

PRAKTIKUM I

Pembacaan dan Penampilan Image

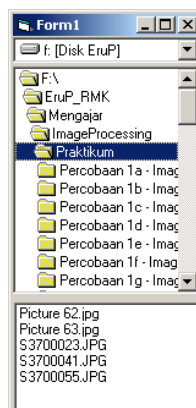
Tujuan

- Memperkenalkan bahasa pemrograman tertentu sebagai alat bantu pemrograman Image Processing
- Memperkenalkan Image dan komponen-komponennya
- Memperkenalkan cara-cara pembacaan dan menampilkan Image

Teori Penunjang

Prosedur Percobaan

1. Pembacaan dan Penampilan Image
 - a. Percobaan berikut hanya berkaitan dengan membaca image dari file dan langsung ditambahkan pada komponen visual tertentu
 - b. Mengambil Gambar dari File
 - i. Buat suatu project baru yang berisi satu form dengan nama Form1
 - ii. Tambahkan komponen DriveListBox, DirListBox, dan FileListBox dengan nama masing-masing Drive1, Dir1 dan File1 pada Form1



- iii. Masukkan program berikut pada Form1

```

Option Explicit

Private Sub Dir1_Change()
    File1.Path = Dir1.Path
End Sub

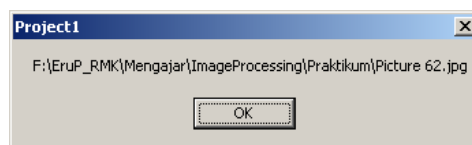
Private Sub Drive1_Change()
    Dir1.Path = Drive1.Drive
End Sub

Private Sub File1_Click()
    MsgBox File1.Path + "\" + File1.FileName
End Sub

Private Sub Form_Load()
    File1.Pattern = "*.bmp;*.jpg;*.jpeg;*.gif;*.tif"
End Sub

```

iv. Jalankan program dan pilih file tertentu



c. Menampilkan Menggunakan Komponen Form

i. Tambahkan Form baru pada project dengan nama Form2



ii. Ubah program pada sub File1_Click()

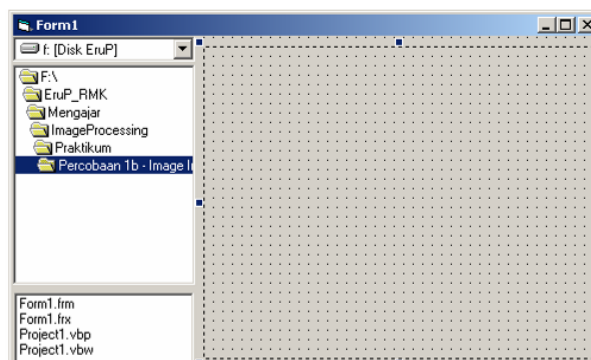
```

Private Sub File1_Click()
    Form2.Picture = LoadPicture(File1.Path + "\" + File1.FileName)
    Form2.Show
End Sub

```

d. Menampilkan Menggunakan Komponen Image

i. Pasang komponen Image1 pada Form1



ii. Ubah program pada sub File1_Click()

```

Private Sub File1_Click()

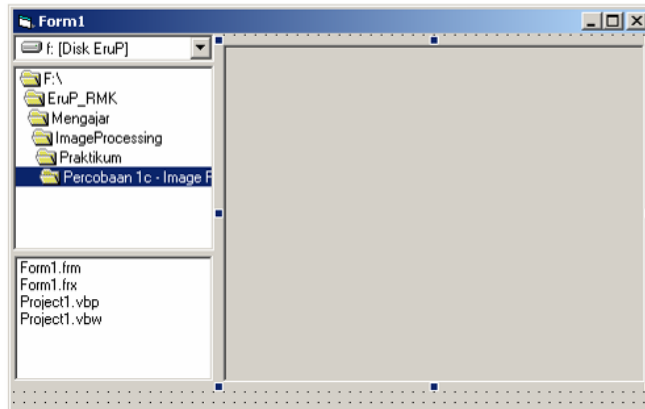
```

```
Image1.Picture = LoadPicture(File1.Path + "\" + File1.FileName)
End Sub
```

iii. Set/ubah Property Stretch dari Image1 bernilai True atau False dan coba sekali lagi

