

LAMPIRAN A
PROGRAM PADA FORM HOME

```
Private Sub Command1_Click()  
Form1.Visible = False  
Form2.Visible = True  
If Dir(App.Path & "\seq XY\") = "" Then  
    MkDir (App.Path & "\seq XY\")  
End If  
If Dir(App.Path & "\seq XZ\") = "" Then  
    MkDir (App.Path & "\seq XZ\")  
End If  
End Sub
```

```
Private Sub Command2_Click()  
Form1.Visible = False  
Form3.Visible = True  
End Sub
```

```
Private Sub Command3_Click()  
End  
End Sub
```

LAMPIRAN B
PROGRAM PADA FORM EXTRACT

Option Explicit

Private Sub Command2_Click()

Form1.Visible = True

Form2.Visible = False

End Sub

Private Sub Form_Load()

Call AVIFileInit '// opens AVIFile library

End Sub

Private Sub Form_Unload(Cancel As Integer)

Call AVIFileExit '// releases AVIFile library

End Sub

Private Sub cmdOpenAVIFile1_Click()

Dim res As Long 'result code

Dim ofd As cFileDialog 'OpenFileDialog class

Dim szFile As String 'filename

Dim pAVIFile As Long 'pointer to AVI file interface (PAVIFILE handle)

Dim pAVIStream As Long 'pointer to AVI stream interface (PAVISTREAM handle)

Dim numFrames As Long 'number of frames in video stream

Dim firstFrame As Long 'position of the first video frame

Dim fileInfo As AVI_FILE_INFO 'file info struct

Dim streamInfo As AVI_STREAM_INFO 'stream info struct

Dim dib As cDIB

Dim pGetFrameObj As Long 'pointer to GetFrame interface

Dim pDIB As Long 'pointer to packed DIB in memory

Dim bih As BITMAPINFOHEADER 'infoheader to pass to GetFrame functions

Dim i As Long

```

'Get the name of an AVI file to work with
Set ofd = New cFileDialog
With ofd
    .OwnerHwnd = Me.hWnd
    .Filter = "AVI Files|*.avi"
    .DlgTitle = "Open AVI File"
End With
res = ofd.VBGetOpenFileNamePreview(szFile)
If res = False Then GoTo ErrorOut
'Open the AVI File and get a file interface pointer (PAVIFILE)
res = AVIFileOpen(pAVIFile, szFile, OF_SHARE_DENY_WRITE, 0&)
If res <> AVIERR_OK Then GoTo ErrorOut
'Get the first available video stream (PAVISTREAM)
res = AVIFileGetStream(pAVIFile, pAVIStream, streamtypeVIDEO, 0)
If res <> AVIERR_OK Then GoTo ErrorOut
'get the starting position of the stream (some streams may not start
simultaneously)
firstFrame = AVIStreamStart(pAVIStream)
If firstFrame = -1 Then GoTo ErrorOut 'this function returns -1 on error
'get the length of video stream in frames
numFrames = AVIStreamLength(pAVIStream)
If numFrames = -1 Then GoTo ErrorOut ' this function returns -1 on error
' MsgBox "PAVISTREAM handle is " & pAVIStream & vbCrLf & _
'     "Video stream length - " & numFrames & vbCrLf & _
'     "Stream starts on frame #" & firstFrame & vbCrLf & _
'     "File and Stream info will be written to Immediate Window (from IDE -
Ctrl+G to view)", vbInformation, App.title
'get file info struct (UDT)
res = AVIFileInfo(pAVIFile, fileInfo, Len(fileInfo))
If res <> AVIERR_OK Then GoTo ErrorOut

```

```

' 'print file info to Debug Window
' Call DebugPrintAVIFileInfo(fileInfo)
'get stream info struct (UDT)
res = AVIStreamInfo(pAVIStream, streamInfo, Len(streamInfo))
If res <> AVIERR_OK Then GoTo ErrorOut
' 'print stream info to Debug Window
' Call DebugPrintAVIStreamInfo(streamInfo)
'set bih attributes which we want GetFrame functions to return
With bih
    .biBitCount = 24
    .biClrImportant = 0
    .biClrUsed = 0
    .biCompression = BI_RGB
    .biHeight = streamInfo.rcFrame.bottom - streamInfo.rcFrame.top
    .biPlanes = 1
    .biSize = 40
    .biWidth = streamInfo.rcFrame.right - streamInfo.rcFrame.left
    .biXPelsPerMeter = 0
    .biYPelsPerMeter = 0
    .biSizeImage = (((.biWidth * 3) + 3) And &HFFFC) * .biHeight 'calculate
total size of RGBQUAD scanlines (DWORD aligned)
End With
'init AVIStreamGetFrame* functions and create GETFRAME object
'pGetFrameObj = AVIStreamGetFrameOpen(pAVIStream, ByVal
AVIGETFRAMEF_BESTDISPLAYFMT) 'tell AVIStream API what format we
expect and input stream
pGetFrameObj = AVIStreamGetFrameOpen(pAVIStream, bih) 'force function
to return 24bit DIBS
If pGetFrameObj = 0 Then
    MsgBox "No suitable decompressor found for this video stream!",
vbInformation, App.title

