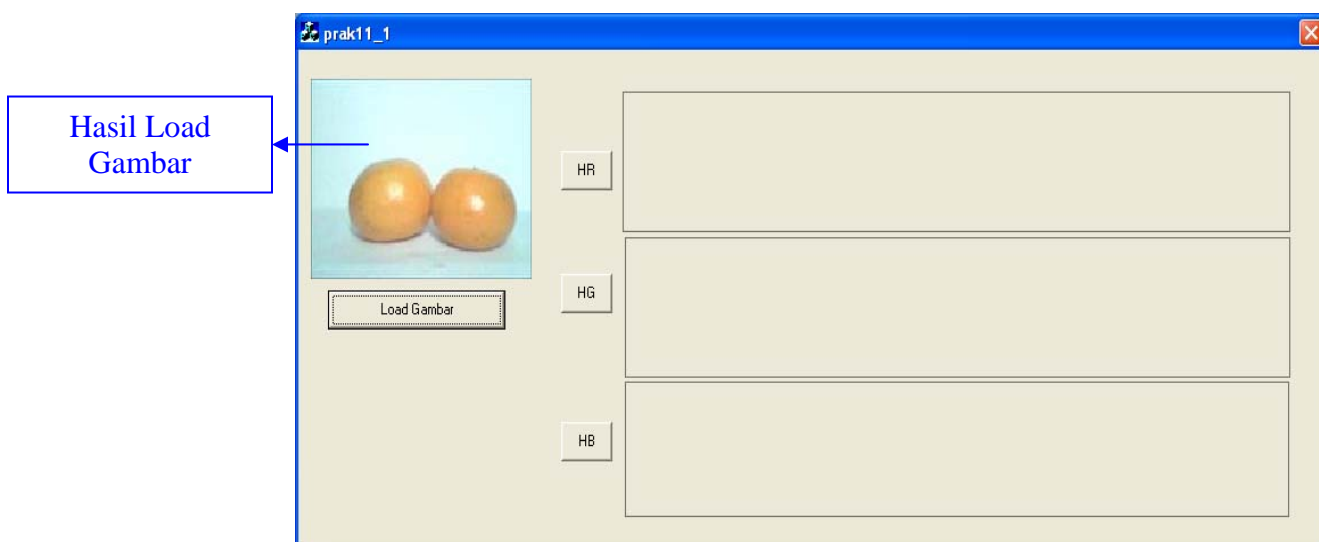
```
CBitmap m_bmpBitmap; → Load Gambar disimpan pada var m_bmpBitmap
```

```
CString name; → Variable temp untuk gambar
```

Saat Load Gambar di klik, maka hasilnya akan seperti berikut ini :



Setelah di klik open :



- f. Tambahkan program pada saat button2 click seperti di bawah ini, akan menampilkan histogram untuk warna red :

```

void CPrak11_1Dlg::OnButton2()
{
    // TODO: Add your control notification handler code here
    int i,j,red,green,blue;
    long int warna;
    CDC* pDC = m_pic1.GetDC();// mengarah kontrol picture
    CDC dcMem1; //mengarahkan obyek yg sdh dipilih pd bitmap
    CRect rect;//kotak di picture
    BITMAP bm;//mendefinisikan fungsi2 yg ada d bitmap
    int hr[256];
    HBITMAP
    hBitmap=(HBITMAP)::LoadImage(AfxGetInstanceHandle(),
    name,IMAGE_BITMAP, 0, 0,
    LR_LOADFROMFILE|LR_CREATEDIBSECTION);
    if(hBitmap)
    {
        if(m_bmpBitmap.DeleteObject())
            m_bmpBitmap.Detach();
        m_bmpBitmap.Attach(hBitmap);
    }
    m_pic1.GetClientRect(rect);//
    m_bmpBitmap.GetBitmap(&bm);//

    dcMem1.CreateCompatibleDC(pDC);
    dcMem1.SelectObject(&m_bmpBitmap);

    for(i=0;i<256;i++)
        hr[i]=0;
    for(i=0;i<bm.bmHeight;i++)
        for(j=0;j<bm.bmWidth;j++)
        {
            warna=dcMem1.GetPixel(j,i);
            WarnaToRGB(warna,&red,&green,&blue);

            hr[red]=hr[red]+1;
        }
    float hmax=hr[0];
    for(i=1;i<256;i++)
    {
        if(hr[i]>hmax)
            hmax=hr[i];
    }
    for(i=0;i<256;i++)
        hr[i]=120*hr[i]/hmax;

    CDC* pDC1=m_pic2.GetDC();
    pDC1->MoveTo(0,100);
    pDC1->LineTo(550,100); //sb x
    pDC1->MoveTo(0,100-(int)hr[0]); //ttk h[0]
    for(i=1;i<256;i++)
    {
        pDC1->MoveTo(i*2,100);
        pDC1->LineTo(i*2,100-(int)hr[i]);
    }
}

```

Menampilkan bagian warna merah saja pada gambar histogramnya

Fungsi untuk menggambar histogram

- g. Tambahkan fungsi untuk merubah data pixel ke RGB, dan fungsi untuk merubah dari RGB ke data pixel

```
// merubah data pixel ke RGB
```

```

void WarnaToRGB(long int warna,int *Red, int
*Green, int *Blue)
{
    *Red = warna & 0x000000FF;
    *Green = (warna & 0x0000FF00) >> 8;
    *Blue = (warna & 0x00FF0000) >> 16;
}

```

Fungsi untuk membaca warna RGB

```

//merubah RGB ke data pixel
long int RGBToWarna(int Red, int Green, int Blue)
{
    return(Red+(Green<<8)+(Blue<<16));
}

```

Fungsi untuk membuat warna RGB

- h. Tambahkan program pada saat button3 click seperti di bawah ini, akan menampilkan histogram untuk warna hijau :

```

void CPrak11_1Dlg::OnButton3()
{
    // TODO: Add your control notification handler code here
    int i,j,red,green,blue;
    long int warna;
    CDC* pDC = m_pic1.GetDC();// mengarah kontrol picture
    CDC dcMem1; //mengarahkan obyek yg sdh dipilih pd bitmap
    CRect rect;//kotak di picture
    BITMAP bm;//mendefinisikan fungsi2 yg ada d bitmap
    int hg[256];
    HBITMAP
    hBitmap=(HBITMAP)::LoadImage(AfxGetInstanceHandle(),
    name,IMAGE_BITMAP, 0, 0,
    LR_LOADFROMFILE|LR_CREATEDIBSECTION);
    if(hBitmap)
    {
        if(m_bmpBitmap.DeleteObject())
            m_bmpBitmap.Detach();
        m_bmpBitmap.Attach(hBitmap);
    }
    m_pic1.GetClientRect(rect);//
    m_bmpBitmap.GetBitmap(&bm);//

    dcMem1.CreateCompatibleDC(pDC);
    dcMem1.SelectObject(&m_bmpBitmap);

    for(i=0;i<256;i++)
        hg[i]=0;
    for(i=0;i<bm.bmHeight;i++)
        for(j=0;j<bm.bmWidth;j++)
        {
            warna=dcMem1.GetPixel(j,i);
            WarnaToRGB(warna,&red,&green,&blue);

            hg[green]=hg[green]+1;
        }
    float hmax=hg[0];
    for(i=1;i<256;i++)
    {
        if(hg[i]>hmax)
            hmax=hg[i];
    }
    for(i=0;i<256;i++)
        hg[i]=120*hg[i]/hmax;

    CDC* pDC1=m_pic3.GetDC();
}

```

Menampilkan bagian warna green (hijau) saja pada gambar hisogramnya

Fungsi untuk menggambar

```

pDC1->MoveTo(0,100);
pDC1->LineTo(450,100); //sb x
pDC1->MoveTo(0,100-(int)hg[0]); //ttk h[0]
for(i=1;i<256;i++)
{
    pDC1->MoveTo(i*2,100);
    pDC1->LineTo(i*2,100-(int)hg[i]);
}
}