e. Menampilkan Menggunakan Komponen PictureBox

i. Pasang komponen Picture1 pada Form1



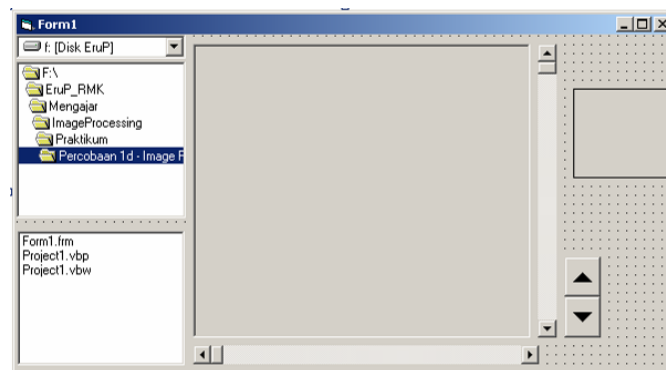
ii. Ubah program pada sub File1_Click()

```
Private Sub File1_Click()
    Picture1.Picture = LoadPicture(File1.Path + "\" + File1.FileName)
End Sub
```

iii. Set/ubah Property Auto Size dari Picture1 bernilai True atau False dan coba sekali lagi

f. Menampilkan Menggunakan Komponen PictureBox

i. Pasang komponen Picture1, PictureBox1, ScrollBar1, ScrollBar2 dan SpinButton1 pada Form1.



ii. Ubah program pada sub File1_Click(), Form_Load() dan tambahkan beberapa sub program berikut.

```
Private Sub File1_Click()
    PictureBox1.Picture = LoadPicture(File1.Path + "\" + File1.FileName)
    ScrollBar1.Max = PictureBox1.Height - 1
    ScrollBar2.Max = PictureBox1.Width - 1
    ScrollBar1.Value = 0
    ScrollBar2.Value = 0
    SpinButton1.Min = 1
    PictureBox1.ScaleMode = 3
```

```

    If PictureClip1.Height / Picture1.ScaleHeight < _
        PictureClip1.Width / Picture1.ScaleWidth Then
        SpinButton1.Max = 100 & * PictureClip1.Height / Picture1.ScaleHeight
    Else
        SpinButton1.Max = 100 & * PictureClip1.Width / Picture1.ScaleWidth
    End If
    If SpinButton1.Max < 1 Then SpinButton1.Max = 1
    SpinButton1.Value = SpinButton1.Max
    Tampil
End Sub

Private Sub Form_Load()
    File1.Pattern = "*.bmp;*.jpg;*.jpeg;*.gif;*.tif"
    ScrollBar1.Max = 1000
    ScrollBar2.Max = 1000
    SpinButton1.Min = 10
    SpinButton1.Max = 1000
End Sub

Private Sub Tampil()
    On Error Resume Next
    PictureClip1.ClipX = ScrollBar2.Value
    PictureClip1.ClipY = ScrollBar1.Value
    PictureClip1.ClipHeight = Picture1.ScaleHeight * SpinButton1.Value / 100
    If PictureClip1.ClipHeight > PictureClip1.Height Then
        PictureClip1.ClipHeight = PictureClip1.Height
    End If
    PictureClip1.ClipWidth = Picture1.ScaleWidth * SpinButton1.Value / 100
    If PictureClip1.ClipWidth > PictureClip1.Width Then
        PictureClip1.ClipWidth = PictureClip1.Width
    End If
    PictureClip1.StretchX = Picture1.ScaleWidth
    PictureClip1.StretchY = Picture1.ScaleHeight
    Picture1.Picture = PictureClip1.Clip
    On Error GoTo 0
End Sub

Private Sub ScrollBar1_Change()
    Tampil
End Sub

Private Sub ScrollBar2_Change()
    Tampil
End Sub

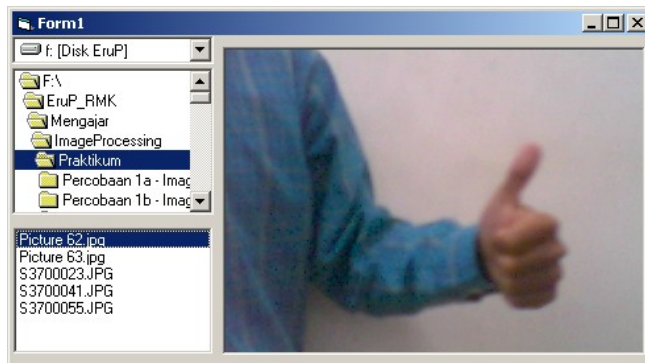
Private Sub SpinButton1_Change()
    Tampil
End Sub

```

iii. Uji coba program dengan membaca file gambar yang berukuran besar dan geser-geser ScrollBar serta Zoom gambar menggunakan SpinButton.

