

00:01:45:00

• Dahlan Abdullah • Fadlisyah

Strips Show

Bass D ✓ Bass DI

Vocal ✓ Vocal

Clap ✓

Keys L ✓

Keys R ✓

AC Gtr ✓

E Gtr L ✓

E Gtr L ✓

F Gtr R ✓

Group Show

Keys ✓

Guitar ✓

Mute Solo Mute Solo Mute

-0.0 -9.6 -0.0 -10.7 -0.0

-3 -6 -3 -6 -3

-5 -10 -5 -10 -5

-10 -15 -10 -15 -10

-15 -18 -15 -18 -15

-20 -25 -20 -25 -20

-25 -30 -25 -30 -25

-30 -35 -30 -35 -30

-35 -40 -35 -40 -35

M C Pkt Grp

Master

Comments

File

File

Guitar Rig 5 FX

AUDynamicsPro

compression threshold

expansion ratio

attack time

release time

use attack gate

fader

# PENGOLAHAN CITRA & SUARA



# **PENGOLAHAN CITRA & SUARA**

**DAHLAN ABDULLAH  
FADLISYAH**



**2020**

# **PENGOLAHAN CITRA & SUARA**

Hak Cipta©2020 pada

## **Penulis**

Dahlan Abdullah

Fadlisyah

## **Cover Design**

*T.M. Siddiqi(SEFA)*

## **Layout**

*Rizka Indriani(SEFA)*

Pracetak dan Produksi

*CV.Sefa Bumi Persada*

*Hak Cipta dilindungi undang-undang.*

*Dilarang memperbanyak atau memindahkan sebagian atau seluruh  
isi buku ini dalam bentuk apapun,baik secara elektronis maupun  
mekanis,termasuk memfotokopi,merekam atau dengan sistem  
penyimpanan lainnya,tanpa izin tertulis dari Penulis*

Penerbit:

**SEFA BUMI PERSADA**

**Anggota IKAPI:No.021/DIA/2018**

Jl.B.Aceh-Medan,Alue Awe-Lhokseumawe

email:[sefabumipersada@gmail.com](mailto:sefabumipersada@gmail.com)

*Telp.085260363550*

*Cetakan I:2020*

**ISBN-978-623-7648-18-5**

1.Hal.137:16,5 X 7,5 cm

I.Judul

## **KATA PENGANTAR**

Prinsip pengembangan perangkat lunak otomatisasi adalah selaras dengan prinsip-prinsip yang ditekankan pada era revolusi industri 4.0., artinya perangkat lunak pada era 4.0 sudah mampu meminimisir peran manusia baik dalam pengembangannya, kemampuan belajar, ataupun operasional-operasional teknis lainnya.

Untuk menjawab tantangan model 4.0, maka buku ini memfasilitasi tentang cara-cara pembuatan perangkat lunak otomatisasi untuk pengolahan citra dan suara yang diterapkan untuk berbagai aplikasi yang terkait pemanfaatan Al Qur'an secara digital.

Adapun materi yang hadir di dalam buku ini meliputi : Resizing, Model, Integral, dan berbagai kasus yang menjadi trend untuk dibuatkan perangkat lunaknya secara 4.0.

Penulis

## DAFTAR ISI

<b>KATA PENGANTAR</b>	<b>iii</b>
<b>DAFTAR ISI</b>	<b>iv</b>
<b>BAB 1 – RESIZING</b>	
1.1 Pendahuluan	1
1.2 Perangkat Lunak Resizing	1
1.3 Listing Keseluruhan	5
<b>BAB 2 – DETEKSI POLA TAJWID</b>	
2.1 Pendahuluan	11
2.2 Pola Hukum Tajwid	12
2.3 Skema Sistem	17
2.4 Pembangunan Perangkat Lunak Deteksi Tajwid	18
2.5 Listing Keseluruhan	34
<b>BAB 3 – ALGORITMA SIMILARITY BINER DAN DISTANCE UNTUK PENGENALAN POLA</b>	
3.1 Pendahuluan	49
3.2 Dafta Algoritma Similarity Biner Dan Pengukuran Distance	50
3.3 Deteksi Tajwid Menggunakan Algoritma Similarity Biner Kulczynski	57
<b>BAB 4 – TRANSFORTASI INTEGRAL</b>	
4.1 Transforsiasi Fourier	81
4.2 Transformasi Sinyal lainnya	85
4.3 Terapan Sistem	86
<b>BAB 5 – SISTEM WAKTU NYATA PENGUJIAN HAFALAN AYAT SUCI AL-QURAN</b>	
5.1 Pendahuluan	107
5.2 Skema sistem	112
5.3 Hasil	114
5.4 Implementasi Sistem	116

**BAB 6 – SISTEM WAKTU NYATA PENGUJIAN  
MENGGUNAKAN TRANSFORMASI NON-SINUSOIDAL**

6.1 Transformasi Hadamard	131
6.2 Transformasi Slant	134

<b>DAFTAR PUSTAKA</b>	137
-----------------------	-----



# **BAB 1**

## **RESIZING**

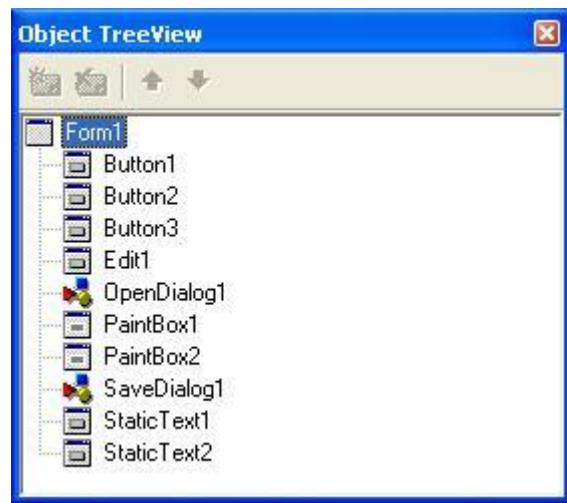
### **1.1 Pendahuluan**

Resizing adalah salah satu proses awal penting di dalam sebuah skema pemrosesan citra. Resizing berfungsi sebagai pengkondisian ukuran awal sebuah citra masukan, sebelum citra tersebut dikomputasi lebih lanjut. Dengan mengkondisikan ukuran citra melalui resizing, maka berbagai citra masukan yang memiliki sebarang ukuran tertentu, diharapkan dapat lebih sesuai diterapkan untuk berbagai komputasi yang digunakan user dalam pengenalan pola. Banyak pendekatan yang digunakan untuk melakukan resizing pada citra, dan di dalam pembahasan bab materi ini, pemetaan memori digunakan untuk membangun sistem resizing citra. Selanjutnya sistem resizing citra yang dibangun akan selalu dominan muncul sebagai proses awal dalam berbagai sistem pengenalan pola pada citra yang akan dibahas dalam materi-materi selanjutnya.

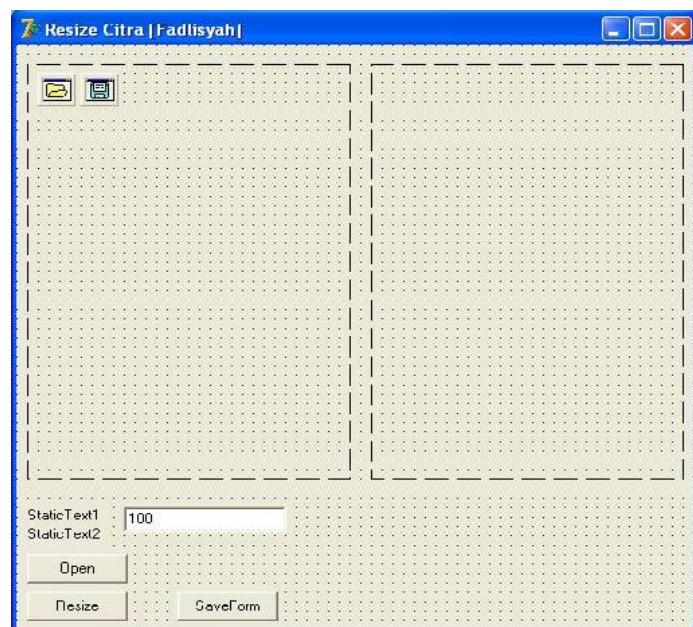
### **1.2 Pembangunan Perangkat Lunak Resizing**

Untuk membangun sistem resizing citra, maka langkah-langkah kerja yang harus diikuti adalah sebagai berikut :

1. Jalankan Delphi, setelah muncul halaman Form1, maka tambahan komponen-komponen berikut :



2. Atur sedemikian rupa agar desain tampilan Form1 akan berupa tampilan berikut :



3. Klik 2x pada Button1, dan tuliskan kode program di bawah ini:

```
procedure TForm1.Button1Click(Sender:  
 TObject);  
 //load picture  
 begin
```

```

opendialog1.filter := 'bitmaps | *bmp' ;
if opendialog1.execute then
begin
  citral.loadfromfile(opendialog1.filename);
  citral.pixelformat := pf24bit; //in case
other format loaded
end;
with paintbox1 do
  with canvas do
begin
  brush.color := 0;
  fillrect(rect(0,0,width,height));
  draw(0,0,citral);
end;
sw := citral.width;
sh := citral.height;
statictext1.caption := inttostr(sw);
statictext2.caption := inttostr(sh);
end;

```

4. Klik 2x pada Button2, dan tuliskan kode program di bawah ini:

```

procedure TForm1.Button2Click(Sender:
TObject);
//reduce or enlarge
var sx1,sy1,sx2,sy2 : single;           //source
field positions
  x,y : word;                           //dest field
pixels
  destR,destG,destB : single;           //destination colors
  sR,sG,sB : byte;                     //source
colors
  destWidth, destheight : word;
  f,fi2 : single;
  i,j : word;
  dx,dy,PC : single;
  color : longInt;
begin
  with paintbox2 do
    with canvas do
      fillrect(rect(0,0,width,height));

```

```

destwidth := strtoint(edit1.text);
f := sw / destwidth;
fi2 := 1/f;
fi2 := fi2*fi2;
destheight := trunc(citral.height/f);
with citra2 do
begin
  width := destwidth;
  height := destheight;
end;
//---
for y := 0 to destheight-1 do
//vertical destination pixels
begin
  syl := f * y;
  sy2 := syl + f;
  for x := 0 to destwidth-1 do
//horizontal destination pixels
begin
  sx1 := f * x;
  sx2 := sx1 + f;
  destR := 0; destG := 0; destB := 0;
//clear colors
  for j := floor(syl) to ceiling(sy2)-1 do
//vertical source pixels
begin
  dy := 1;
  if syl > j then begin
    dy := dy-(syl-j);
  end;
  if sy2 < j+1 then begin
    dy := dy-(j+1-sy2);
  end;
  for i := floor(sx1) to ceiling(sx2)-1
do //horizontal source pixels
begin
  dx := 1;
  if sx1 > i then begin
    dx := dx-(sx1-i);
  end;
  if sx2 < i+1 then begin
    dx := dx-(i+1-
sx2);
  end;
  color := citral.canvas.pixels[i,j];