```

Fungsi untuk
menggambar
histogram

- i. Tambahkan program pada saat button4 click seperti di bawah ini, akan menampilkan histogram untuk warna biru :

```

void CPrak11_1Dlg::OnButton4()
{
    // TODO: Add your control notification handler code here
    int i,j,red,green,blue;
    long int warna;
    CDC* pDC = m_pic1.GetDC();// mengarah kontrol picture
    CDC dcMem1; //mengarahkan obyek yg sdh dipilih pd bitmap
    CRect rect;//kotak di picture
    BITMAP bm;//mendefinisikan fungsi2 yg ada d bitmap
    int hb[256];
    HBITMAP
    hBitmap=(HBITMAP)::LoadImage(AfxGetInstanceHandle(),
    name,IMAGE_BITMAP, 0, 0,
    LR_LOADFROMFILE|LR_CREATEDIBSECTION);
    if(hBitmap)
    {
        if(m_bmpBitmap.DeleteObject())
            m_bmpBitmap.Detach();
        m_bmpBitmap.Attach(hBitmap);
    }
    m_pic1.GetClientRect(rect);//
    m_bmpBitmap.GetBitmap(&bm);//

    dcMem1.CreateCompatibleDC(pDC);
    dcMem1.SelectObject(&m_bmpBitmap);

    for(i=0;i<256;i++)
        hb[i]=0;
    for(i=0;i<bm.bmHeight;i++)
        for(j=0;j<bm.bmWidth;j++)
        {
            warna=dcMem1.GetPixel(j,i);
            WarnaToRGB(warna,&red,&green,&blue);

            hb[blue]=hb[blue]+1;
        }
    float hmax=hb[0];
    for(i=1;i<256;i++)
    {
        if(hb[i]>hmax)
            hmax=hb[i];
    }
    for(i=0;i<256;i++)
        hb[i]=120*hb[i]/hmax;

    CDC* pDC1=m_pic4.GetDC();
    pDC1->MoveTo(0,100);
    pDC1->LineTo(450,100); //sb x
    pDC1->MoveTo(0,100-(int)hb[0]); //ttk h[0]
    for(i=1;i<256;i++)
    {
        pDC1->MoveTo(i*2,100);
        pDC1->LineTo(i*2,100-(int)hb[i]);
    }
}

```

Menampilkan bagian
warna blue (biru) saja
pada gambar
hisogramnya

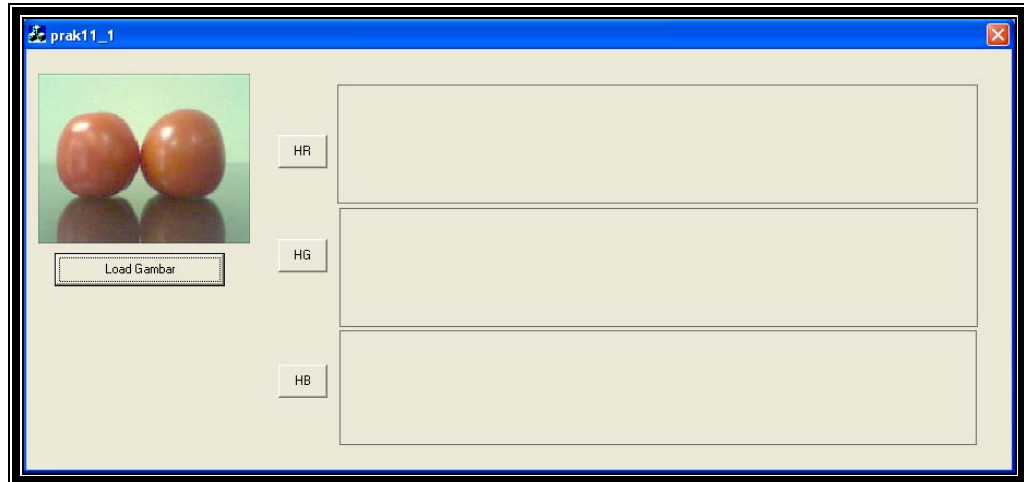
Fungsi untuk
menggambar
histogram

}

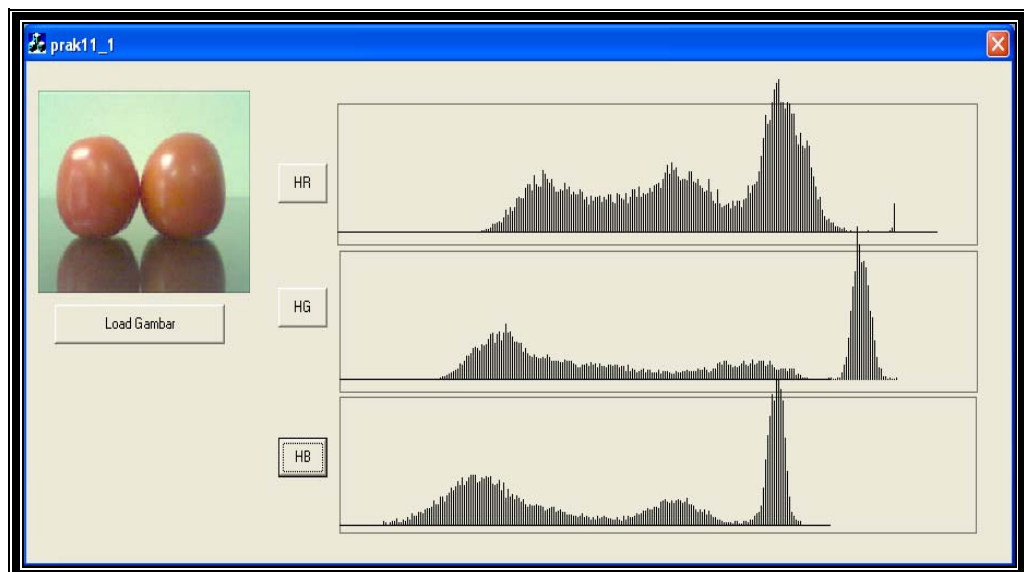
j. Cara menjalankan program

- Pilih menu : Build->Execute (!)
- Klik button Load Gambar : melakukan load gambar
- Klik button HR : melakukan proses histogram untuk warna merah
- Klik button HG : melakukan proses histogram untuk warna hijau
- Klik button HB : melakukan proses histogram untuk warna blue

Hasil running, saat button Load Gambar di klik :



Setelah itu, klik HR, HG, HB, untuk melihat Color Histogram pada gambar tomat merah, maka hasilnya seperti di bawah ini :