2. Pembacaan Data Image, dianggap menggunakan PictureBox dengan ukuran terbatas
 - a. Program berikut akan melakukan pembacaan data-data image yang telah diambil dari file menjadi sebuah variable vImage yang dapat digunakan untuk keperluan image processing.
 - b. Untuk memudahkan percobaan, dianggap proses pembacaan dan penampilan gambar dari file menggunakan komponen PictureBox dengan ukuran gambar yang tertentu (320 x 240).

- c. Anda dapat menggunakan project sebelumnya yang membaca file gambar ke PictureBox.



- d. Membaca Data Menggunakan Point

i. Sifat

- ✓ Tidak menggunakan deklarasi tambahan
- ✓ Tidak bergantung setting warna sistem
- ✓ Bergantung mode skala dari PictureBox/ Image/ Form yang digunakan (Twip, Pixel, cm, mm, User)
- ✓ Bergantung dari mode "Auto Redraw"
- ✓ Lambat

- ii. Tambahkan Module1 pada Project untuk memasukkan program berikut

```
Option Explicit

' Deklarasi Jenis type Data RGB, untuk keperluan Image Processing
Public Type tRGB24
    B As Byte
    G As Byte
    R As Byte
End Type

Global vImage(0 To 319, 0 To 239) As tRGB24
```

- ii. Masukkan program berikut pada Form1

```
Option Explicit

Private Sub Dir1_Change()
    File1.Path = Dir1.Path
End Sub

Private Sub Drive1_Change()
    Dir1.Path = Drive1.Drive
End Sub

Private Sub File1_Click()
    Picture1.Picture = LoadPicture(File1.Path + "\" + File1.FileName)
    Dim y As Integer, x As Integer
    Dim p As Long
    For y = 0 To Picture1.ScaleHeight - 1
        For x = 0 To Picture1.ScaleWidth - 1
            p = Picture1.Point(x, y)
        
```

```

        vImage(x, y).R = p And &HFF
        vImage(x, y).G = (p \ &H100) And &HFF
        vImage(x, y).B = (p \ &H10000) And &HFF
    Next
Next
MsgBox "Selesai membaca data"
End Sub

Private Sub Form_Load()
    File1.Pattern = "*.bmp;*.jpg;*.jpeg;*.gif;*.tif"
    Picture1.ScaleMode = 3
End Sub

```

e. Membaca Data Menggunakan GetPixel

i. Sifat

- ✓ Memerlukan deklarasi panggilan ke Window API
- ✓ Tidak bergantung setting warna sistem
- ✓ Tidak bergantung mode skala
- ✓ Tidak bergantung dari mode "Auto Redraw"
- ✓ Lebih cepat

ii. Masukkan program berikut pada Module1

```

Option Explicit

' Deklarasi Jenis type Data RGB, untuk keperluan Image Processing
Public Type tRGB24
    B As Byte
    G As Byte
    R As Byte
End Type

Public Declare Function GetPixel Lib "gdi32" ( _
    ByVal hdc As Long, ByVal x As Long, ByVal y As Long) As Long

Global vImage(0 To 319, 0 To 239) As tRGB24

```

iii. Masukkan program berikut pada Form1

```

Option Explicit

Private Sub Dir1_Change()
    File1.Path = Dir1.Path
End Sub

Private Sub Drive1_Change()
    Dir1.Path = Drive1.Drive
End Sub

Private Sub File1_Click()
    Picture1.Picture = LoadPicture(File1.Path + "\" + File1.FileName)
    Dim y As Integer, x As Integer
    Dim p As Long
    For y = 0 To Picture1.ScaleHeight - 1
        For x = 0 To Picture1.ScaleWidth - 1
            p = GetPixel(Picture1.hdc, x, y)
            vImage(x, y).R = p And &HFF
            vImage(x, y).G = (p \ &H100) And &HFF
            vImage(x, y).B = (p \ &H10000) And &HFF
        Next
    Next
    MsgBox "Selesai membaca data"
End Sub