```

```

        GoTo ErrorOut
    End If

    'create a DIB class to load the frames into
    Set dib = New cDIB

    For i = firstFrame To (numFrames - 1) + firstFrame
        pDIB = AVIStreamGetFrame(pGetFrameObj, i) 'returns "packed DIB"
        If dib.CreateFromPackedDIBPointer(pDIB) Then
            Call dib.WriteToFile(App.Path & "\seq XY\" & Format(i, "0000") &
".bmp")
            txtStatus = "Bitmap " & i + 1 & " of " & numFrames & " written to app
folder"
            txtStatus.Refresh
        Else
            End If
        Next
        Set dib = Nothing
    End If

ErrorOut:
    If pGetFrameObj <> 0 Then
        Call AVIStreamGetFrameClose(pGetFrameObj) '//deallocates the GetFrame
resources and interface
    End If
    If pAVIStream <> 0 Then
        Call AVIStreamRelease(pAVIStream) '//closes video stream
    End If
    If pAVIFile <> 0 Then
        Call AVIFileRelease(pAVIFile) '// closes the file
    End If
    If (res <> AVIERR_OK) Then 'if there was an error then show feedback to user
        MsgBox "There was an error working with the file:" & vbCrLf & szFile,
vbInformation, App.title
    End If

```

End Sub

Private Sub cmdOpenAVIFile2_Click()

Dim res As Long 'result code

Dim ofd As cFileDialog 'OpenFileDialog class

Dim szFile As String 'filename

Dim pAVIFile As Long 'pointer to AVI file interface (PAVIFILE handle)

Dim pAVIStream As Long 'pointer to AVI stream interface (PAVISTREAM handle)

Dim numFrames As Long 'number of frames in video stream

Dim firstFrame As Long 'position of the first video frame

Dim fileInfo As AVI_FILE_INFO 'file info struct

Dim streamInfo As AVI_STREAM_INFO 'stream info struct

Dim dib As cDIB

Dim pGetFrameObj As Long 'pointer to GetFrame interface

Dim pDIB As Long 'pointer to packed DIB in memory

Dim bih As BITMAPINFOHEADER 'infoheader to pass to GetFrame functions

Dim i As Long

'Get the name of an AVI file to work with

Set ofd = New cFileDialog

With ofd

.OwnerHwnd = Me.hWnd

.Filter = "AVI Files|*.avi"

.DlgTitle = "Open AVI File"

End With

res = ofd.VBGetOpenFileNamePreview(szFile)

If res = False Then GoTo ErrorOut

'Open the AVI File and get a file interface pointer (PAVIFILE)

res = AVIFileOpen(pAVIFile, szFile, OF_SHARE_DENY_WRITE, 0&)

If res <> AVIERR_OK Then GoTo ErrorOut

```

'Get the first available video stream (PAVISTREAM)
res = AVIFileGetStream(pAVIFile, pAVIStream, streamtypeVIDEO, 0)
If res <> AVIERR_OK Then GoTo ErrorOut
'get the starting position of the stream (some streams may not start
simultaneously)
firstFrame = AVIStreamStart(pAVIStream)
If firstFrame = -1 Then GoTo ErrorOut 'this function returns -1 on error
'get the length of video stream in frames
numFrames = AVIStreamLength(pAVIStream)
If numFrames = -1 Then GoTo ErrorOut 'this function returns -1 on error
' MsgBox "PAVISTREAM handle is " & pAVIStream & vbCrLf & _
'      "Video stream length - " & numFrames & vbCrLf & _
'      "Stream starts on frame #" & firstFrame & vbCrLf & _
'      "File and Stream info will be written to Immediate Window (from IDE -
Ctrl+G to view)", vbInformation, App.title
'get file info struct (UDT)
res = AVIFileInfo(pAVIFile, fileInfo, Len(fileInfo))
If res <> AVIERR_OK Then GoTo ErrorOut
' 'print file info to Debug Window
' Call DebugPrintAVIFileInfo(fileInfo)
'get stream info struct (UDT)
res = AVIStreamInfo(pAVIStream, streamInfo, Len(streamInfo))
If res <> AVIERR_OK Then GoTo ErrorOut
' 'print stream info to Debug Window
' Call DebugPrintAVIStreamInfo(streamInfo)
'set bih attributes which we want GetFrame functions to return
With bih
    .biBitCount = 24
    .biClrImportant = 0
    .biClrUsed = 0
    .biCompression = BI_RGB