```

```

    sR := color and $ff;
    sG := (color shr 8) and $ff;
    sB := (color shr 16) and $ff;
    PC := dx*dy*fi2;
    destR := destR + sR*PC;
    destG := destG + sG*PC;
    destB := destB + sB*PC;
    end;//for i
    end;//for j
    citra2.Canvas.Pixels[x,y]           :=
RGB(trunc(destR),trunc(destG),trunc(destB));
    end;//for x
    end;//for y
    paintbox2.Canvas.Draw(0,0,citra2);
    citra2.SaveToFile('resizinghasil.bmp');
end;

```

5. Klik 2x pada Button3, dan tuliskan kode program di bawah ini:

```

procedure TForm1.Button3Click(Sender:
TObject);
var formImage : Tbitmap;
begin
  formimage := Tbitmap.create;
  formimage := getformimage;
  formImage.canvas.draw(paintbox1.left,paintbox
1.top,citra1);
  formImage.canvas.draw(paintbox2.left,paintbox
2.top,citra2);
  if savedialog1.execute then
    formImage.savetofile(savedialog1.filename);
    formImage.free;
end;

```

### 1.3 Listing Keseluruhan

Setelah langkah-langkah di atas dikerjakan, maka langkah terakhir adalah memastikan keseluruhan kode program akan berupa listing kode berikut :

```

unit Unit1;
interface

```

```

uses
  Windows, Messages, SysUtils, Variants, Classes,
  Graphics, Controls, Forms,
  Dialogs, ExtCtrls, StdCtrls, ExtDlgs;
type
  TForm1 = class(TForm)
    Button1: TButton;
    Button2: TButton;
    StaticText1: TStaticText;
    StaticText2: TStaticText;
    PaintBox1: TPaintBox;
    PaintBox2: TPaintBox;
    OpenDialog1: TOpenDialog;
    SaveDialog1: TSaveDialog;
    Edit1: TEdit;
    Button3: TButton;
    procedure Button1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    procedure Button3Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
var
  Form1: TForm1;
implementation
{$R *.dfm}
var citral,citra2 : TBitmap;
  sw,sh : word;
procedure TForm1.Button1Click(Sender: TObject);
//load picture
begin
  opendialog1.filter := 'bitmaps | *bmp' ;
  if opendialog1.execute then
  begin
    citral.loadfromfile(opendialog1.filename);
    citral.pixelformat := pf24bit; //in case other
format loaded
  end;
  with paintbox1 do
    with canvas do
      begin
        brush.color := 0;
        fillrect(rect(0,0,width,height));

```

```

        draw(0,0,citra1);
    end;
sw := citra1.width;
sh := citra1.height;
statictext1.caption := inttostr(sw);
statictext2.caption := inttostr(sh);
end;
function floor(a : single) : word;
//return a rounded down to integer
begin
    result := trunc(a);
end;
function ceiling(a : single) : word;
//return a rounded up to integer
begin
    result := trunc(a);
    if frac(a) > 0.0001 then inc(result);
//fix acces violation if small fraction
end;
procedure TForm1.Button2Click(Sender: TObject);
//reduce or enlarge
var sx1,sy1,sx2,sy2 : single;           //source field
pixels
    x,y : word;                         //dest field
destR,destG,destB : single;           //destination
colors
    sR,sG,sB : byte;                  //source colors
    destWidth, destheight : word;
    f,fi2 : single;
    i,j : word;
    dx,dy,PC : single;
    color : longInt;
begin
    with paintbox2 do
        with canvas do fillrect(rect(0,0,width,height));
destwidth := strtoint(edit1.text);
f := sw / destwidth;
fi2 := 1/f;
fi2 := fi2*fi2;
destheight := trunc(citra1.height/f);
with citra2 do
begin
    width := destwidth;
    height := destheight;

```

```

        end;
//---
    for y := 0 to destheight-1 do           //vertical
destination pixels
begin
    sy1 := f * y;
    sy2 := sy1 + f;
    for x := 0 to destwidth-1 do
//horizontal destination pixels
begin
    sx1 := f * x;
    sx2 := sx1 + f;
    destR := 0; destG := 0; destB := 0;
//clear colors
    for j := floor(sy1) to ceiling(sy2)-1 do
//vertical source pixels
begin
    dy := 1;
    if sy1 > j then begin
        dy := dy-(sy1-j);
    end;
    if sy2 < j+1 then begin
        dy := dy-(j+1-sy2);
    end;
    for i := floor(sx1) to ceiling(sx2)-1 do
//horizontal source pixels
begin
    dx := 1;
    if sx1 > i then begin
        dx := dx-(sx1-i);
    end;
    if sx2 < i+1 then begin
        dx := dx-(i+1-sx2);
    end;
    color := citral.canvas.pixels[i,j];
    sR := color and $ff;
    sG := (color shr 8) and $ff;
    sB := (color shr 16) and $ff;
    PC := dx*dy*fi2;
    destR := destR + sR*PC;
    destG := destG + sG*PC;
    destB := destB + sB*PC;
end;//for i
end;//for j
citra2.Canvas.pixels[x,y] :=

```

```

RGB(trunc(destR),trunc(destG),trunc(destB));
end;//for x
end;//for y
paintbox2.canvas.draw(0,0,citra2);
citra2.SaveToFile('resizinghasil.bmp');
end;
procedure TForm1.Button3Click(Sender: TObject);
var formImage : Tbitmap;
begin
  formimage := Tbitmap.create;
  formimage           :=      getformimage;
formImage.canvas.draw(paintbox1.left,paintbox1.top
,citra1);

formImage.canvas.draw(paintbox2.left,paintbox2.top
,citra2);
  if savedialog1.execute then
    formImage.savetofile(savedialog1.filename);
  formImage.free;
end;
initialization
  citra1 := Tbitmap.create;
  with citra1 do
  begin
    pixelformat := pf24bit;
    width := 100; height := 100;
  end;
  citra2 := TBitmap.create;
  with citra2 do
  begin
    pixelformat := pf24bit;
    width := 100; height := 100;
  end;
  sw := 100; sh := 100;
finalization
  citra1.free;
  citra2.free;
end.

```

dan tampilan hasil eksekusinya adalah :



## BAB 2

# DETEKSI POLA TAJWID

### 2.1 Pendahuluan

Tajwid secara harfiah bermakna melakukan sesuatu dengan cantik dan indah atau bagus dan membaguskan, tajwid berasal dari kata Jawwada dalam bahasa Arab.

Dalam ilmu Qiraah, tajwid berarti mengeluarkan huruf dari tempatnya dengan memberikan sifat-sifat yang dimilikinya. Jadi ilmu tajwid adalah suatu ilmu yang mempelajari bagaimana cara membunyikan atau mengucapkan huruf-huruf yang terdapat dalam kitab suci al-Quran maupun bukan.

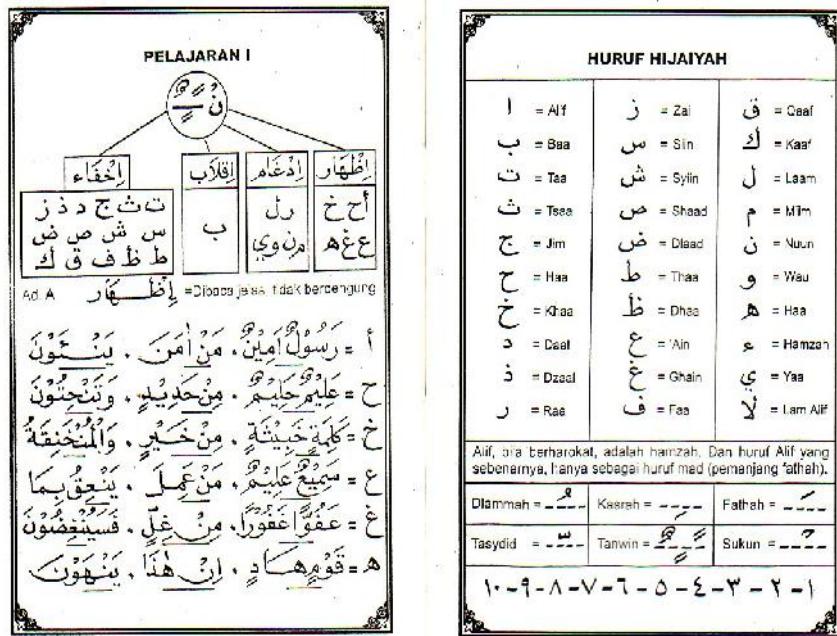
Adapun masalah-masalah yang dikemukakan dalam ilmu ini adalah makharijul huruf (tempat keluar-masuk huruf), shifatul huruf (cara pengucapan huruf), ahkamul huruf (hubungan antar huruf), ahkamul maddi wal qasr (panjang dan pendek ucapan), ahkamul waqaf wal ibtida' (memulai dan menghentikan bacaan) dan al-Khat al-Utsmani.

Pengertian lain dari ilmu tajwid ialah menyampaikan dengan sebaik-baiknya dan sempurna dari tiap-tiap bacaan ayat al-Quran. Para ulama menyatakan bahwa hukum bagi mempelajari tajwid itu adalah fardhu kifayah tetapi mengamalkan tajwid ketika membaca al-Quran adalah fardhu ain atau wajib kepada lelaki dan perempuan yang mukallaf atau dewasa.