```

```

Private Sub Form_Load()
    File1.Pattern = "*.bmp;*.jpg;*.jpeg;*.gif;*.tif"
    Picture1.ScaleMode = 3
End Sub

```

f. Membaca Data Menggunakan Copy

i. Sifat

- ✓ Memerlukan deklarasi panggilan ke Window API
- ✓ Bergantung setting warna sistem
- ✓ Bergantung dengan ukuran pixel file gambar
- ✓ Tidak bergantung mode skala
- ✓ Tidak bergantung dari mode "Auto Redraw"
- ✓ Sangat Cepat

ii. Masukkan program berikut pada Module1

```

Option Explicit

' Deklarasi Jenis type Data RGB, untuk keperluan Image Processing
Public Type tRGB24
    B As Byte
    G As Byte
    R As Byte
End Type

Public Type BITMAP '14 bytes
    bmType As Long
    bmWidth As Long
    bmHeight As Long
    bmWidthBytes As Long
    bmPlanes As Integer
    bmBitsPixel As Integer
    bmBits As Long
End Type

Public Declare Function GetObject Lib "gdi32" Alias "GetObjectA" ( _
    ByVal hObject As Long, ByVal nCount As Long, lpObject As Any) As Long
Public Declare Sub RtlMoveMemory Lib "kernel32" ( _
    ByVal hpvDest As Long, ByVal hpvSource As Long, ByVal cbCopy As Long)

Global vImage(0 To 319, 0 To 239) As tRGB24

```

iii. Masukkan program berikut pada Form1

```

Option Explicit

Private Sub Dir1_Change()
    File1.Path = Dir1.Path
End Sub

Private Sub Drive1_Change()
    Dir1.Path = Drive1.Drive
End Sub

Private Sub File1_Click()
    Picture1.AutoSize = True
    Picture1.Picture = LoadPicture(File1.Path + "\" + File1.FileName)
    Dim bmp As BITMAP, u As Long
    GetObject Picture1.Picture.Handle, Len(bmp), bmp
    u = bmp.bmWidthBytes * bmp.bmHeight
    If u > 320# * 240 * 3 Then u = 320# * 240 * 3

```

```

    RtlMoveMemory VarPtr(vImage(0, 0)), bmp.bmBits, u
    MsgBox "Selesai membaca data"
End Sub

Private Sub Form_Load()
    File1.Pattern = "*.bmp;*.jpg;*.jpeg;*.gif;*.tif"
    Picture1.ScaleMode = 3
End Sub

```

iv. Sebelumnya atur setting warna window pada RGB 24 bit dan baca file gambar dengan ukuran 320 x 240

3. Menampilkan Kembali Data Image, asumsi menggunakan GetPixel

- a. Program berikut digunakan untuk menampilkan kembali data dari variable Image ke komponen visual
- b. Untuk memudahkan dianggap proses pembacaan data gambar menggunakan komponen PictureBox dengan cara GetPixel
- c. Menampilkan Data Menggunakan Pset

i. Sifat

- ✓ Tidak menggunakan deklarasi tambahan
- ✓ Tidak bergantung setting warna sistem
- ✓ Bergantung mode skala dari PictureBox/ Image/ Form yang digunakan (Twip, Pixel, cm, mm, User)
- ✓ Bergantung dari mode "Auto Redraw"
- ✓ Lambat

ii. Masukkan Program berikut pada Module1

```

Option Explicit

' Deklarasi Jenis type Data RGB, untuk keperluan Image Processing
Public Type tRGB24
    B As Byte
    G As Byte
    R As Byte
End Type

Public Declare Function GetPixel Lib "gdi32" ( _
    ByVal hdc As Long, ByVal x As Long, ByVal y As Long) As Long

Global vImage(0 To 319, 0 To 239) As tRGB24

```

iii. Masukkan Program berikut pada Form1

```

Option Explicit

Private Sub Dir1_Change()
    File1.Path = Dir1.Path
End Sub

Private Sub Drive1_Change()
    Dir1.Path = Drive1.Drive
End Sub

Private Sub File1_Click()
    Picture1.Picture = LoadPicture(File1.Path + "\" + File1.FileName)