```

```

.biHeight = streamInfo.rcFrame.bottom - streamInfo.rcFrame.top
.biPlanes = 1
.biSize = 40
.biWidth = streamInfo.rcFrame.right - streamInfo.rcFrame.left
.biXPelsPerMeter = 0
.biYPelsPerMeter = 0
.biSizeImage = (((.biWidth * 3) + 3) And &HFFFC) * .biHeight 'calculate
total size of RGBQUAD scanlines (DWORD aligned)
End With
'init AVIStreamGetFrame* functions and create GETFRAME object
pGetFrameObj = AVIStreamGetFrameOpen(pAVIStream, ByVal
AVIGETFRAMEF_BESTDISPLAYFMT) 'tell AVIStream API what format we
expect and input stream
pGetFrameObj = AVIStreamGetFrameOpen(pAVIStream, bih) 'force function
to return 24bit DIBS
If pGetFrameObj = 0 Then
    MsgBox "No suitable decompressor found for this video stream!",
vbInformation, App.title
    GoTo ErrorOut
End If
'create a DIB class to load the frames into
Set dib = New cDIB
For i = firstFrame To (numFrames - 1) + firstFrame
    pDIB = AVIStreamGetFrame(pGetFrameObj, i) 'returns "packed DIB"
    If dib.CreateFromPackedDIBPointer(pDIB) Then
        Call dib.WriteToFile(App.Path & "\seq XZ\" & Format(i, "0000") &
".bmp")
        txtStatus = "Bitmap " & i + 1 & " of " & numFrames & " written to app
folder"
        txtStatus.Refresh
    Else

```

```

        End If
    Next
    Set dib = Nothing
ErrorOut:
    If pGetFrameObj <> 0 Then
        Call AVIStreamGetFrameClose(pGetFrameObj) '//deallocates the GetFrame
resources and interface
    End If
    If pAVIStream <> 0 Then
        Call AVIStreamRelease(pAVIStream) '//closes video stream
    End If
    If pAVIFile <> 0 Then
        Call AVIFileRelease(pAVIFile) '// closes the file
    End If
    If (res <> AVIERR_OK) Then 'if there was an error then show feedback to user
        MsgBox "There was an error working with the file:" & vbCrLf & szFile,
vbInformation, App.title
    End If
End Sub

```

LAMPIRAN C
PROGRAM PADA FORM ANALYZE

```

Private Sub Check1_Click()
If Check1.Value = 0 Then
    Text1.Enabled = True
    Text2.Enabled = True
    Text3.Enabled = True
    Text4.Enabled = True
    Text5.Enabled = True
    Text6.Enabled = True
ElseIf Check1.Value = 1 Then
    Text1.Enabled = False
    Text2.Enabled = False
    Text3.Enabled = False
    Text4.Enabled = False
    Text5.Enabled = False
    Text6.Enabled = False
End If
End Sub

```

```

Private Sub Command1_Click()
Dim sFile As String
List1.Clear
List2.Clear
Command1.Enabled = False
Command3.Enabled = True
mypath = App.Path
sFile = Dir(mypath & "\seq XY\")
    Do While sFile <> ""
        List1.AddItem sFile
        sFile = Dir
    Loop
mypath = App.Path

```

```

sFile = Dir(mypath & "\seq XZ\")
    Do While sFile <> ""
        List2.AddItem sFile
        sFile = Dir
    Loop
End Sub

```

```

Private Sub Command2_Click()
Label11.Caption = Last - First
End Sub

```

```

Private Sub Command3_Click()
List1.Clear
List2.Clear
Command1.Enabled = True
Command3.Enabled = False
End Sub

```

```

Private Sub Command5_Click()
X = Picture1.ScaleWidth
Y = Picture1.ScaleHeight
Z = Picture2.ScaleWidth
W = Picture2.ScaleHeight
ProgressBar1.Value = 0
Label10 = 0
'Menentukan nilai awal first frame dan last frame
If Text4.Text = "First" Then
First = 0
ElseIf Text4.Text <> "First" Then
First = Text4.Text