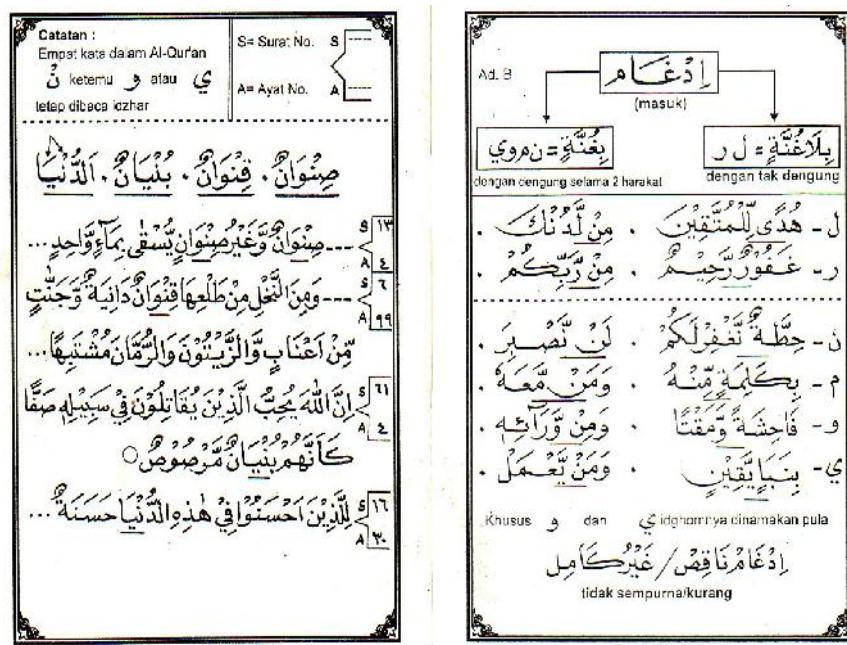
## 2.2 Pola Hukum Tajwid

Ada banyak ketentuan atau hukum di dalam ilmu tajwid, diantaranya dapat dilihat pada gambar 2.1 Huruf Hijaiyah, gambar 2.2 Pola Hukum Idh-har, Jika nun mati atau tanwin bertemu dengan salah satu huruf halqy yaitu , , , ^, gambar 2.3 (kanan) Hukum Iqlab, Iqlab adalah salah satu hukum tajwid yang berlaku apabila huruf Nun Sukun ( ) atau tanwin ( , , ) bertemu dengan huruf Ba ( ). Menurut bahasa, Iqlab artinya mengubah atau menggantikan sesuatu dari bentuknya. Cara membacanya adalah dengan menggantikan huruf atau , , menjadi suara huruf mim sukun ( ) sehingga pada saat akan bertemu dengan huruf bibir atas dan bawah dalam posisi tertutup, diiringi dengan suara dengung sekitar 2 harakat. Hukum Iqlab di dalam Al-Quran, sudah ditandai dengan huruf mim kecil ( ) – dan diletakkan di atas – antara atau , , dengan huruf , gambar 2.3 (kiri) hukum Ikhfa', Ikhfa' secara harfiah berarti menyamarkan atau menyembunyikan. Di dalam ilmu tajwid, Ikhfa' adalah menyamarkan huruf Nun Sukun ( ) atau tanwin ( , , ) ke dalam huruf sesudahnya – ada 15 huruf – yaitu: - - - - - - - - - - - - - - - - . Ke-15 huruf tersebut tidak bertasydid dan harus dibaca dengung (ghunnah). Cara membacanya adalah dengan mengeluarkan suara atau , , dari rongga hidung sehingga terlihat samar atau menjadi suara "N" atau "NG", kemudian disambut dengan dengung 1 – 1 1/2 Alif atau sekitar 2 – 3 harakat, setelah itu baru masuk ke huruf sesudahnya. Dan gambar 2.5 2.6, dan 2.7 Hukum Idgham, Idgham Bighunnah atau sering disebut Idgham Ma'al Ghunnah adalah hukum tajwid yang berlaku apabila Nun Sukun ( ) atau tanwin ( , , ) bertemu dengan huruf Mim, Nun, Waw, Ya ( ), secara terpisah atau tidak dalam satu kata/kalimat. Bi artinya dengan, Ghunnah artinya dengung. Sementara Idgham artinya meleburkan satu huruf ke dalam

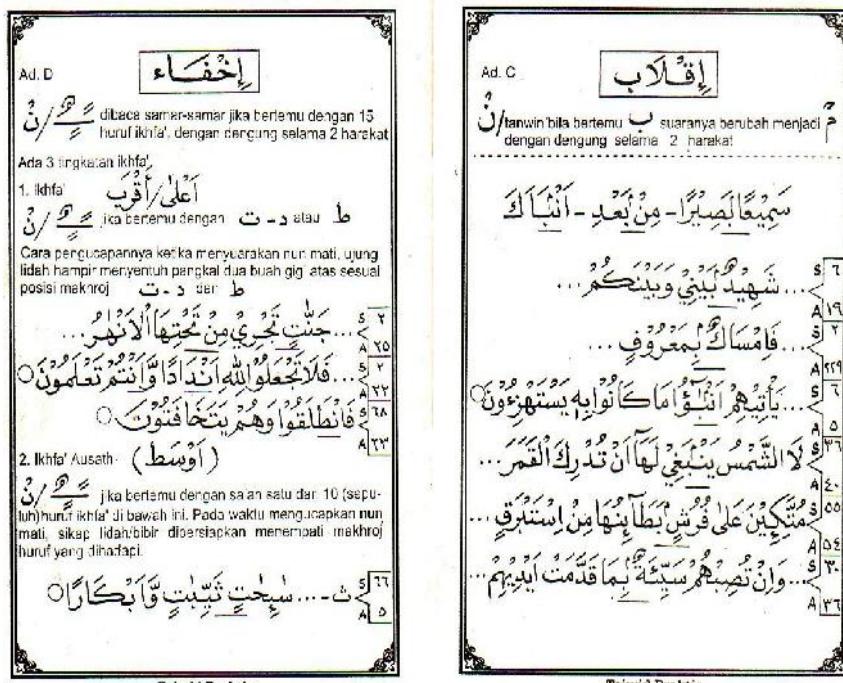
huruf setelahnya, atau bahasa lainnya di-tasyid-kan. Cara membaca Idgham Bighunnah adalah dengan meleburkan atau , , menjadi suara huruf di depannya , atau keempat huruf tersebut seolah diberi tanda tasyid, diiring dengan menggunakan suara dengung 1 Alif - 1 1/2 Alif atau sekitar 2 - 3 harakat. Hukum Idgham Bilaghunnah adalah hukum tajwid yang berlaku apabila Nun Sukun ( ) atau tanwin ( , , ) bertemu dengan huruf lam ( ) atau Ro ( ), tanpa menggunakan suara dengung. Bila artinya tidak, Ghunnah artinya dengung. Sementara Idgham artinya meleburkan satu huruf ke dalam huruf setelahnya, atau bahasa lainnya di-tasyid-kan. Cara membacanya adalah dengan meleburkan atau , , menjadi suara huruf atau , atau lafaz kedua huruf tersebut seolah diberi tanda tasyid, tanpa dikuti suara dengung (ghunnah). Dengan adanya perbedaan dengung ini, dapat dikatakan bahwa Idgham Bilaghunnah adalah kebalikan dari Idgham Bighunnah. (sumber gambar diambil dari buku Cara Cepat Belajar Tajwid Praktis yang ditulis oleh KH. As'ad Humam)



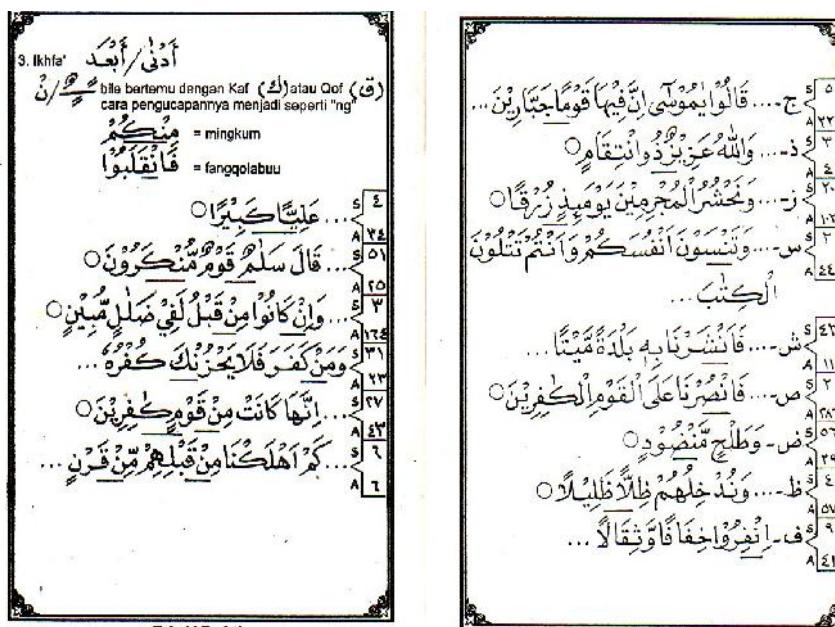
## Gambar 2.1 Huruf Hijaiyah



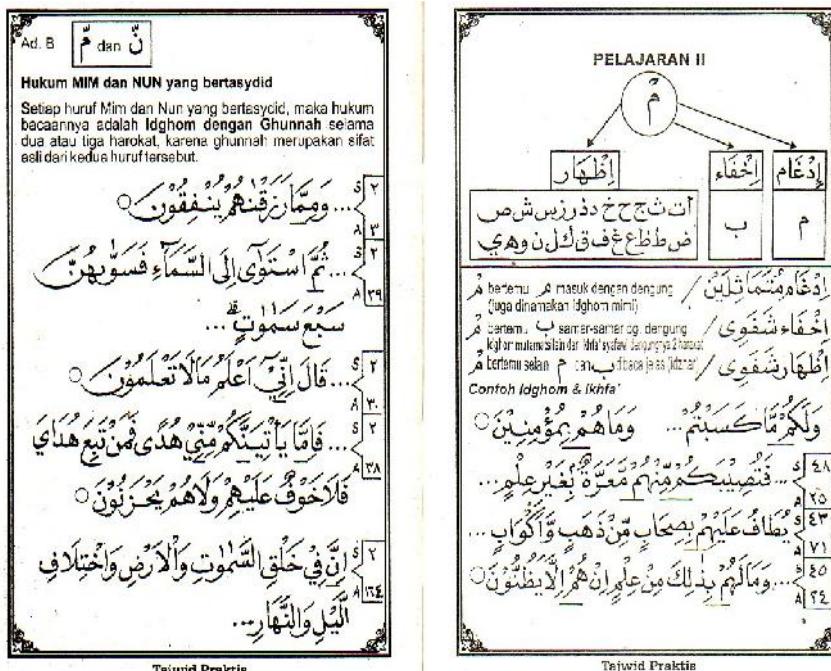
Gambar 2.2 Hukum Idh-har



Gambar 2.3 Hukum Iqlab dan Ikhfa'



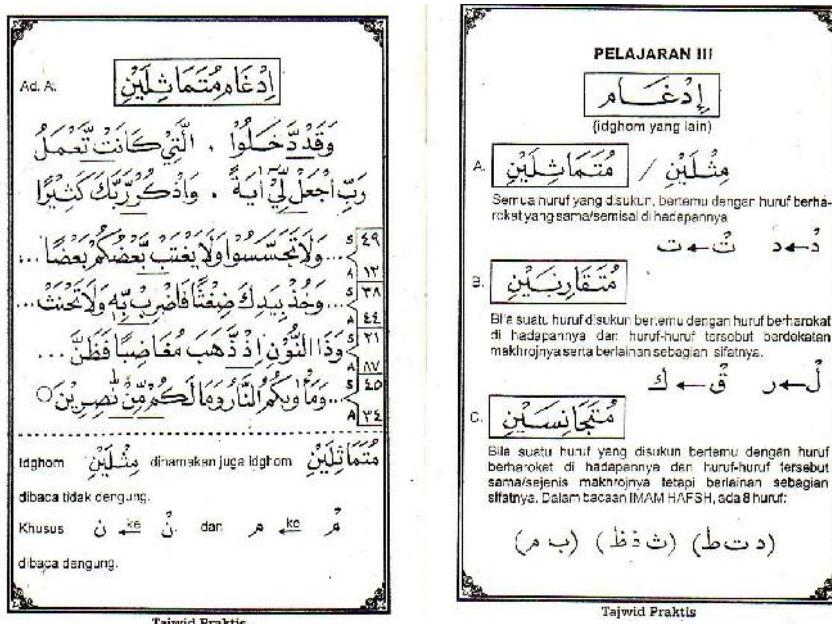
Gambar 2.4 Hukum Ikhfa'