```



```

Dim y As Integer, x As Integer
Dim p As Long
For y = 0 To Picture1.ScaleHeight - 1
    For x = 0 To Picture1.ScaleWidth - 1
        p = GetPixel(Picture1.hdc, x, y)
        vImage(x, y).R = p And &HFF
        vImage(x, y).G = (p \ &H100) And &HFF
        vImage(x, y).B = (p \ &H10000) And &HFF
    Next
Next
For y = 0 To Picture1.ScaleHeight - 1
    For x = 0 To Picture1.ScaleWidth - 1
        vImage(x, y).B = 0
    Next
Next
For y = 0 To Picture1.ScaleHeight - 1
    For x = 0 To Picture1.ScaleWidth - 1
        Picture1.PSet (x, y), _
            RGB(vImage(x, y).R, vImage(x, y).G, vImage(x, y).B)
    Next
Next
End Sub

Private Sub Form_Load()
    File1.Pattern = "*.bmp;*.jpg;*.jpeg;*.gif;*.tif"
    Picture1.ScaleMode = 3
End Sub

```

d. Menampilkan Data Menggunakan SetPixel

i. Sifat

- ✓ Memerlukan deklarasi panggilan ke Window API
- ✓ Tidak bergantung setting warna sistem
- ✓ Tidak bergantung mode skala
- ✓ Tidak bergantung dari mode "Auto Redraw"
- ✓ Lebih cepat

ii. Masukkan Program berikut pada Module1

```

Option Explicit

' Deklarasi Jenis type Data RGB, untuk keperluan Image Processing
Public Type tRGB24
    B As Byte
    G As Byte
    R As Byte
End Type

Public Declare Function SetPixel Lib "gdi32" ( _
    ByVal hdc As Long, ByVal x As Long, ByVal y As Long, _
    ByVal crColor As Long) As Long
Public Declare Function GetPixel Lib "gdi32" ( _
    ByVal hdc As Long, ByVal x As Long, ByVal y As Long) As Long

Global vImage(0 To 319, 0 To 239) As tRGB24

```

iii. Masukkan Program berikut pada Form1

```

Option Explicit

Private Sub Dir1_Change()
    File1.Path = Dir1.Path
End Sub

```

```

Private Sub Drive1_Change()
    Dir1.Path = Drive1.Drive
End Sub

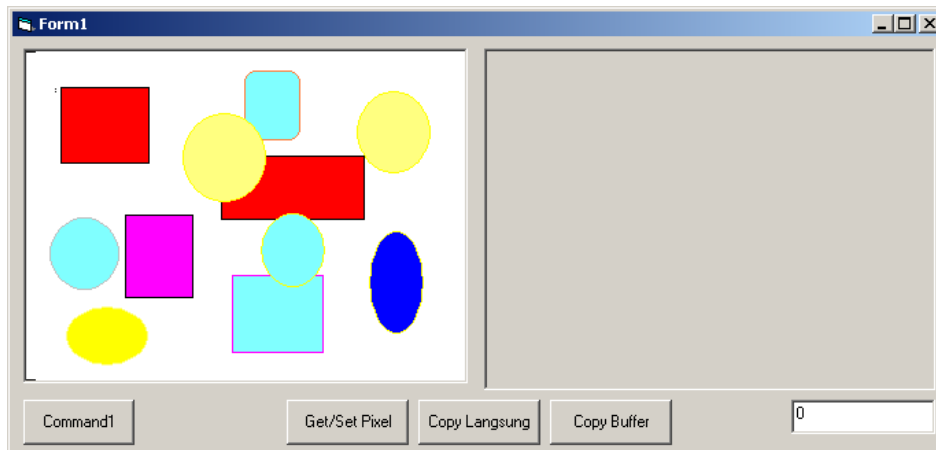
Private Sub File1_Click()
    Picture1.Picture = LoadPicture(File1.Path + "\" + File1.FileName)
    Dim y As Integer, x As Integer
    Dim p As Long
    For y = 0 To Picture1.ScaleHeight - 1
        For x = 0 To Picture1.ScaleWidth - 1
            p = GetPixel(Picture1.hdc, x, y)
            vImage(x, y).R = p And &HFF
            vImage(x, y).G = (p \ &H100) And &HFF
            vImage(x, y).B = (p \ &H10000) And &HFF
        Next
    Next
    For y = 0 To Picture1.ScaleHeight - 1
        For x = 0 To Picture1.ScaleWidth - 1
            vImage(x, y).B = 0
        Next
    Next
    For y = 0 To Picture1.ScaleHeight - 1
        For x = 0 To Picture1.ScaleWidth - 1
            SetPixel Picture1.hdc, x, y, RGB(vImage(x, y).R, vImage(x, y).G,
vImage(x, y).B)
        Next
    Next
End Sub

Private Sub Form_Load()
    File1.Pattern = "*.bmp;*.jpg;*.jpeg;*.gif;*.tif"
    Picture1.ScaleMode = 3
End Sub