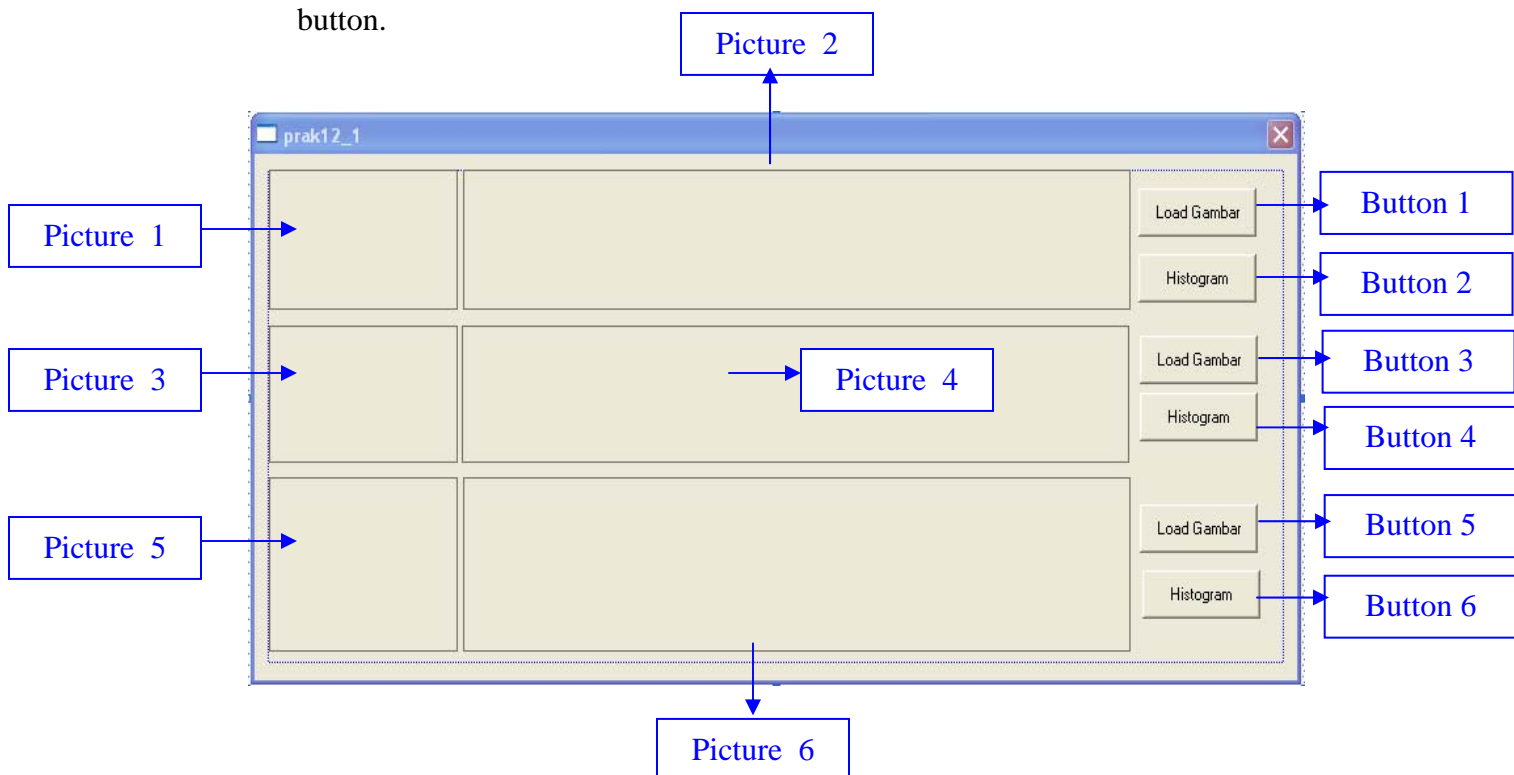
→ Gambar load gambar buah tomat diatas ini, merupakan hasil color histogram untuk masing warna RGB.

Color Histogram

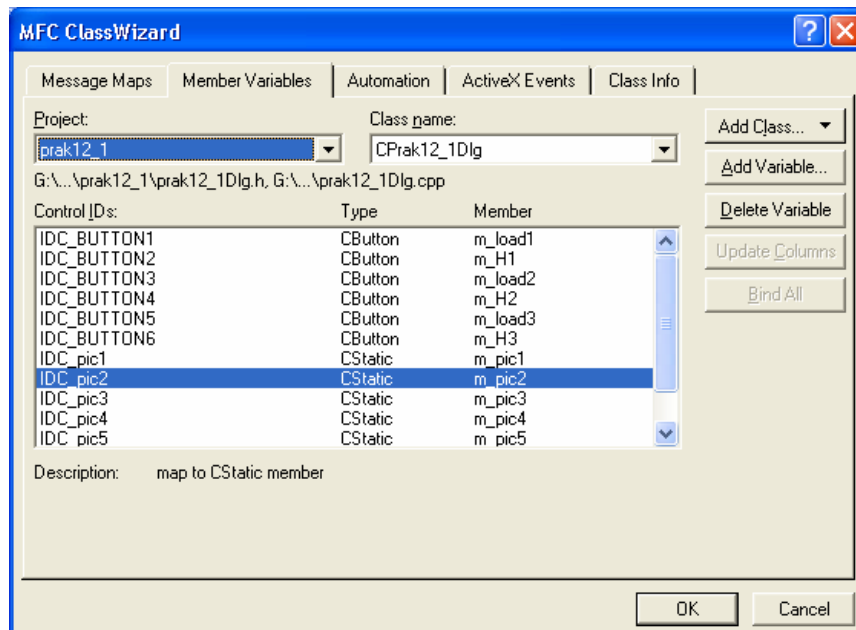
1. Membuat suatu project color histogram dengan mendeteksi warna RGB pada suatu citra secara bersamaan pada satu gambar histogram.

Langkah – langkah membuat project diatas adalah sebagai berikut :

- a. Buat aplikasi AppWizard seperti pada praktikum sebelumnya dan beri nama project dengan prak12_1
- b. Buat disain dialog dengan menggunakan 6 buah control picture, dan 6 buah button.



- c. Merubah properti control dan tambahkan class variabel control tipe dan membernya seperti berikut :



- d. Tambahkan program pada saat button1, 3, dan 5 click seperti di bawah ini, akan melakukan load gambar dalam kotak control picture dengan cara melakukan browse gambar, pilih gambar yang akan diproses.

```
void CPrak12_1Dlg::OnButton1()
{
```

Untuk button 3 dan 4 isi programnya sama seperti button 1

```
// TODO: Add your control notification handler code here
static char BASED_CODE szFilter[]="Bitmap Files (*.bmp) | *.bmp | | ";
```

```
CFileDialog m_ldFile(TRUE, "*.bmp", name, OFN_HIDEREADONLY|OFN_OVERWRITEPROMPT, szFilter);
if(m_ldFile.DoModal()==IDOK)
{
    name=m_ldFile.GetPathName();
}
```

```
CDC* pDC = m_pic1.GetDC();
CDC dcMem; //mengarahkan obyek
CRect rect;//kotak di picture
BITMAP bm;//mendefinisikan fungsi2 yg ada d bitmap
HBITMAP
hBitmap=(HBITMAP)::LoadImage(AfxGetInstanceHandle(), name ,IMAGE_BITMAP, 0, 0, LR_LOADFROMFILE|LR_CREATEDIBSECTION);
if(hBitmap)
{
    if(m_bmpBitmap.DeleteObject())
        m_bmpBitmap.Detach();
    m_bmpBitmap.Attach(hBitmap);
}
```

Untuk Button 3 : CDC* pDC = m_pic3.GetDC();
Untuk Button 5 : CDC* pDC = m_pic5.GetDC();

```
m_pic1.GetClientRect(rect);
m_bmpBitmap.GetBitmap(&bm);
dcMem.CreateCompatibleDC(pDC);
```