Tajwid Praktis

Tajwid Praktis

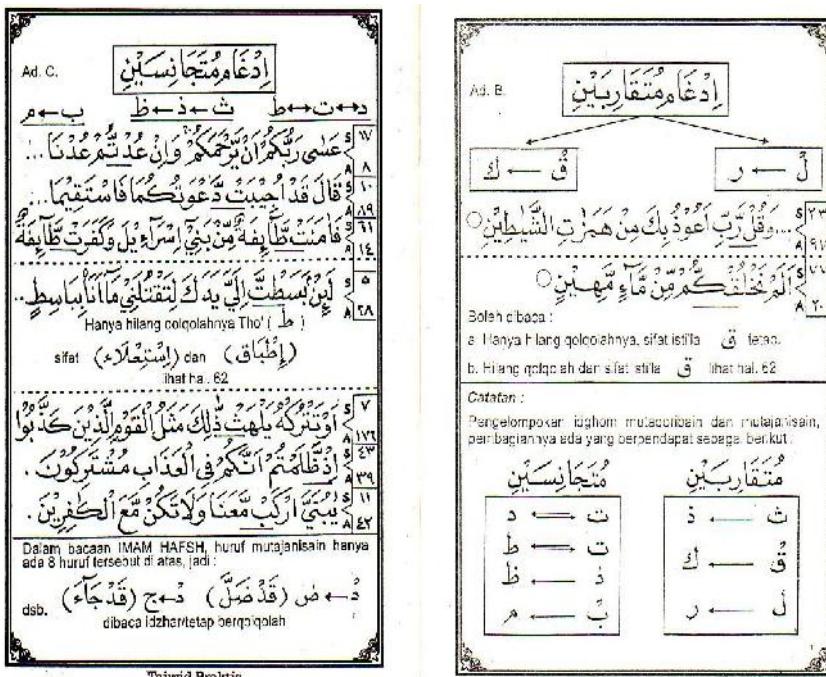
Gambar 2.5 Hukum Idgham



Tajwid Praktis

Tajwid Praktis

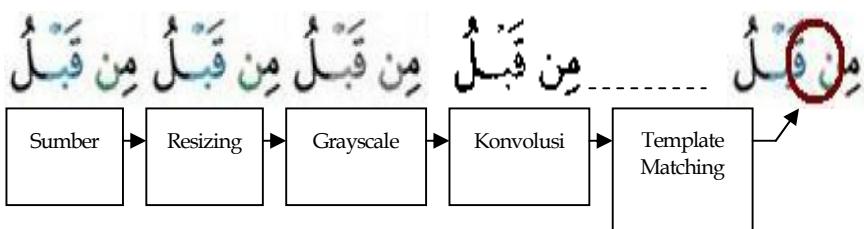
Gambar 2.6 Hukum Idgham



Gambar 2.7 Hukum Idgham

### 2.3 Skema Sistem

Adapun skema sistem pendekripsi pola tajwid untuk satu pola disajikan pada gambar 2.8.



Gambar 2.8 Skema sistem pendekripsi pola tajwid

Adapun tahapan sistem yang dibangun adalah citra sumber diresizing, setelah proses resizing, sistem melalukan tahapan grayscale,

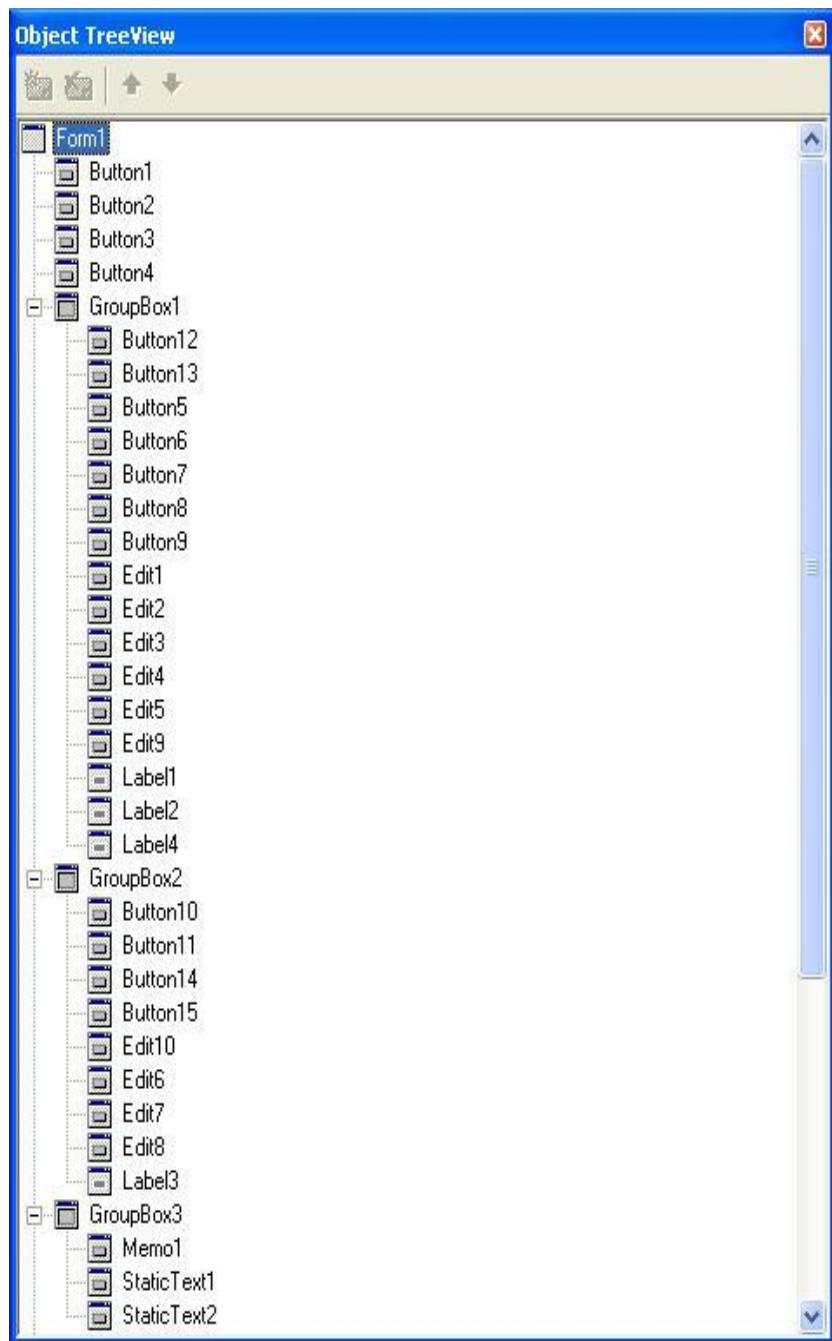
dan dilanjutkan ke proses konvolusi, dan setelah proses konvolusi, algoritma atau metode pendekatan yang digunakan untuk pendekripsi berperan dalam mengenali pola-pola yang dilatih.

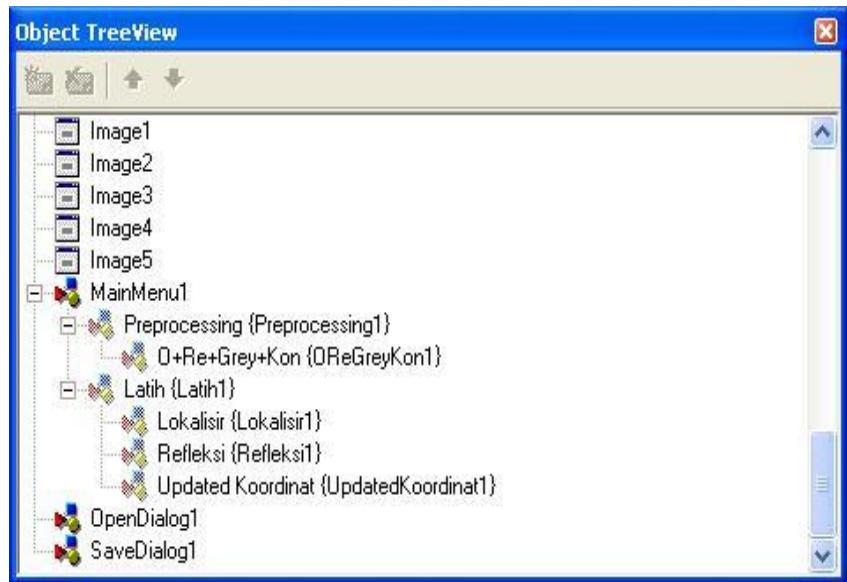
#### **2.4 Pembangunan Perangkat Lunak Deteksi Tajwid**

Untuk membangun sistem pendekripsi tajwid untuk satu pola pada citra, maka langkah-langkah kerja yang harus diikuti adalah sebagai berikut :

1. Jalankan Delphi, setelah muncul halaman Form1, maka tambahan komponen-komponen berikut :







2. Atur sedemikian rupa agar desain tampilan Form1 akan berupa tampilan berikut :



3. Klik 2x pada Button1, dan tuliskan kode program di bawah ini:

```
procedure TForm1.Button1Click(Sender: TObject);
//load picture
```

```

begin
  opendialog1.filter := 'bitmaps | *bmp' ;
  if      opendialog1.execute      then
  citral.loadfromfile(opendialog1.filename);
  citral.pixelformat := pf24bit; //in case
other format loaded

  Image1.Picture.Bitmap := citral;
  sw := citral.width;
  sh := citral.height;
  statictext1.caption := inttostr(sw);
  statictext2.caption := inttostr(sh);
end;

```

4. Klik 2x pada Button2, dan tuliskan kode program di bawah ini:

```

procedure      TForm1.Button2Click(Sender:
TObject);
//reduce or enlarge
var sx1,sy1,sx2,sy2 : single;
//source field positions
  x,y : word;
//dest field pixels
  destR,destG,destB : single;
//destination colors
  sR,sG,sB : byte;
//source colors
  destWidth, destheight : word;
  f,fi2 : single;
  i,j : word;
  dx,dy,PC : single;
  color : longInt;
begin
  destwidth := strtoint(edit1.text);
  f := sw / destwidth;
  fi2 := 1/f;
  fi2 := fi2*fi2;
  //destheight := trunc(citral.height/f);
  destheight := 600;
  with citra2 do
    begin
      width := destwidth;
      height := destheight;

```

```

        end;
//---
for y := 0 to destheight-1 do
//vertical destination pixels
begin
    sy1 := f * y;
    sy2 := sy1 + f;
    for x := 0 to destwidth-1 do
//horizontal destination pixels
begin
    sx1 := f * x;
    sx2 := sx1 + f;
    destR := 0; destG := 0; destB := 0;
//clear colors
    for j := floor(sy1) to ceiling(sy2)-1 do
//vertical source pixels
begin
    dy := 1;
    if sy1 > j then begin
        dy := dy-(sy1-j);
    end;
    if sy2 < j+1 then begin
        dy := dy-(j+1-sy2);
    end;
    for i := floor(sx1) to ceiling(sx2)-1
do //horizontal source pixels
begin
    dx := 1;
    if sx1 > i then begin
        dx := dx-(sx1-i);
    end;
    if sx2 < i+1 then begin
        dx := dx-(i+1-
sx2);
    end;
    color := citral.canvas.pixels[i,j];
    sR := color and $ff;
    sG := (color shr 8) and $ff;
    sB := (color shr 16) and $ff;
    PC := dx*dy*fi2;
    destR := destR + sR*PC;
    destG := destG + sG*PC;
    destB := destB + sB*PC;
end;//for i
end;//for j

```

```

        citra2.Canvas.Pixels[x,y]           := 
RGB(trunc(destR),trunc(destG),trunc(destB));
    end;//for x
end;//for y
Image2.Picture.Bitmap := citra2;
citra2.SaveToFile('resizinghasil.bmp');
end;