```

4. Menampilkan Data dari satu PictureBox ke PictureBox lainnya Menggunakan Copy Image

- a. Beberapa cara yang dapat digunakan
 - i. Point – Pset → Sangat lambat (lebih dari satu detik)
 - ii. GetPixel – SetPixel → Lebih cepat
 - iii. Copy Image – Buffer – Image → Sangat cepat
 - iv. Copy Image – Image → Paling cepat
- b. Pasang dua PictureBox pada Form1 dengan nama pSumber dan pTujuan
- c. Pasang beberapa Command Button dengan nama End, cmdGetSet, cmdCopyLangsung dan cmdCopyBuffer
- d. Pasang TextBox dengan nama Text1



e. Tambahkan Module1 pada Project dan masukkan program berikut

```
Option Explicit

Public Declare Function BitBlt Lib "gdi32" ( _
    ByVal hDestDC As Long, ByVal x As Long, ByVal y As Long, _
    ByVal nWidth As Long, ByVal nHeight As Long, ByVal hSrcDC As Long, _
    ByVal XSrc As Long, ByVal YSrc As Long, ByVal dwRop As Long) As Long
Public Declare Function SetPixel Lib "gdi32" ( _
    ByVal hdc As Long, ByVal x As Long, ByVal y As Long, _
    ByVal crColor As Long) As Long
Public Declare Function GetPixel Lib "gdi32" ( _
    ByVal hdc As Long, ByVal x As Long, ByVal y As Long) As Long
Public Declare Function CreateCompatibleDC Lib "gdi32" ( _
    ByVal hdc As Long) As Long
Public Declare Function DeleteDC Lib "gdi32" (ByVal hdc As Long) As Long
Public Declare Function SelectObject Lib "gdi32" ( _
    ByVal hdc As Long, ByVal hObject As Long) As Long
Public Declare Function GetObject Lib "gdi32" Alias "GetObjectA" ( _
    ByVal hObject As Long, ByVal nCount As Long, lpObject As Any) As Long
Public Declare Sub CopyMemory Lib "kernel32" Alias "RtlMoveMemory" ( _
    Destination As Any, Source As Any, ByVal Length As Long)
Declare Sub RtlMoveMemory Lib "kernel32" ( _
    ByVal hpvDest As Long, ByVal hpvSource As Long, ByVal cbCopy As Long)
Public Declare Function CreateBitmap Lib "gdi32" ( _
    ByVal nWidth As Long, ByVal nHeight As Long, ByVal nPlanes As Long, _
    ByVal nBitCount As Long, lpBits As Any) As Long

Public Type BITMAP '14 bytes
    bmType As Long
    bmWidth As Long
    bmHeight As Long
    bmWidthBytes As Long
    bmPlanes As Integer
    bmBitsPixel As Integer
    bmBits As Long
End Type

Type tRGB24
    b As Byte
    G As Byte
    R As Byte
End Type

Public VideoData(0 To 319, 0 To 239) As tRGB24
Public Data(0 To 319, 0 To 239) As tRGB24
Public Data2(0 To 319, 0 To 239) As tRGB24
```