Untuk Button 3 : m_pic3.GetClientRect(rect);
Untuk Button 5 : m_pic5.GetClientRect(rect);

```
dcMem.SelectObject(&m_bmpBitmap);
pDC->StretchBlt(0,0,rect.Width(),rect.Height(),&dcMem, 0,0,bm.bmWidth,bm.bmHeight,SRCCOPY);
```

```
}
```

e. Menambah header file

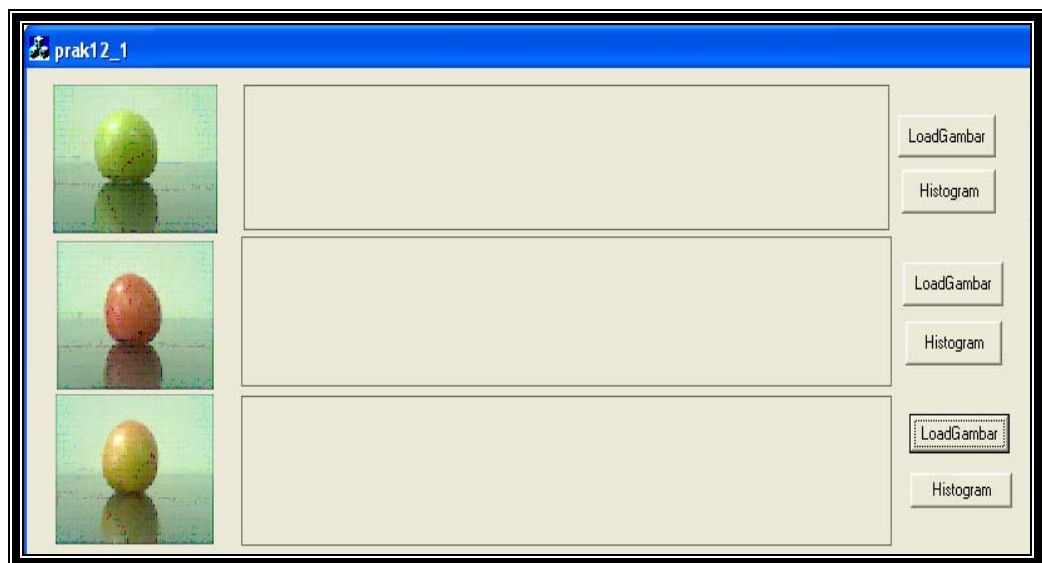
- Buka file prak12_1Dlg.h
- Tambahkan program seperti dibawah ini

```
public:
    CPrak10_1Dlg(CWnd* pParent = NULL); // standard constructor
```

```
CBitmap m_bmpBitmap; → Load Gambar disimpan pada var m_bmpBitmap
```

```
CString name; → Variable temp untuk gambar
```

Saat Ketiga Button Load Gambar di klik, maka hasilnya akan seperti berikut ini :



- f. Tambahkan program pada saat button2, button 4, dan button 6 di click seperti di bawah ini, akan menampilkan color histogram :

```
void CPrak12_1Dlg::OnButton2()
{
    // TODO: Add your control notification handler code here
    int i,j,red,green,blue;
    long int warna;
    CDC* pDC = m_pic1.GetDC();// mengarah kontrol picture
    CDC dcMem1; //mengarahkan obyek yg sdh dipilih pd bitmap
    CRect rect;//kotak di picture
    BITMAP bm;//mendefinisikan fungsi2 yg ada d bitmap
    int h[384]; // Inisialisasi awal jumlahnya 384 karena R
                punya 128, G punya 128 dan B punya 128

    HBITMAP
    hBitmap=(HBITMAP)::LoadImage(AfxGetInstanceHandle(),
    name,IMAGE_BITMAP, 0, 0,
    LR_LOADFROMFILE|LR_CREATEDIBSECTION);
    if(hBitmap)
    {
        if(m_bmpBitmap.DeleteObject())
            m_bmpBitmap.Detach();
        m_bmpBitmap.Attach(hBitmap);
    }
    m_pic1.GetClientRect(rect);//
```

```

m_bmpBitmap.GetBitmap(&bm); //

dcMem1.CreateCompatibleDC(pDC);
dcMem1.SelectObject(&m_bmpBitmap);

for(i=0;i<384;i++)
    h[i]=0;
for(i=0;i<bm.bmHeight;i++)
    for(j=0;j<bm.bmWidth;j++)
    {
        warna=dcMem1.GetPixel(j,i);
        WarnaToRGB(warna,&red,&green,&blue);
        red=red/2;
        green=green/2;
        blue=blue/2;
        h[red]++;
        h[128+green]++;
        h[256+blue]++;
    }

float hmax=h[0];
for(i=1;i<384;i++)
{
    if(h[i]>hmax)
        hmax=h[i];
}
for(i=0;i<384;i++)
    h[i]=120*h[i]/hmax;

CDC* pDC1=m_pic2.GetDC();
pDC1->MoveTo(0,100);
pDC1->LineTo(450,100); //sb x
pDC1->MoveTo(0,100-(int)h[0]); //ttk h[0]
for(i=1;i<384;i++)
{
    pDC1->MoveTo(i*3,100);
    pDC1->LineTo(i*3,100-(int)h[i]);
}
}

```

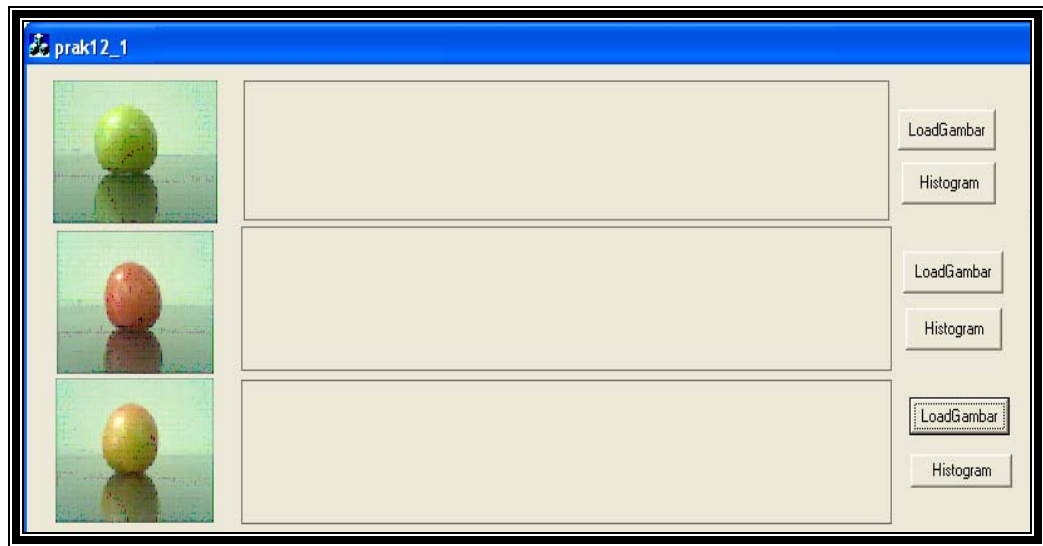
→ Untuk Red jangkauannya: 0 - 127
 Untuk Green jangkauannya: 128 -255
 Untuk Blue jangkauannya: 256-512

→ Fungsi untuk menggambar histogram

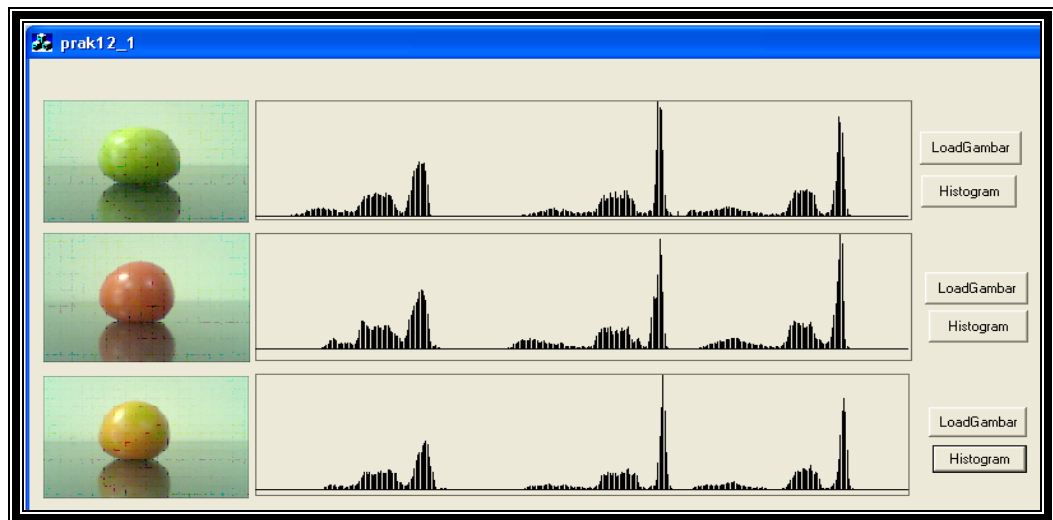
k. Cara menjalankan program

- Pilih menu : Build->Execute (!)
- Klik button LoadGambar untuk button 1,3,5 : melakukan load gambar
- Klik button Histogram untuk button 2,,4,6 : untuk menampilkan color histogram