```

5. Klik 2x pada Button3, dan tuliskan kode program di bawah ini:

```

procedure TForm1.Button3Click(Sender:
TObject);
var temp  : PByteArray;
    i,j   : integer;
    x     : byte;
    digit : byte;
begin
    citra3 := TBitmap.Create;
    citra3.LoadFromFile('resizinghasil.bmp');

    for j:=0 to citra3.Height-1 do
    begin
        temp := citra3.ScanLine[j];
        i:=0;
        repeat
x := round((0.11*temp[i]) + (0.59*temp[i+1]) +
(0.3*temp[i+2]));
            for digit:=0 to 2 do
                temp[i+digit]:=x;
            inc(i,3);
            until i >= 3*citra3.Width-1;
    end;
    Image3.Picture.Bitmap := citra3;
    citra3.SaveToFile('greyscale.bmp');
    citra4:=citra3;
end;

```

6. Klik 2x pada Button4, dan tuliskan kode program di bawah ini:

```

procedure TForm1.Button4Click(Sender:
TObject);

```

```

const sobel : array[0..1,0..2,0..2] of
smallint =
(((-1,0,1),(-2,0,2),(-1,0,1)),
((-1,-2,-1),(0,0,0),(1,2,1)));
prewitt : array[0..1,0..2,0..2] of
smallint=
(((-1,0,1),(-1,0,1),(-1,0,1)),
((-1,-1,-1),(0,0,0),(1,1,1)));
var row : array[0..8] of
pbytearray;
col : pbytearray;
x,y : smallint;
i,j,k,p : smallint;
citra_tampung : tbitmap;
sum,jum : longint;
pos_x,pos_y : integer;

begin
P:=-100;
pos_y:=0;
citra_tampung := tbitmap.Create;
citra_tampung.Assign(citra3);
for y:=1 to citra4.Height-2 do
begin
for i:=-1 to 1 do
row[i+1]:= citra_tampung.ScanLine[y+i];
col := citra4.ScanLine[y];
x:=3;
inc(pos_y,1);
pos_x:=1;
repeat
sum := 0;
for i:=-1 to 1 do
for j:=-1 to 1 do

sum:=sum+(sobel[0,i+1,j+1]*row[i+1,x+j*3]);
jum:=0;
for i:=-1 to 1 do
for j:=-1 to 1 do

jum:=jum+(sobel[1,i+1,j+1]*row[i+1,x+j*3]);
sum := (sum + jum)+p;
if sum>255 then sum:=255;
if sum<0 then sum:=0;
for k:=0 to 2 do col[x+k]:=sum;

```

```

        t_edge[pos_y,pos_x]:=sum;
        if t_edge[pos_y,pos_x]<20 then
t_edge[pos_y,pos_x]:=0 else
            if t_edge[pos_y,pos_x]>=20 then
t_edge[pos_y,pos_x]:=1;
            if t_edge[pos_y,pos_x]<>0 then
memol.Lines.Add('['+
inttostr(pos_y)      + ','+inttostr(pos_x)+'
']=+' '+inttostr( t_edge[pos_y,pos_x]));
inc(x,3);
inc(pos_x,1);
until x>=3*(citra4.Width-4);
end;
Image4.Picture.bitmap := citra4;
citra4.SaveToFile('convolusi.bmp');
citra_tampung.free;
edit6.Text:=inttostr(pos_x);
edit7.Text:=inttostr(pos_y);
end;

```

7. Klik 2x pada Button5, Button6, Button7, Button8 dan berturut-turut tuliskan kode program di bawah ini:

```

procedure TForm1.Button5Click(Sender:
TObject);
begin
X_atas:=X_atas+2;
X_baw:=X_baw+2;
lokalisir1.click;
Updatedkoordinat1.Click;
end;
procedure TForm1.Button6Click(Sender:
TObject);
begin
X_atas:=X_atas-2;
X_baw:=X_baw-2;
lokalisir1.click;
Updatedkoordinat1.Click;
end;
procedure TForm1.Button7Click(Sender:
TObject);

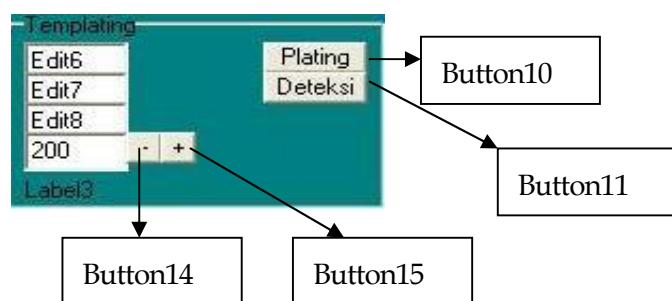
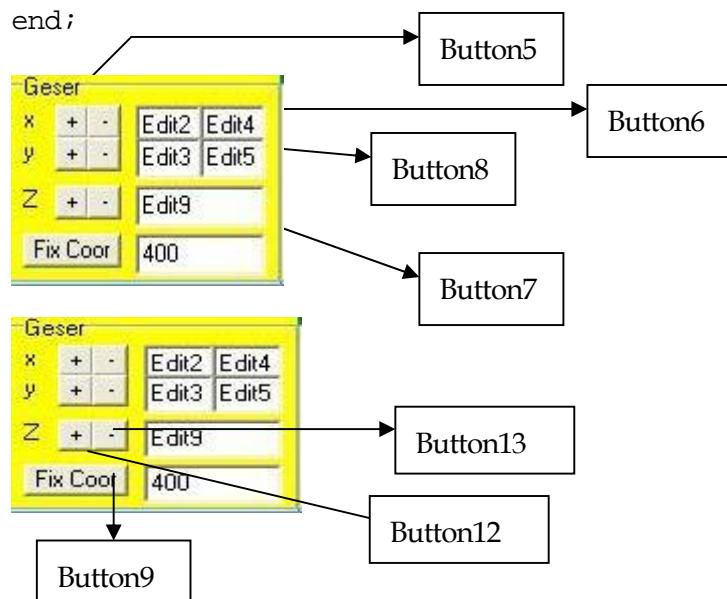
```

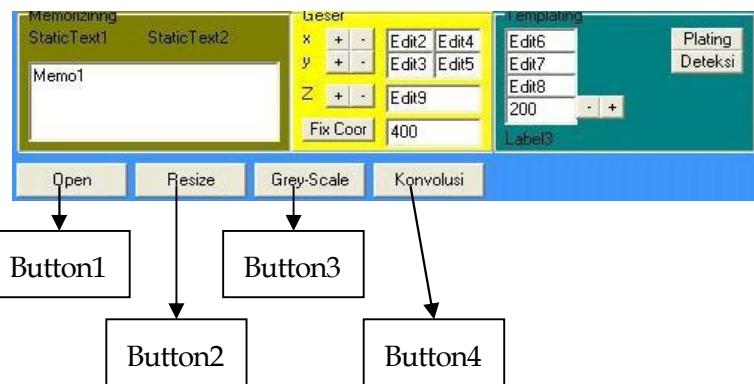
```

begin
Y_atas:=Y_atas+2;
Y_baw:=Y_baw+2;
lokalisir1.click;
Updatedkoordinat1.Click;
end;

procedure TForm1.Button8Click(Sender:
TObject);
begin
Y_atas:=Y_atas-2;
Y_baw:=Y_baw-2;
lokalisir1.click;
Updatedkoordinat1.Click;
end;

```





8. Klik 2x pada Button9, Button10, Button11, Button12, Button13, Button14, Button15 dan berturut-turut tuliskan kode program di bawah ini:

```

procedure TForm1.Button9Click(Sender: TObject);
begin
  Image2.Picture.Bitmap := citra2;
  lokalisir1.click;
  refleksil.click;
  Updatedkoordinat1.Click;
end;

procedure TForm1.Button10Click(Sender: TObject);
var temp : PByteArray;
    i,j : integer;
    x : byte;
    pos_x,pos_y : integer;

begin
  //plating
  pos_y:=0;

  //Arsip
  NamaArsip:='bit.txt';
  AssignFile(Fi,NamaArsip);
  rewrite(Fi);

  for j:=Y_atas to Y_baw-1 do
    begin

```

```

        i:=X_atas;
        inc(pos_y,1);
        pos_x:=1;
        repeat
            x:=t_edge[j,i];
        //templating adalah template base yg akan
        dimatch dgn t_edge d detek
            templating[pos_y,pos_x]:=x;

        write(Fi,inttostr(templating[pos_y,pos_x]));
            inc(i,1);
            inc(pos_x,1);
            until i >= X_baw-1;
            writeln(Fi);
        end;
        CloseFile(Fi);
        edit6.Text:=inttostr(pos_x);
        edit7.Text:=inttostr(pos_y);

        //-----
        -----
    end;

procedure TForm1.Button11Click(Sender:
TObject);
var pos_x,pos_y      : integer;
    cac_x,cac_y      : integer;
    ini_x,ini_y      : integer;
    cac_waj          : integer;

begin
    Image5.Picture.bitmap := citra2;

    for pos_y:=1 to citra2.height-40 do
        for pos_x:=1 to (citra2.width-40) do
        //if pos_y mod 2 = 0 then
        begin

            ini_y:=0;
            cac_waj:=0;
            for cac_y:=pos_y to (pos_y+Y_baw-Y_atas-
1) do
                begin

```

```

inc(ini_y,1);
ini_x:=0;
for cac_x:=pos_x to (pos_x+X_baw-X_atas-
1) do
begin
inc(ini_x,1);

//bagian vip pembanding pola

if t_edge[cac_y,
cac_x]=templatting[ini_y,ini_x] then
inc(cac_waj,1);

end; //end for cac_x
end; //end for cac_y
//total
if cac_waj>=(luasframe-sensitif) then
begin
label3.caption:=inttostr(cac_waj);
//maka catat koordinat awal dan ahir
frame
koor_x_atas_waj:=pos_x;
koor_y_atas_waj:=pos_y;
koor_x_baw_waj:=koor_x_atas_waj+ X_baw-
X_atas ;
koor_y_baw_waj:=pos_y+ Y_baw-Y_atas;

//frame

image5.Canvas.Brush.Style:=bsclear;
image5.Canvas.Pen.Width:=5;
image5.Canvas.Pen.Color:=clblue;
image5.Canvas.Ellipse(koor_x_atas_waj,
koor_y_atas_waj, koor_x_baw_waj,
koor_y_baw_waj);
cac_waj:=0;
end; // end if cac_waj
end; //end for awal
if cac_waj>1500 then
edit8.Text:=inttostr(cac_waj);

end;

procedure TForm1.Button12Click(Sender:
TObject);

```

```

begin
X_atas:=X_atas-1;
Y_atas:=Y_atas-1;
X_baw:=X_baw+1;
Y_baw:=Y_baw+1;
luasframe:=(Y_baw-Y_atas)*(X_baw-X_atas);
edit9.Text:=inttostr(luasframe);
end;

procedure TForm1.Button13Click(Sender:
TObject);
begin
X_atas:=X_atas+1;
Y_atas:=Y_atas+1;
X_baw:=X_baw-1;
Y_baw:=Y_baw-1;
luasframe:=(Y_baw-Y_atas)*(X_baw-X_atas);
edit9.Text:=inttostr(luasframe);
end;

procedure TForm1.Button14Click(Sender:
TObject);
begin
sensitif:=sensitif-1;
edit10.Text:=inttostr(sensitif);
end;

procedure TForm1.Button15Click(Sender:
TObject);
begin
sensitif:=sensitif+1;
edit10.Text:=inttostr(sensitif);
end;