```

```

ElseIf Text4.Text > List1.ListCount Or Text5.Text > List2.ListCount Or
Text5.Text > Text4.Text Then
MsgBox "Please check again your First/Last frame in the File Information !"
End If
If Text5.Text = "Last" Then
Last = List1.ListCount
ElseIf Text5.Text <> "Last" Then
Last = Text5.Text
End If
'Membuat dan mengisi File *.CSM yang ada pada folder aplikasi
Open App.Path & "\" & Text1.Text & ".csm" For Output As #1
Print #1, "$Date"; " "; Date
Print #1, "$Time"; " "; Time
Print #1, "$Filename"; " "; Text1.Text
Print #1, "$Actor"; " "; Text2.Text
Print #1, ""
Print #1, "$Comments"
Print #1, Text3.Text
Print #1, ""
Print #1, "$FirstFrame"; " "; First
Print #1, "$LastFrame"; " "; Last
Print #1, "$Rate"; " "; Text6.Text
Print #1, "$Order"
Print #1, "LFHD LBHD RBHD RFHD CLAV STRN LFWT LBWT RBWT
RFWT C7 T10 LKNE LANK LMT5 LTOE RKNE RANK RMT5 RTOE LSHO
LELB LWRE LWRI LFIN RSHO RELB RWRE RWRI RFIN"
Print #1, ""
Print #1, "$Points"
For i = First To Last - 1
    List1.ListIndex = i
    List2.ListIndex = i

```

```

Picture1.Picture = LoadPicture(App.Path & "\seq XY\" & Format(i, "0000") &
".bmp")
Picture2.Picture = LoadPicture(App.Path & "\seq XZ\" & Format(i, "0000") &
".bmp")
'scanning pixel picture1
For brs1 = 1 To Y Step 15
For klm1 = 1 To X Step 15
wrn1 = Picture1.Point(klm1, brs1)
r1 = wrn1 And RGB(255, 0, 0)
g1 = Int((wrn1 And RGB(0, 255, 0)) / 256)
b1 = Int(Int((wrn1 And RGB(0, 0, 255)) / 256) / 256)
Picture1.PSet (klm1, brs1), RGB(r1, g1, b1)
If r1 > 110 And g1 < 50 And b1 < 50 Then
Picture1.PSet (klm1, brs1), RGB(255, 0, 0)
X1 = Round((((klm1 / 15) - 222) * 1) + 30, 2)
Y1 = Round((((brs1 / 15) - 358) * 1) - 1, 2)
ElseIf r1 < 50 And g1 > 70 And b1 < 60 Then
Picture1.PSet (klm1, brs1), RGB(0, 255, 0)
X2 = Round((((klm1 / 15) - 222) * 1) + 30, 2)
Y2 = Round((((brs1 / 15) - 358) * 1) - 1, 2)
ElseIf r1 > 50 And g1 > 70 And b1 < 60 Then
Picture1.PSet (klm1, brs1), RGB(255, 255, 0)
X3 = Round((((klm1 / 15) - 222) * 1) + 30, 2)
Y3 = Round((((brs1 / 15) - 358) * 1) - 1, 2)
ElseIf r1 < 70 And g1 < 70 And b1 > 70 Then
Picture1.PSet (klm1, brs1), RGB(0, 0, 255)
X4 = Round((((klm1 / 15) - 222) * 1) + 30, 2)
Y4 = Round((((brs1 / 15) - 358) * 1) - 1, 2)
ElseIf r1 > 90 And g1 > 90 And b1 > 90 Then
Picture1.PSet (klm1, brs1), RGB(255, 255, 225)
X5 = Round((((klm1 / 15) - 222) * 1) + 30, 2)