Hasil running, saat button LoadGambar di klik :



Setelah itu, saat button Histogram di klik :

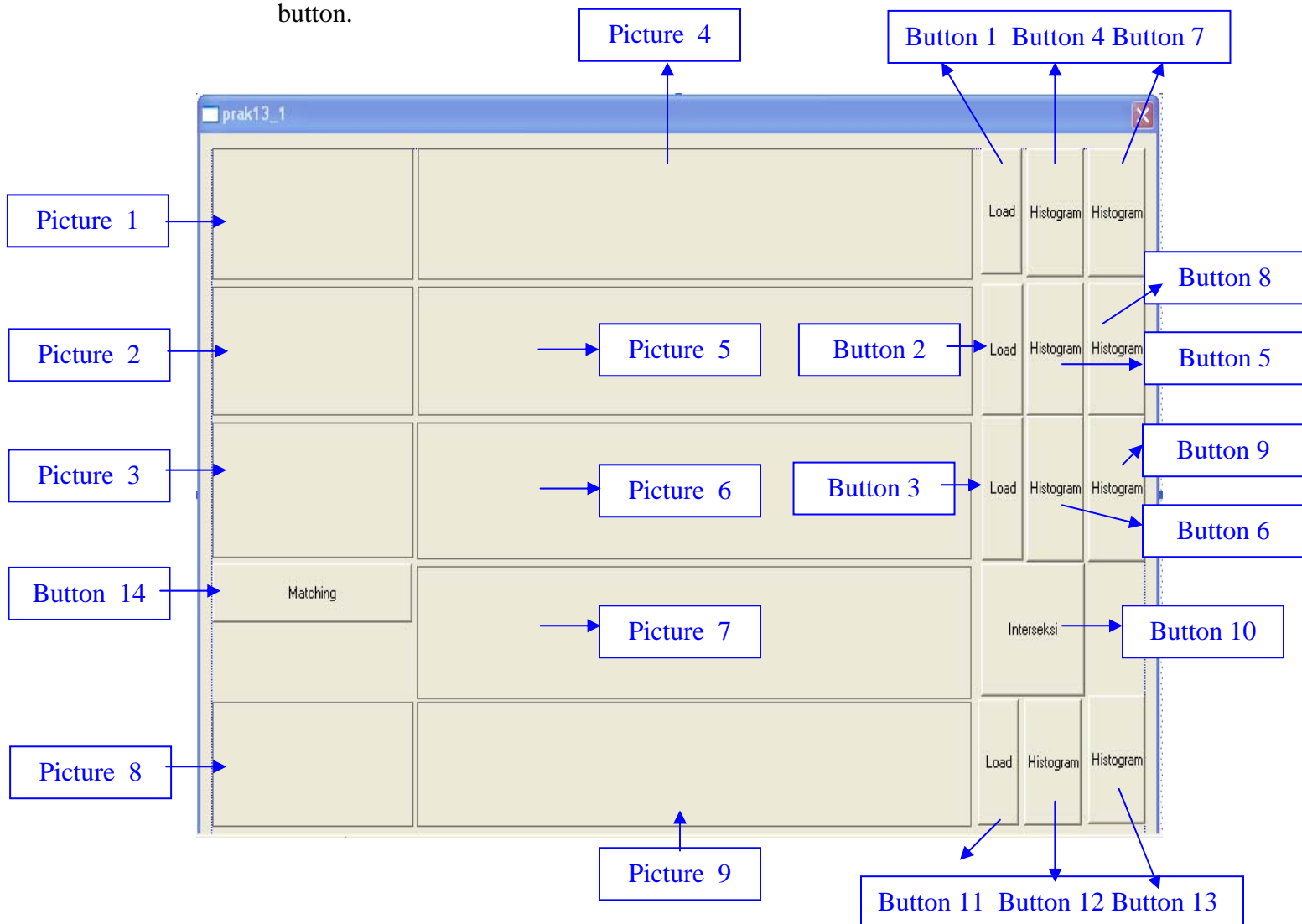


Color Histogram

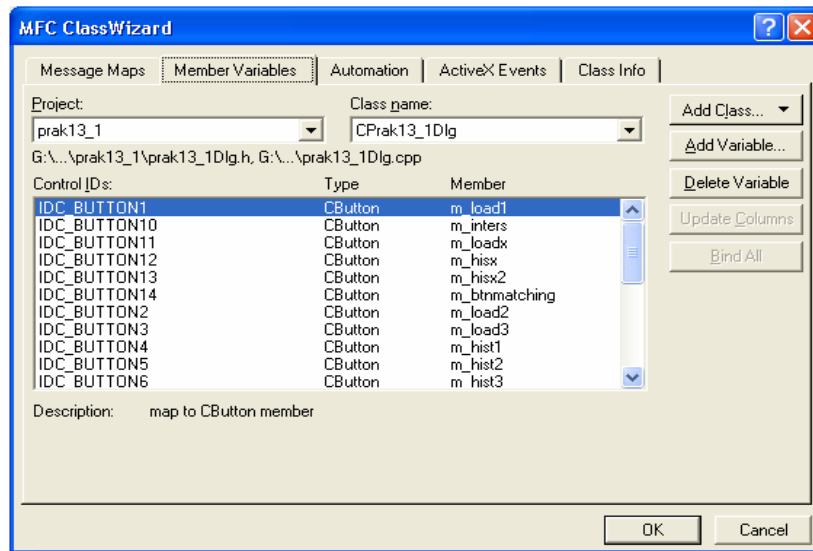
1. Membuat suatu project color histogram dengan mendeteksi warna RGB pada suatu citra secara bersamaan pada satu gambar histogram.

Langkah – langkah membuat project diatas adalah sebagai berikut :

- Buat aplikasi AppWizard seperti pada praktikum sebelumnya dan beri nama project dengan prak13_1
- Buat disain dialog dengan menggunakan 6 buah control picture, dan 6 buah button.



- Merubah properti control dan tambahkan class variabel control tipe dan membernya seperti berikut :



- d. Tambahkan program pada saat button1, 2, 3 dan 11 click seperti di bawah ini, akan melakukan load gambar dalam kotak control picture dengan menentukan gambar apa yang akan di load .

```
void CPrak13_1Dlg::OnButton1()
{
    CDC* pDC = m_pic1.GetDC();//
    CDC dcMem1;
    CRect rect;//kotak di picture
    BITMAP bm;//
    HBITMAP
    hBitmap=(HBITMAP)::LoadImage(AfxGetInstanceHandle(),
    "10.bmp", IMAGE_BITMAP, 0, 0,
    LR_LOADFROMFILE|LR_CREATEDIBSECTION);
    if(hBitmap) {
        if(m_bmpBitmap.DeleteObject())
            m_bmpBitmap.Detach();
        m_bmpBitmap.Attach(hBitmap);
    }
    m_pic1.GetClientRect(rect);//
    m_bmpBitmap.GetBitmap(&bm);//
    dcMem1.CreateCompatibleDC(pDC);
    dcMem1.SelectObject(&m_bmpBitmap);
    pDC->StretchBlt(0,0,rect.Width(),rect.Height(),&dcMem1,
    0,0,bm.bmWidth,bm.bmHeight,SRCCOPY);
}
```