```

9. Klik 2x pada sel OnActivate pada Form1, dan tuliskan :



```
procedure TForm1.FormActivate(Sender:  
TObject);  
begin  
X_atas:=34;  
Y_atas:=45;  
X_baw:=74;  
Y_baw:=85;  
//----- BOBOT  
NamaArsip:='koordinat.txt';  
AssignFile(Fi,NamaArsip);  
reset(Fi);  
//-----  
while not Eof(Fi) do  
begin  
Readln(Fi,tampungkoordinat);  
X_atas:=strToInt(tampungkoordinat);  
Readln(Fi,tampungkoordinat);  
Y_atas:=strToInt(tampungkoordinat);
```

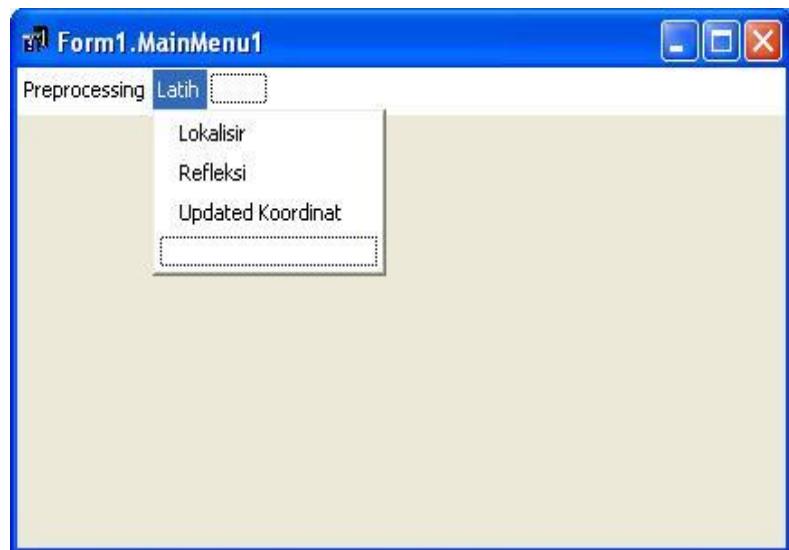
```
Readln(Fi,tampungkoordinat);
X_baw:=strtoint(tampungkoordinat);
Readln(Fi,tampungkoordinat);
Y_baw:=strtoint(tampungkoordinat);

end;

CloseFile(Fi);
luasframe:=(Y_baw-Y_atas)*(X_baw-X_atas);
edit9.Text:=inttostr(luasframe);
sensitif:=strtoint(edit10.Text);
end;
```

10. Klik 2x masing-masing item menu pada MainMenu1, dan tuliskan :





```
procedure TForm1.OReGreyKon1Click(Sender:  
 TObject);  
begin  
 button1.click;  
 button2.click;  
 button3.click;  
 button4.click;  
 Image5.Picture.bitmap := citra2;  
  
end;  
  
procedure TForm1.Lokalisir1Click(Sender:  
 TObject);  
begin  
 Image4.Picture.bitmap := citra4;  
 //-----melokalisir-----  
 image4.Canvas.Brush.Style:=bsClear;  
 image4.Canvas.Pen.Width:=3;  
 image4.Canvas.Pen.Color:=clRed;  
 image4.Canvas.Rectangle(X_atas, Y_atas,  
 X_baw, Y_baw);  
 //-----melokalisir-----  
end;  
procedure TForm1.Refleksi1Click(Sender:  
 TObject);
```

```

begin
//-----melokalisir-----
image2.Canvas.Brush.Style:=bsClear;
image2.Canvas.Pen.Width:=3;
image2.Canvas.Pen.Color:=clRed;
image2.Canvas.Rectangle(X_atas,           Y_atas,
X_baw, Y_baw);
//-----melokalisir-----
end;

procedure
TForm1.UpdatedKoordinat1Click(Sender:
 TObject);
begin
edit2.Text:=IntToStr(X_atas);
edit3.Text:=IntToStr(Y_atas);
edit4.Text:=IntToStr(X_baw);
edit5.Text:=IntToStr(Y_baw);
//Arsip
NamaArsip:='koordinat.txt';
AssignFile(Fi,NamaArsip);
rewrite(Fi);

writeln(Fi,IntToStr(X_atas));
writeln(Fi,IntToStr(Y_atas));
writeln(Fi,IntToStr(X_baw));
writeln(Fi,IntToStr(Y_baw));

CloseFile(Fi);
end;

```

## 2.5 Listing Keseluruhan

Setelah langkah-langkah di atas dikerjakan, maka langkah terakhir adalah memastikan keseluruhan kode program akan berupa listing kode berikut :

```

unit Unit1;
interface
uses

```

```
Windows, Messages, SysUtils, Variants, Classes,
Graphics, Controls, Forms,
Dialogs, ExtCtrls, StdCtrls, ExtDlgs, Menus,
jpeg;
type
TForm1 = class(TForm)
  Button1: TButton;
  Button2: TButton;
  OpenDialog1: TOpenDialog;
  SaveDialog1: TSaveDialog;
  Image1: TImage;
  Image2: TImage;
  Image3: TImage;
  Button3: TButton;
  Image4: TImage;
  Button4: TButton;
  MainMenu1: TMainMenu;
  Preprocessing1: TMenuItem;
  OReGreyKon1: TMenuItem;
  Latih1: TMenuItem;
  Lokalisir1: TMenuItem;
  GroupBox1: TGroupBox;
  Button5: TButton;
  Label1: TLabel;
  Button6: TButton;
  Label2: TLabel;
  Button7: TButton;
  Button8: TButton;
  Refleksil: TMenuItem;
  Edit2: TEdit;
  Edit3: TEdit;
  Edit4: TEdit;
  Edit5: TEdit;
  UpdatedKoordinat1: TMenuItem;
  Button9: TButton;
  GroupBox2: TGroupBox;
  Button10: TButton;
  Button11: TButton;
  Edit6: TEdit;
  Edit7: TEdit;
  Edit8: TEdit;
  Image5: TImage;
  Label3: TLabel;
  Label4: TLabel;
  Button12: TButton;
```

```

        Button13: TButton;
        Edit9: TEdit;
        Edit10: TEdit;
        Button14: TButton;
        Button15: TButton;
        Edit1: TEdit;
        GroupBox3: TGroupBox;
        Memo1: TMemo;
        StaticText1: TStaticText;
        StaticText2: TStaticText;
        procedure Button1Click(Sender: TObject);
        procedure Button2Click(Sender: TObject);
        procedure Button3Click(Sender: TObject);
        procedure Button4Click(Sender: TObject);
        procedure OReGreyKon1Click(Sender: TObject);
        procedure Lokalisir1Click(Sender: TObject);
        procedure FormActivate(Sender: TObject);
        procedure Button5Click(Sender: TObject);
        procedure Button6Click(Sender: TObject);
        procedure Button7Click(Sender: TObject);
        procedure Button8Click(Sender: TObject);
        procedure Refleksi1Click(Sender: TObject);
        procedure UpdatedKoordinat1Click(Sender:
TObject);
        procedure Button9Click(Sender: TObject);
        procedure Button10Click(Sender: TObject);
        procedure Button11Click(Sender: TObject);
        procedure Button12Click(Sender: TObject);
        procedure Button13Click(Sender: TObject);
        procedure Button14Click(Sender: TObject);
        procedure Button15Click(Sender: TObject);

private
  { Private declarations }
public
  { Public declarations }
end;
var
  Form1: TForm1;
  citra1,citra2, citra3, citra4 : TBitmap;
  citra5, citra6 : TBitmap;
  sw,sh : word;
  X_atas, Y_atas, X_baw, Y_baw : integer;
  templating, t_edge : array
[0..1000, 0..1000] of byte;

```

```

Fi                      : Textfile;
NamaArsip               : String;
koor_x_atas_waj         : integer;
koor_y_atas_waj         : integer;
koor_x_baw_waj          : integer;
koor_y_baw_waj          : integer;
luasframe               : integer;
tampungkoordinat        : string;
sensitif                : integer;

implementation

{$R *.dfm}

procedure TForm1.Button1Click(Sender: TObject);
//load picture
begin
  opendialog1.filter := 'bitmaps | *bmp' ;
  if           opendialog1.execute      then
  citral.loadfromfile(opendialog1.filename);
  citral.pixelformat := pf24bit; //in case other
format loaded
  Image1.Picture.Bitmap := citral;
  sw := citral.width;
  sh := citral.height;
  statictext1.caption := inttostr(sw);
  statictext2.caption := inttostr(sh);
end;

function floor(a : single) : word;
begin
  result := trunc(a);
end;

function ceiling(a : single) : word;
begin
  result := trunc(a);
  if frac(a) > 0.0001 then inc(result);
end;

procedure TForm1.Button2Click(Sender: TObject);
//reduce or enlarge
var sx1,sy1,sx2,sy2 : single;      //source field
positions

```

```

        x,y : word;                                //dest field
pixels
        destR,destG,destB : single;      //destination
colors
        sR,sG,sB : byte;                  //source colors
        destWidth, destheight : word;
        f,fi2 : single;
        i,j : word;
        dx,dy,PC : single;
        color : longInt;
begin
    destwidth := strtoint(edit1.text);
    f := sw / destwidth;
    fi2 := 1/f;
    fi2 := fi2*fi2;
    //destheight := trunc(citral.height/f);
    destheight := 600;
    with citra2 do
        begin
            width := destwidth;
            height := destheight;
        end;
    //---
    for y := 0 to destheight-1 do           //vertical
destination pixels
        begin
            syl := f * y;
            sy2 := syl + f;
            for x := 0 to destwidth-1 do
//horizontal destination pixels
                begin
                    sx1 := f * x;
                    sx2 := sx1 + f;
                    destR := 0; destG := 0; destB := 0;
//clear colors
                    for j := floor(syl) to ceiling(sy2)-1 do
//vertical source pixels
                        begin
                            dy := 1;
                            if syl > j then begin
                                dy := dy-(syl-j);
                            end;
                            if sy2 < j+1 then begin
                                dy := dy-(j+1-sy2);
                            end;

```

```

        for i := floor(sx1) to ceiling(sx2)-1 do
//horizontal source pixels
begin
  dx := 1;
  if sx1 > i then begin
    dx := dx-(sx1-i);
  end;
  if sx2 < i+1 then begin
    dx := dx-(i+1-sx2);
  end;
  color := citra1.canvas.pixels[i,j];
  SR := color and $ff;
  SG := (color shr 8) and $ff;
  SB := (color shr 16) and $ff;
  PC := dx*dy*f12;
  destR := destR + SR*PC;
  destG := destG + SG*PC;
  destB := destB + SB*PC;
end;//for i
end;//for j
citra2.Canvas.pixels[x,y]           :=
RGB(trunc(destR),trunc(destG),trunc(destB));
end;//for x
end;//for y

Image2.Picture.Bitmap := citra2;
citra2.SaveToFile('resizinghasil.bmp');
end;

procedure TForm1.Button3Click(Sender: TObject);
var temp : PByteArray;
  i,j   : integer;
  x     : byte;
  digit : byte;
begin
  citra3 := TBitmap.Create;
  citra3.LoadFromFile('resizinghasil.bmp');

  for j:=0 to citra3.Height-1 do
  begin
    temp := citra3.ScanLine[j];
    i:=0;
    repeat