```

```

        Y5 = Round((((brs1 / 15) - 358) * 1) - 1, 2)
    Else
        Picture1.PSet (klm1, brs1), RGB(0, 0, 0)
    End If
Next klm1: load
Next brs1
'scanning pixel picture2
For brs2 = 1 To W Step 15
For klm2 = 1 To Z Step 15
    wrn2 = Picture2.Point(klm2, brs2)
    r2 = wrn2 And RGB(255, 0, 0)
    g2 = Int((wrn2 And RGB(0, 255, 0)) / 256)
    b2 = Int(Int((wrn2 And RGB(0, 0, 255)) / 256) / 256)
    Picture2.PSet (klm2, brs2), RGB(r2, g2, b2)
    If r2 > 110 And g2 < 50 And b2 < 50 Then
        Picture2.PSet (klm2, brs2), RGB(255, 0, 0)
        z1 = Round((((klm2 / 15) - 158) * 1) - 14, 2)
    ElseIf r2 < 50 And g2 > 70 And b2 < 60 Then
        Picture2.PSet (klm2, brs2), RGB(0, 255, 0)
        z2 = Round((((klm2 / 15) - 158) * 1) - 14, 2)
    ElseIf r2 > 50 And g2 > 70 And b2 < 60 Then
        Picture2.PSet (klm2, brs2), RGB(255, 255, 0)
        z3 = Round((((klm2 / 15) - 158) * 1) - 14, 2)
    ElseIf r2 < 70 And g2 < 70 And b2 > 70 Then
        Picture2.PSet (klm2, brs2), RGB(0, 0, 255)
        z4 = Round((((klm2 / 15) - 158) * 1) - 14, 2)
    ElseIf r2 > 90 And g2 > 90 And b2 > 90 Then
        Picture2.PSet (klm2, brs2), RGB(255, 255, 225)
        z5 = Round((((klm2 / 15) - 158) * 1) - 14, 2)
    Else
        Picture2.PSet (klm2, brs2), RGB(0, 0, 0)
    End If

```

```

        End If
    Next klm2: load
    Next brs2
    Picture1.Refresh
    Picture2.Refresh

    Print #1, CStr(i); " "; "10 -309 11 10 -309 -23 -11 -309 -23 -11 -309 11 0 -
262 3 0 -236 21 20 -183 9 20 -183 -19 -20 -183 -19 -20 -183 9 0 -262 -29 0 -
236 -31 54 -84 -19 75 -8 -19 31 -129 11 65 0 2 -55 -84 -19 -77 -8 -19 -32 -
129 11 -66 0 2 30 -264 -7 76 -249 -15 120 -258 -8 123 -242 -8 141 -254 -1 ";
CStr(X1); " "; CStr(Y1); " "; CStr(z1); " "; CStr(X2); " "; CStr(Y2); " "; CStr(z2);
" "; CStr(X3); " "; CStr(Y3); " "; CStr(z3); " "; CStr(X4); " "; CStr(Y4); " ";
CStr(z4); " "; CStr(X5); " "; CStr(Y5); " "; CStr(z5)

    'Print #1, CStr(i); " "; CStr(X1); " "; CStr(Y1); " "; CStr(z1); " "; CStr(X2); "
"; CStr(Y2); " "; CStr(z2); " "; CStr(X3); " "; CStr(Y3); " "; CStr(z3); " ";
CStr(X4); " "; CStr(Y4); " "; CStr(z4); " "; CStr(X5); " "; CStr(Y5); " "; CStr(z5)

    'Print #1, CStr(i); ""; CStr(x5); CStr(y5)
Next
Close #1
'delay
For k = 0 To 15000000
    k = k + 1
Next
Label1.Caption = "It's Done !"
Shell "notepad.exe " & (App.Path & "\" & Text1.Text & ".csm"), vbNormalFocus
End Sub

Private Sub Command6_Click()
    Form1.Visible = True
    Form3.Visible = False
End Sub

```

```
Private Sub Form_Load()  
    Command3.Enabled = False  
    Label1.FontSize = 8  
End Sub
```

```
Private Sub Text1_Click()  
    Text1.Text = ""  
End Sub
```

```
Private Sub Text2_Click()  
    Text2.Text = ""  
End Sub
```

```
Private Sub Text3_Click()  
    Text3.Text = ""  
End Sub
```

```
Private Sub Text4_Click()  
    Text4.Text = ""  
End Sub
```

```
Private Sub Text5_Click()  
    Text5.Text = ""  
End Sub
```

```
Private Sub Text6_Click()  
    Text6.Text = ""  
End Sub
```

```

Sub load()
'Menentukan nilai awal first frame dan last frame
If Text4.Text = "First" Then
First = 0
ElseIf Text4.Text <> "First" Then
First = Text4.Text
End If
If Text5.Text = "Last" Then
Last = List2.ListCount
ElseIf Text5.Text <> "Last" Then
Last = Text5.Text
End If
Label10 = Label10 + 1: Label10.Refresh
Sum = (Label10 / ((Last - First) * 480)) * 100
Label1 = Round(Sum, 2) & " %": Label1.Refresh
ProgressBar1.Value = Sum
End Sub

```