Untuk button 2, 3 dan 11 isi programnya sama seperti button 1

Untuk Button 2 : CDC* pDC = m_pic2.GetDC();
 Untuk Button 3 : CDC* pDC = m_pic3.GetDC();
 Untuk Button 11 : CDC* pDC = m_pic8.GetDC();

Untuk button 2, gambar yang di load : "50.bmp"
 Untuk button 3, gambar yang di load : "90.bmp"
 Untuk button 11, gambar yang di load : secara browse gambar

- e. Menambah header file

- Buka file prak12_1Dlg.h
- Tambahkan program seperti dibawah ini :

```
public:
    CPrak13_1(CWnd* pParent = NULL); // standard constructor
    CString name;
    CBitmap m_bmpBitmap;
```

Load Gambar disimpan pada var m_bmpBitmap

```
float gn[384];
float h[4][384];
float hs[4][384];
```

Variable untuk histogram

Saat Ketiga Button Load Gambar di klik, maka hasilnya akan seperti berikut ini :



- f. Tambahkan program pada saat button 4, button 5, 6 dan button 12 di click seperti di bawah ini, akan menampilkan color histogram :

```
void Cprak13_1Dlg::OnButton4()
{
    CDC* pDC = m_pic1.GetDC();
    CDC dcMem;
    CRect rect;
    BITMAP bm;
    int i,j;
    int red, green, blue;
    long int warna;
    LoadGambar1();
    m_pic1.GetClientRect(rect);
    m_bmpBitmap.GetBitmap(&bm);
    dcMem.CreateCompatibleDC(pDC);
    dcMem.SelectObject(&m_bmpBitmap);

    for(i=0;i<384;i++) h[0][i]=0;

    for(i=0;i<bm.bmHeight;i++)
        for(j=0;j<bm.bmWidth;j++)
        {
            warna=dcMem.GetPixel(j,i);
            WarnaToRGB(warna,&red,&green,&blue);
            red=red/2;
            green=green/2;
            blue=blue/2;
            h[0][red]++;
            h[0][128+green]++;
            h[0][256+blue]++;
        }

    float hmax=h[0][0];
}
```

Untuk button 5, 6 dan 12 isi programnya sama seperti button 4
 Untuk Button 5 : CDC* pDC = m_pic2.GetDC();
 Untuk Button 6 : CDC* pDC = m_pic3.GetDC();
 Untuk Button 12 : CDC* pDC = m_pic8.GetDC();
 Untuk Button 5 : m_pic2.GetClientRect(rect);
 Untuk Button 6 : m_pic3.GetClientRect(rect);
 Untuk Button 12 : m_pic8.GetClientRect(rect);
 Untuk Button 5 : h[1][i] = 0;
 Untuk Button 6 : h[2][i] = 0;
 Untuk Button 5 : h[1][red]++;
 h[1][128+green]++;
 h[1][256+blue]++;
 Untuk Button 6 : h[2][red]++;
 h[2][128+green]++;
 h[2][256+blue]++;
 Untuk Button 5 : h[1][0];
 Untuk Button 6 : h[2][0];


```
for(i=1;i<384;i++)
    if(h[0][i]>hmax) hmax=h[0][i];
```

Untuk Button 5 :
if(h[1][i]>hmax)hmax=h[1][i];
Untuk Button 6 :
if(h[2][i]>hmax)hmax=h[2][i];

```
for(i=0;i<384;i++)
    h[0][i]=110*h[0][i]/hmax;
```

Untuk Button 5 :
h[1][i]=110*h[1][i]/hmax;
Untuk Button 6 :
h[2][i]=110*h[2][i]/hmax;

Untuk Button 5 :
CDC* pDC = m_pic5.GetDC();
Untuk Button 6 :
CDC* pDC = m_pic6.GetDC();

```
pDC1->MoveTo(0,120);
pDC1->LineTo(512,120);
pDC1->MoveTo(0,120-(int)h[0][0]);
```

Untuk Button 5 :
pDC1->MoveTo(0,120-(int)h[1][0]);
Untuk Button 6 :
pDC1->MoveTo(0,120-(int)h[2][0]);

```
for(i=1;i<384;i++){
    pDC1->MoveTo(int(i*1.3),120);
    pDC1->LineTo(int(i*1.3),120-(int)h[0][i]);
}
}
```

Untuk Button 5 :
pDC1->LineTo(int(i*1.3),120-(int)h[1][i]);
Untuk Button 6 :
pDC1->LineTo(int(i*1.3),120-(int)h[2][i]);

- g. Tambahkan program pada saat button interseksi di klik, button ini untuk menampilkan button histogram interseksi :

```
void CColorHistDlg::OnButton10()
{
    int i;
    for(i=0;i<384;i++){
        gn[i]=h[0][i];
        if(h[1][i]<gn[i]) gn[i]=h[1][i];
        if(h[2][i]<gn[i]) gn[i]=h[2][i];
    }
    CDC* pDC1=m_pic7.GetDC();
    pDC1->MoveTo(0,120);
    pDC1->LineTo(512,120);
    pDC1->MoveTo(0,120-(int)gn[0]);
    for(i=1;i<384;i++){
        pDC1->MoveTo(int(i*1.3),120);
        pDC1->LineTo(int(i*1.3),120-(int)gn[i]);
    }
}
```

Fungsi untuk histogram interseksi

Grafik histogram interseksi berada pada picture 7

- h. Tambahkan program pada saat button 7,8 , 9, dan 13 di klik, button ini untuk menampilkan histogram setelah dilakukan histogram interseksi :

```
void CColorHistDlg::OnButton7()
{
    CDC* pDC1=m_pic4.GetDC();
    long w;
    int i;
    w=RGBToWarna(255,255,255);
    pDC1->FillSolidRect(0,0,512,130,w);
    for(i=0;i<384;i++)
        hs[0][i]=fabs(double(h[0][i]-gn[i]));
```

Untuk button 7, 8, 9 dan 13 isi programnya sama seperti button 7

Untuk Button 8 : CDC* pDC = m_pic5.GetDC();
Untuk Button 9 : CDC* pDC = m_pic6.GetDC();
Untuk Button 13 : CDC* pDC = m_pic9.GetDC();

```

pDC1->MoveTo(0,120);
pDC1->LineTo(512,120);
pDC1->MoveTo(0,120-(int)hs[0][0]);
w=RGBToWarna(255,255,255);

for(i=1;i<384;i++){
    pDC1->MoveTo(int(i*1.3),120);
    pDC1->LineTo(int(i*1.3),120-(int)hs[0][i]);
}
}

```

Untuk Button 8 :

```
hs[1][i]=fabs(double(h[1][i]-gn[i]));
```

Untuk Button 9 :

```
hs[2][i]=fabs(double(h[2][i]-gn[i]));
```

Untuk Button 13 :

```
hs[3][i]=fabs(double(h[3][i]-gn[i]));
```

- i. Tambahkan program pada saat button 14, untuk melakukan proses matching :

```

void CColorHistDlg::OnButton14()
{
    // TODO: Add your control notification handler code here
    CDC* pDC1=GetDC();
    float d[3],di=0;
    int i,j,k=0;
    long w;
    char str[5],str2[5];
    for(i=0;i<3;i++)
    {
        d[i]=0.;
        for(j=0;j<384;j++)
            di=di+fabs (double(hs[3][j]-hs[i][j]));
        di=di/384.;
        d[i]=di;
        sprintf(str2,"%d",i);
        pDC1->TextOut(50,500+15*i,str2);
        sprintf(str,"%f",d[i]);
        pDC1->TextOut(100,500+15*i,str);
    }
    w=RGBToWarna(255,255,255);
    pDC1->SetBkColor(w);
    if ((d[0]<d[1])&&(d[0]<d[2]))
    {
        k=0;
        pDC1->TextOut(50,545,"HIJAU");
    }
    if ((d[1]<d[0])&&(d[1]<d[2]))
    {
        k=1;
        pDC1->TextOut(50,545,"MERAH");
    }
    if ((d[2]<d[0])&&(d[2]<d[1]))
    {
        k=2;
        pDC1->TextOut(50,545,"CAMPURAN");
    }
}
}