```

```

        x
:=round((0.11*temp[i])+(0.59*temp[i+1])+(0.3*temp[
i+2]));
        for digit:=0 to 2 do
            temp[i+digit]:=x;
            inc(i,3);
        until i >= 3*citra3.Width-1;
    end;
Image3.Picture.Bitmap := citra3;
citra3.SaveToFile('greyscale.bmp');
citra4:=citra3;
end;

procedure TForm1.Button4Click(Sender: TObject);
const sobel : array[0..1,0..2,0..2] of smallint =
      (((-1,0,1),(-2,0,2),(-1,0,1)),
       ((-1,-2,-1),(0,0,0),(1,2,1)));
      prewitt : array[0..1,0..2,0..2] of smallint
      =
      (((-1,0,1),(-1,0,1),(-1,0,1)),
       ((-1,-1,-1),(0,0,0),(1,1,1)));
var row                  : array[0..8] of pbytearray;
    col                  : pbytearray;
    x,y                  : smallint;
    i,j,k,p              : smallint;
    citra_tampung         : tbitmap;
    sum,jum               : longint;
    pos_x,pos_y           : integer;
begin
  P:=-100;
  pos_y:=0;
  citra_tampung := tbitmap.Create;
  citra_tampung.Assign(citra3);
  for y:=1 to citra4.Height-2 do
    begin
      for i:=-1 to 1 do
        row[i+1]:= citra_tampung.ScanLine[y+i];
        col := citra4.ScanLine[y];
        x:=3;
        inc(pos_y,1);
        pos_x:=1;
      repeat
        sum := 0;
        for i:=-1 to 1 do
          for j:=-1 to 1 do

```

```

sum:=sum+(sobel[0,i+1,j+1]*row[i+1,x+j*3]);
    jum:=0;
    for i:=-1 to 1 do
        for j:=-1 to 1 do

jum:=jum+(sobel[1,i+1,j+1]*row[i+1,x+j*3]);
    sum := (sum + jum)+p;
    if sum>255 then sum:=255;
    if sum<0 then sum:=0;
    for k:=0 to 2 do col[x+k]:=sum;
    t_edge[pos_y,pos_x]:=sum;
    if      t_edge[pos_y,pos_x]<20      then
t_edge[pos_y,pos_x]:=0 else
    if      t_edge[pos_y,pos_x]>=20      then
t_edge[pos_y,pos_x]:=1;
    if t_edge[pos_y,pos_x]<>0 then
        memo1.Lines.Add('['+ inttostr(pos_y) +
', '+inttostr(pos_x)+      ']'+'='+'      '+inttostr(
t_edge[pos_y,pos_x]));
        inc(x,3);
        inc(pos_x,1);
    until x>=3*(citra4.Width-4);
end;
Image4.Picture.bitmap := citra4;
citra4.SaveToFile('convolusi.bmp');
citra_tampung.free;
edit6.Text:=inttostr(pos_x);
edit7.Text:=inttostr(pos_y);
end;
procedure           TForm1.OReGreyKon1Click(Sender:
TObject);
begin
button1.click;
button2.click;
button3.click;
button4.click;
Image5.Picture.bitmap := citra2;
end;
procedure TForm1.Lokalisir1Click(Sender: TObject);
begin
Image4.Picture.bitmap := citra4;
//-----melokalisir-----
image4.Canvas.Brush.Style:=bsClear;
image4.Canvas.Pen.Width:=3;
image4.Canvas.Pen.Color:=clRed;

```

```

image4.Canvas.Rectangle(X_atas,      Y_atas,      X_baw,
Y_baw);
//-----melokalisir-----
end;
procedure TForm1.FormActivate(Sender: TObject);
begin
X_atas:=34;
Y_atas:=45;
X_baw:=74;
Y_baw:=85;
//----- BOBOT
NamaArsip:='koordinat.txt';
AssignFile(Fi,NamaArsip);
reset(Fi);
//-----
while not Eof(Fi) do
begin
  Readln(Fi,tampungkoordinat);
  X_atas:=strtoint(tampungkoordinat);
  Readln(Fi,tampungkoordinat);
  Y_atas:=strtoint(tampungkoordinat);
  Readln(Fi,tampungkoordinat);
  X_baw:=strtoint(tampungkoordinat);
  Readln(Fi,tampungkoordinat);
  Y_baw:=strtoint(tampungkoordinat);

end;

CloseFile(Fi);
luasframe:=(Y_baw-Y_atas)*(X_baw-X_atas);
edit9.Text:=inttostr(luasframe);
sensitif:=strtoint(edit10.Text);
end;

procedure TForm1.Button5Click(Sender: TObject);
begin
X_atas:=X_atas+2;
X_baw:=X_baw+2;
lokalisir1.click;
Updatedkoordinat1.Click;
end;
procedure TForm1.Button6Click(Sender: TObject);
begin
X_atas:=X_atas-2;
X_baw:=X_baw-2;

```

```

lokalisir1.click;
Updatedkoordinat1.Click;
end;

procedure TForm1.Button7Click(Sender: TObject);
begin
Y_atas:=Y_atas+2;
Y_baw:=Y_baw+2;
lokalisir1.click;
Updatedkoordinat1.Click;
end;

procedure TForm1.Button8Click(Sender: TObject);
begin
Y_atas:=Y_atas-2;
Y_baw:=Y_baw-2;
lokalisir1.click;
Updatedkoordinat1.Click;
end;

procedure TForm1.RefleksilClick(Sender: TObject);
begin
//-----melokalisir-----
image2.Canvas.Brush.Style:=bsClear;
image2.Canvas.Pen.Width:=3;
image2.Canvas.Pen.Color:=clRed;
image2.Canvas.Rectangle(X_atas, Y_atas, X_baw,
Y_baw);
//-----melokalisir-----
end;

procedure TForm1.UpdatedKoordinat1Click(Sender:
TObject);
begin
edit2.Text:=IntToStr(X_atas);
edit3.Text:=IntToStr(Y_atas);
edit4.Text:=IntToStr(X_baw);
edit5.Text:=IntToStr(Y_baw);
//Arsip
NamaArsip:='koordinat.txt';
AssignFile(Fi,NamaArsip);
Rewrite(Fi);

writeln(Fi,IntToStr(X_atas));
writeln(Fi,IntToStr(Y_atas));
writeln(Fi,IntToStr(X_baw));

```

```

        writeln(Fi,inttostr(Y_baw));

        CloseFile(Fi);
end;

procedure TForm1.Button9Click(Sender: TObject);
begin
Image2.Picture.Bitmap := citra2;
lokalisirl.click;
refleksil.click;
Updatedkoordinat1.Click;
end;

procedure TForm1.Button10Click(Sender: TObject);
var temp           : PByteArray;
    i,j           : integer;
    x             : byte;
    pos_x,pos_y  : integer;
begin
//plating
pos_y:=0;

//Arsip
NamaArsip:='bit.txt';
AssignFile(Fi,NamaArsip);
rewrite(Fi);

for j:=Y_atas to Y_baw-1 do
begin

i:=X_atas;
inc(pos_y,1);
pos_x:=1;
repeat
x:=t_edge[j,i];
//templating adalah template base yg
akan dimatch dgn t_edge d detek
templating[pos_y,pos_x]:=x;

writeln(Fi,inttostr(templating[pos_y,pos_x]));
inc(i,1);
inc(pos_x,1);
until i >= X_baw-1;
writeln(Fi);

```

```

end;
CloseFile(Fi);
edit6.Text:=inttostr(pos_x);
edit7.Text:=inttostr(pos_y);

//-----
-----

end;

procedure TForm1.Button11Click(Sender: TObject);
var pos_x,pos_y      : integer;
    cac_x,cac_y      : integer;
    ini_x,ini_y      : integer;
    cac_waj          : integer;

begin
  Image5.Picture.bitmap := citra2;

for pos_y:=1 to citra2.height-40 do
  for pos_x:=1 to (citra2.width-40) do
  //if pos_y mod 2 = 0 then
begin

  ini_y:=0;
  cac_waj:=0;
  for cac_y:=pos_y to (pos_y+Y_baw-Y_atas-1) do
  begin
  inc(ini_y,1);
  ini_x:=0;
  for cac_x:=pos_x to (pos_x+X_baw-X_atas-1) do
  begin
  inc(ini_x,1);
  //bagian vip pembanding pola

  if                                     t_edge[cac_y,
cac_x]=templatting[ini_y,ini_x]           then
inc(cac_waj,1);

  end; //end for cac_x
  end; //end for cac_y
  //total
  if cac_waj>=(luasframe-sensitif) then
  begin
  label3.caption:=inttostr(cac_waj);

```

```

        //maka catat koordinat awal dan ahir frame
        koor_x_atas_waj:=pos_x;
        koor_y_atas_waj:=pos_y;
        koor_x_baw_waj:=koor_x_atas_waj+    X_baw-X_atas
        ;
        koor_y_baw_waj:=pos_y+ Y_baw-Y_atas;

        //frame

        image5.Canvas.Brush.Style:=bsclear;
        image5.Canvas.Pen.Width:=5;
        image5.Canvas.Pen.Color:=clblue;
        image5.Canvas.Ellipse(koor_x_atas_waj,
        koor_y_atas_waj, koor_x_baw_waj, koor_y_baw_waj);
        cac_waj:=0;
        end; // end      if cac_waj
        end; //end for awal
        if           cac_waj>1500           then
        edit8.Text:=inttostr(cac_waj);
        end;

procedure TForm1.Button12Click(Sender: TObject);
begin
X_atas:=X_atas-1;
Y_atas:=Y_atas-1;
X_baw:=X_baw+1;
Y_baw:=Y_baw+1;
luasframe:=(Y_baw-Y_atas)*(X_baw-X_atas);
edit9.Text:=inttostr(luasframe);
end;

procedure TForm1.Button13Click(Sender: TObject);
begin
X_atas:=X_atas+1;
Y_atas:=Y_atas+1;
X_baw:=X_baw-1;
Y_baw:=Y_baw-1;
luasframe:=(Y_baw-Y_atas)*(X_baw-X_atas);
edit9.Text:=inttostr(luasframe);
end;

procedure TForm1.Button14Click(Sender: TObject);
begin
sensitif:=sensitif-1;
edit10.Text:=inttostr(sensitif);