f. Masukkan program berikut pada Form1

```

Option Explicit

Dim fps As Long
'Dim Data(320, 240, 3) As Byte
'Dim Data2(320, 240, 3) As Byte
Dim DataB() As Byte

Private Sub AmbilRGB(Data() As tRGB24, Sumber())
    n = 0
    For y = 0 To ly - 1
        For x = 0 To lx - 1
            Data(x, y).b = Sumber(n)
            Data(x, y).G = Sumber(n + 1)
            Data(x, y).R = Sumber(n + 2)
            n = n + 3
        Next
    Next
End Sub

Private Sub RGB16_24(Sumber() As Integer, Tujuan() As tRGB24)
    Dim a As Integer
    For y = 0 To 239
        For x = 0 To 319
            a = Sumber(x, y)
            Tujuan(x, y).R = (a \ &H400) And &H1F
            Tujuan(x, y).G = (a \ &H20) And &H1F
            Tujuan(x, y).B = a And &H1F
        Next
    Next
End Sub

Private Sub RGB24_16(Sumber() As tRGB24, Tujuan() As Integer)
    Dim a As Integer
    For y = 0 To 239
        For x = 0 To 319
            a = (Sumber(x, y).R \ 8) * &H400 + _
                (Sumber(x, y).G \ 8) * &H20 + Sumber(x, y).b \ 8
            Tujuan(x, y) = a
        Next
    Next
End Sub

Private Sub cmdCopyBuffer_Click()
    If cmdCopyBuffer.Caption <> "Selesai" Then
        cmdCopyBuffer.Caption = "Selesai"
        fps = 0
        Dim hdc As Long, hBuffer As Long
        Dim bmp As BITMAP, bmp2 As BITMAP
        GetObject pSumber.Picture.Handle, Len(bmp), bmp
        ReDim DataB(bmp.bmWidthBytes * bmp.bmHeight) As Byte
        RtlMoveMemory VarPtr(DataB(0)), bmp.bmBits, _
            bmp.bmWidthBytes * bmp.bmHeight
        'AmbilRGB Data, bmp.bmBits
        'CreateBitmap bmp.bmWidth, bmp.bmHeight, bmp.bmPlanes, _
            bmp.bmBitsPixel, bmp2.bmBits
        pTujuan.Picture = LoadPicture("coba.bmp")
        GetObject pTujuan.Picture.Handle, Len(bmp2), bmp2
        'Data(5, 5).G = 0
        While cmdCopyBuffer.Caption = "Selesai"
            RtlMoveMemory bmp2.bmBits, VarPtr(DataB(0)), _
                bmp.bmWidthBytes * bmp.bmHeight
            fps = fps + 1
            DoEvents
        Wend
    Else
        cmdCopyBuffer.Caption = "Copy Buffer"
    End If
End Sub

```

```

Private Sub cmdCopyLangsung_Click()
    If cmdCopyLangsung.Caption <> "Selesai" Then
        cmdCopyLangsung.Caption = "Selesai"
        fps = 0
        While cmdCopyLangsung.Caption = "Selesai"
            BitBlt pTujuan.hdc, 0, 0, pSumber.ScaleWidth, _
                pSumber.ScaleHeight, pSumber.hdc, 0, 0, &HCC0020
            fps = fps + 1
            DoEvents
        Wend
    Else
        cmdCopyLangsung.Caption = "Copy Langsung"
    End If
End Sub

Private Sub cmdGetSet_Click()
    If cmdGetSet.Caption <> "Selesai" Then
        Dim y As Integer, x As Integer, p As Long
        cmdGetSet.Caption = "Selesai"
        fps = 0
        While cmdGetSet.Caption = "Selesai"
            For y = 0 To pSumber.ScaleHeight - 1
                For x = 0 To pSumber.ScaleWidth - 1
                    p = GetPixel(pSumber.hdc, x, y)
                    SetPixel pTujuan.hdc, x, y, p
                Next
            Next
            fps = fps + 1
            DoEvents
        Wend
    Else
        cmdGetSet.Caption = "Get/Set Pixel"
    End If
End Sub

Private Sub End_Click()
    End
End Sub

Private Sub Form_Load()
    pTujuan.ScaleMode = 3
    pSumber.ScaleMode = 3
    pSumber.AutoSize = True
    pSumber.Picture = LoadPicture("coba.bmp")
End Sub

Private Sub Timer1_Timer()
    Text1 = fps
    fps = 0
End Sub

```

Tugas

1. Cara menampilkan mana yang lebih baik antara menampilkan ke Form, Image, PictureBox atau menggunakan PictureBox ? Jelaskan sifat masing-masing.
2. Apa peran dari penggunaan property Image1.Stretch, Picture1.AutoSize dan Picture1.AutoRedraw ?

3. Uji dengan teliti kecepatan dari penggunaan cara Point, GetPixel dan Copy Image. Mana yang lebih cepat ?
4. Uji dengan teliti kecepatan dari penggunaan cara Pset dan SetPixel. Mana yang lebih cepat ?
5. Coba program kamera tersebut dijadikan file "exe". Bandingkan kecepatan yang didapatkan antara saat dijalankan pada editor VB dan saat dijalankan langsung dari file "exe".