```

→ Fungsi untuk melakukan proses matching

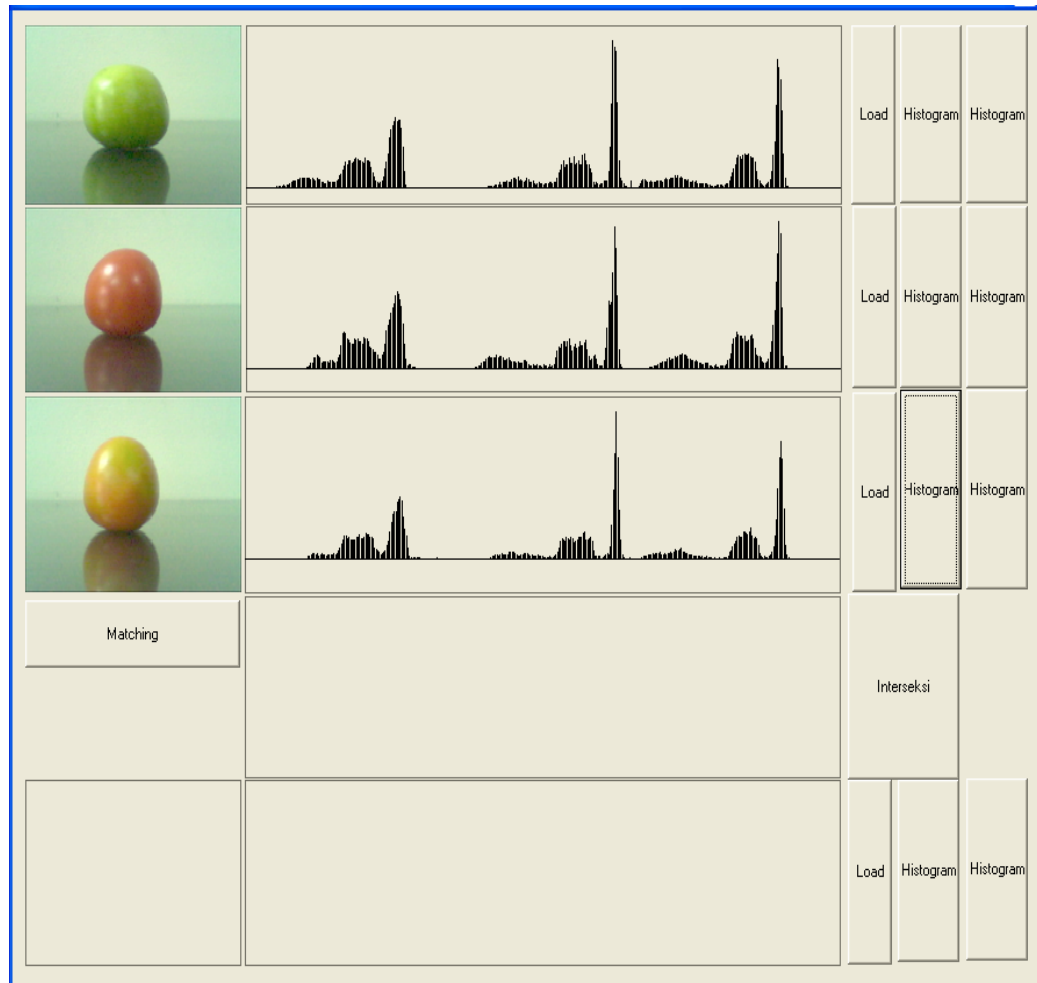
→ Menampilkan tulisan keluar, berdasarkan matching sesuai warna RGBnya

- j. Cara menjalankan program

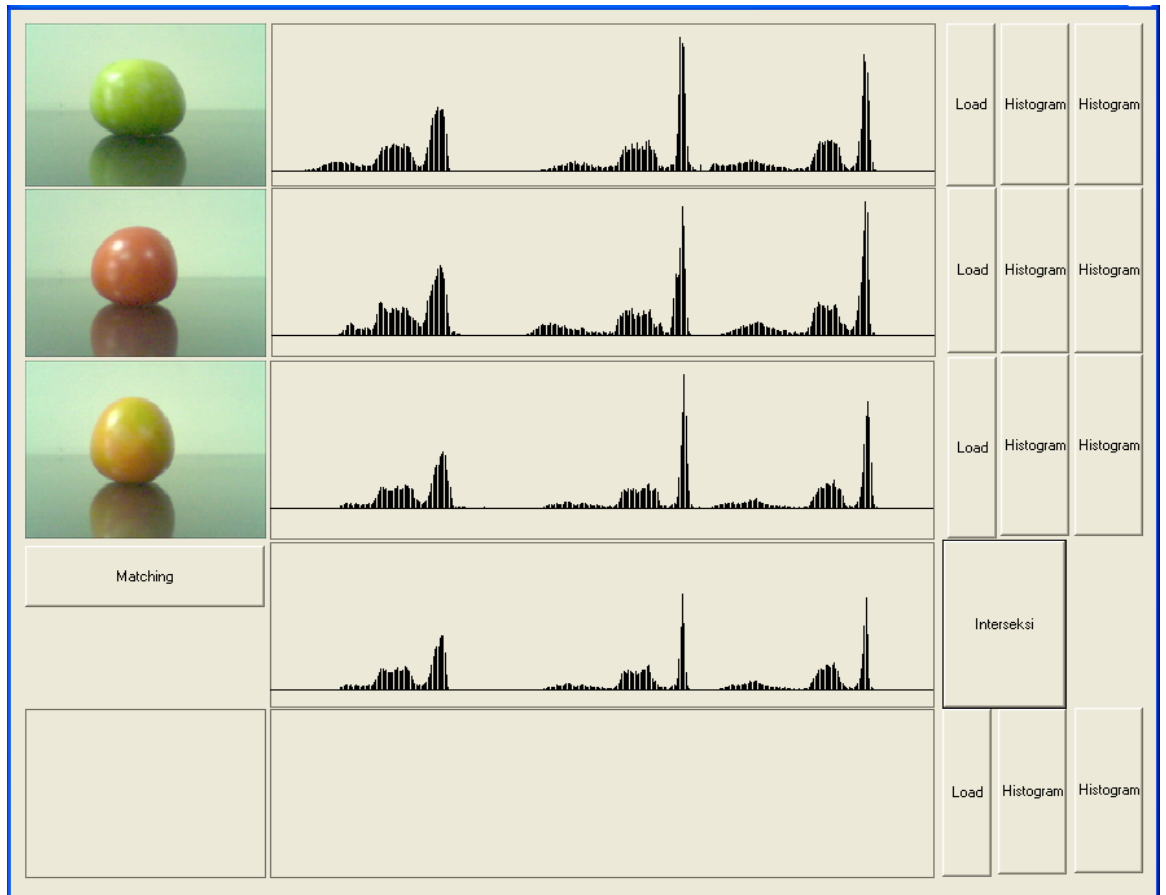
- Pilih menu : Build->Execute (!)
- Klik button LoadGambar untuk button 1,2,3 : melakukan load gambar
- Klik button Histogram untuk button 4,5,6 : untuk menampilkan color histogram
- Klik button Histogram untuk button 7,8,9 : untuk menampilkan histogram setelah dilakukan histogram interseksi
- Klik button Interseksi : untuk melakukan proses histogram interseksi
- Klik button 11,12,13 : untuk membandingkan hasil ketiga picture

- Klik button Matching : untuk melakukan proses matching pada image

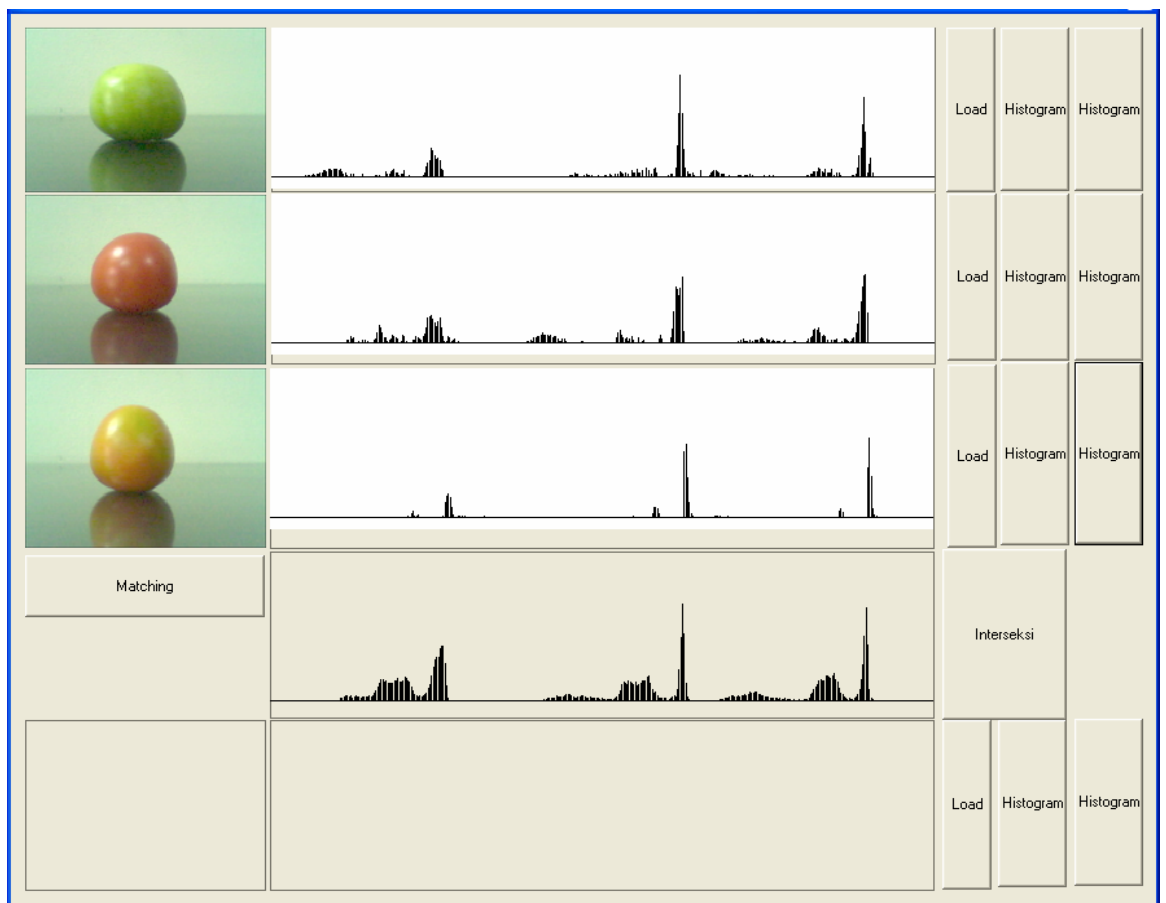
✓ Hasil Load Gambar dan histogram awal pada ketiga obje buah tomat :



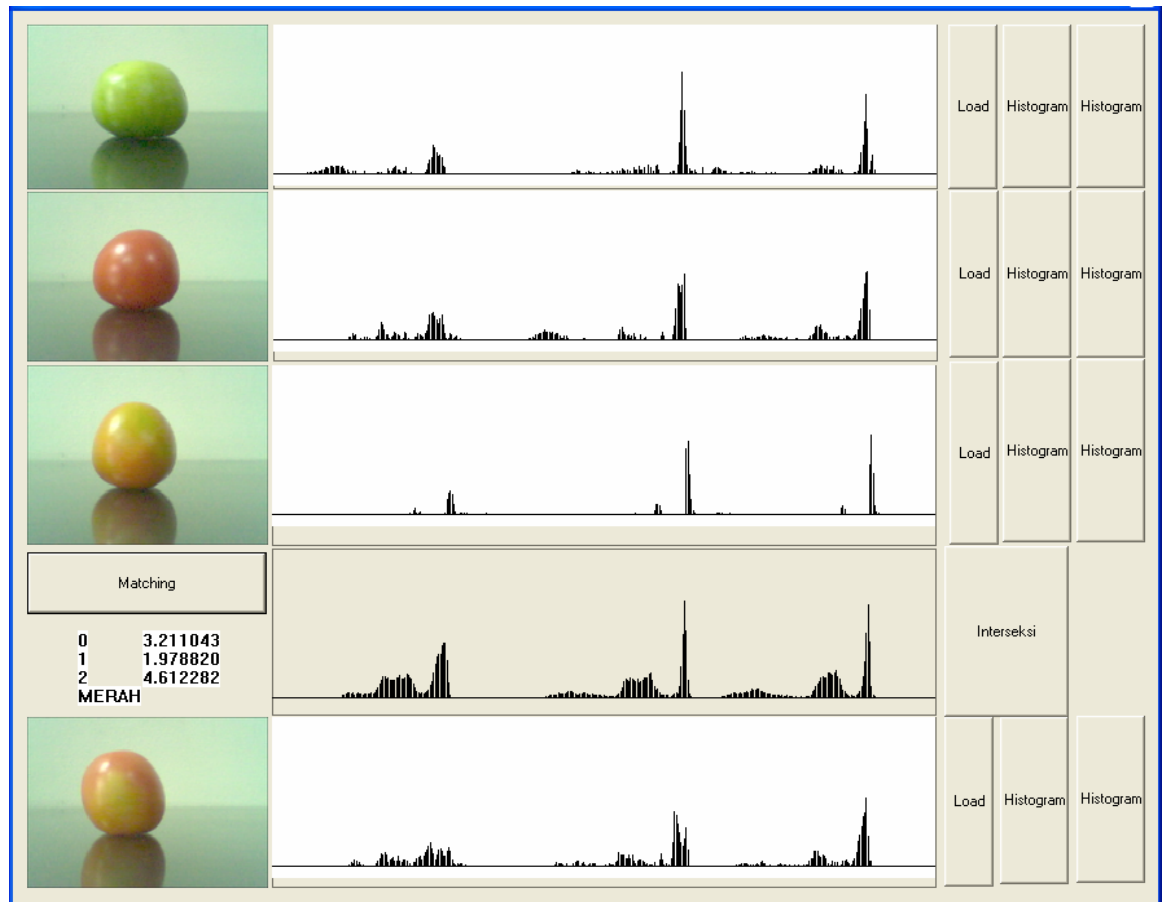
✓ Klik Button Interseksi, maka hasilnya seperti di bawah ini :



- ✓ Setelah Klik Button histogram yang kedua, setelah dilakukan interseksi pada spesifik :



- ✓ Kemudian klik load gambar dan button matching bagian bawah untuk membandingkan hasil citra



IV. KESIMPULAN

1. Buatlah kesimpulan hasil praktikum 10 color histogram pada buah jeruk(kuning), tomat (merah), jambu (hijau)
2. Buatlah kesimpulan hasil praktikum 11 color histogram pada 3 macam jeruk, 3 macam tomat, 3 macam jambu
3. Buatlah kesimpulan hasil praktikum 12 color histogram pada 3 macam jeruk, 3 macam tomat, 3 macam jambu dan berapa persentase keberhasilan pengenalan untuk setiap macam warna buah.