```

```
end;

procedure TForm1.Button15Click(Sender: TObject);
begin
sensitif:=sensitif+1;
edit10.Text:=inttostr(sensitif);
end;

initialization

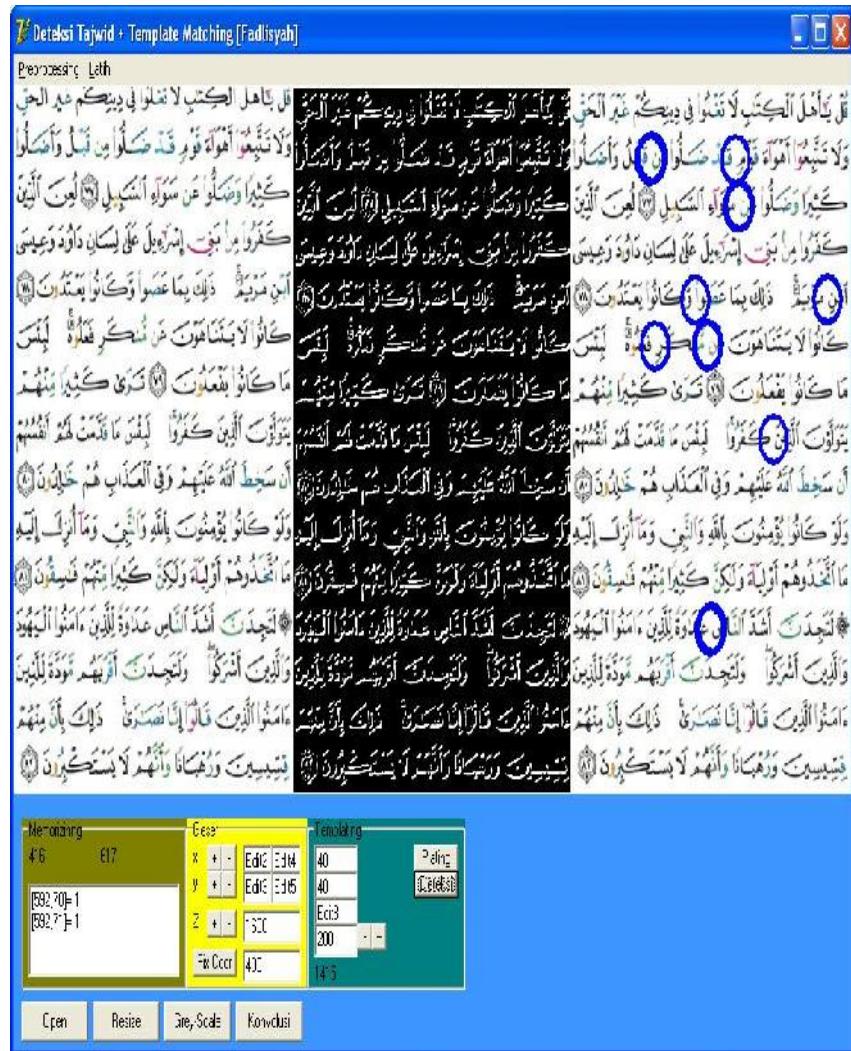
citra1 := Tbitmap.create;
with citra1 do
begin
  pixelformat := pf24bit;
  width := 100; height := 100;
end;
citra2 := TBitmap.create;
with citra2 do
begin
  pixelformat := pf24bit;
  width := 100; height := 100;
end;
sw := 100; sh := 100;

finalization

citra1.free;
citra2.free;

end.
```

dan tampilan hasil eksekusinya adalah :



# BAB 3

## ALGORITMA SIMILARITY BINER DAN DISTANCE UNTUK PENGENALAN POLA

### 3.1 Pendahuluan

Algoritma similarity biner dan algoritma pengukuran distance telah memainkan peranan penting dalam pengenalan pola dan penganalisaan pola seperti klasifikasi, identifikasi, klustering, pengenalan, deteksi, dan lain-lain.

Algoritma similarity biner dan algoritma pengukuran distance biasanya didefinisikan melalui tabel *Operational Taxonomic Units* (OTUs). Asumsikan terdapat dua vektor pola i dan j, maka terdapat suatu variabel a dengan ketentuan nilai vektor pola i dan nilai vektor j pada posisi yang sama yang bernilai [1,1], dimana vektor pola i dan j memiliki n atribut atau dimensi yang sama. Kondisi kecocokan a diistilahkan juga sebagai kondisi kecocokan positif. Untuk nilai b merupakan jumlah dari relasi vektor pola i dan vektor pola j yang memenuhi kondisi [0,1], c memenuhi kondisi [1,0], dan d adalah [0,0]. Kondisi kecocokan d diistilahkan juga sebagai kondisi kecocokan negatif. Tabel 3.1 mengilustrasikan hubungan vektor pola i dan vektor pola j dalam bentuk presentasi *Operational Taxonomic Units* (OTUs).

Tabel 3.1 *Operational Taxonomic Units* (OTUs)

$j \backslash i$	1 (Presence)	0 (Absence)	Sum
1 (Presence)	$a = i \bullet j$	$b = i \bullet \bar{j}$	$a+b$
0 (Absence)	$c = \bar{i} \bullet j$	$d = \bar{i} \bullet \bar{j}$	$c+d$
Sum	$a+c$	$b+d$	$n=a+b+c+d$

### 3.2 Daftar Algoritma Similarity Biner dan Pengukuran Distance

Algoritma yang berkenaan dengan similarity biner dan algoritma pengukuran distance yang akan dibahas di dalam buku ini berjumlah 76 formula, keseluruhan formula disajikan pada tabel 3.2.

Tabel 3.2 Daftar Algoritma

No	Algoritma	Bentuk Persamaan
1	Jaccard	$S = \frac{a}{a + b + c}$
2	Dice	$S = \frac{2a}{2a + b + c}$
3	Czekanowski	$S = \frac{2a}{2a + b + c}$
4	3W-Jaccard	$S = \frac{3a}{3a + b + c}$
5	Nei & Li	$S = \frac{2a}{(a + b) + (a + c)}$

6	Sokal Sneath I	$S = \frac{2a}{(a + 2b + 2c)}$
7	Sokal Michener	$S = \frac{a + d}{(a + b + c + d)}$
8	Sokal Sneath II	$S = \frac{2(a + d)}{(2a + b + c + 2d)}$
9	Roger Tanimoto	$S = \frac{a + d}{(a + 2(b + c) + d)}$
10	Faith	$S = \frac{a + 0.5d}{(a + b + c + d)}$
11	Gower Legendre	$S = \frac{a + d}{(a + 0.5(b + c) + d)}$
12	Intersection	$S = a$
13	Innerproduct	$S = a + d$
14	Russel Rao	$S = \frac{a}{(a + b + c + d)}$
15	Hamming	$D = b + c$
16	Euclid	$D = \sqrt{b + c}$
17	Squared Euclid	$D = \sqrt{(b + c)^2}$
18	Canberra	$D = (b + c)^{\frac{2}{3}}$
19	Manhattan	$D = b + c$
20	Mean Manhattan	$D = \frac{b + c}{(a + b + c + d)}$
21	Cityblock	$D = b + c$
22	Minkowski	$D = (b + c)^{\frac{1}{k}}$

23	Vari	$D = \frac{b+c}{4(a+b+c+d)}$
24	Size difference	$D = \frac{(b+c)^2}{(a+b+c+d)^2}$
25	Shape difference	$D = \frac{n(b+c) - (b-c)^2}{(a+b+c+d)^2}$
26	Pattern difference	$D = \frac{4bc}{(a+b+c+d)^2}$
27	Lance Williams	$D = \frac{b+c}{(2a+b+c)}$
28	Bray & Curtis	$D = \frac{b+c}{(2a+b+c)}$
29	Hellinger	$D = 2\sqrt{1 - \frac{a}{\sqrt{(a+b)(a+c)}}}$
30	Chord	$D = \sqrt{2\left(1 - \frac{a}{\sqrt{(a+b)(a+c)}}\right)}$
31	Cosine	$S = \frac{a}{\sqrt{(a+b)(a+c)}^2}$
32	Gilbert & Wells	$S = \log a - \log n - \log\left(\frac{a+b}{n}\right) - \log\left(\frac{a+c}{n}\right)$
33	Ochiai I	$S = \frac{a}{\sqrt{(a+b)(a+c)}}$
34	Forbes I	$S = \frac{na}{(a+b)(a+c)}$
35	Fossum	$S = \frac{n(a-0.5)^2}{(a+b)(a+c)}$

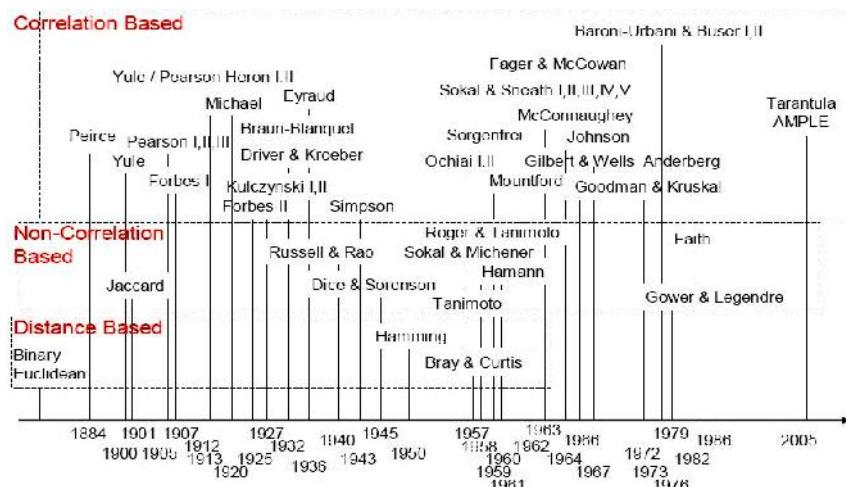
36	Sorgenfrei	$S = \frac{a^2}{(a+b)(a+c)}$
37	Mountford	$S = \frac{a}{0.5(ab+ac)+(bc)}$
38	Otsuka	$S = \frac{a}{((a+b)(a+c))^2}$
39	Mc Connaughey	$S = \frac{a^2 - bc}{(a+b)(a+c)}$
40	Tarwid	$S = \frac{na - (a+b)(a+c)}{na + (a+b)(a+c)}$
41	Kulczynski II	$S = \frac{\frac{a}{2}(2a+b+c)}{(a+b)(a+c)}$
42	Driver & Kroeber	$S = \frac{a}{2} \left( \frac{1}{a+b} + \frac{1}{a+c} \right)$
43	Johnson	$S = \left( \frac{a}{a+b} + \frac{a}{a+c} \right)$
44	Dennis	$S = \frac{ad - bc}{\sqrt{n(a+b)(a+c)}}$
45	Simpson	$S = \frac{a}{\min(a+b, a+c)}$
46	Braun & Banquet	$S = \frac{a}{\max(a+b, a+c)}$
47	Fager & McGowan	$S = \frac{a}{\sqrt{(a+b)(a+c)}} - \frac{\max(a+b, a+c)}{2}$
48	Forbes II	$S = \frac{na - (a+b)(a+c)}{n \min(a+b, a+c) - (a+b)(a+c)}$

49	Sokal & Sneath IV	$S = \frac{\frac{a}{(a+b)} + \frac{a}{(a+c)} + \frac{d}{(b+d)} + \frac{d}{(c+d)}}{4}$
50	Gower	$S = \frac{a+d}{\sqrt{(a+b)(a+c)(b+d)(c+d)}}$
51	Pearson I	$t^2 = \frac{n(ad - bc)^2}{(a+b)(a+c)(b+d)(c+d)}$
52	Pearson II	$S = \left( \frac{t^2}{n+t^2} \right)^{1/2}$
53	Pearson III	$S = \left( \frac{\dots}{n+\dots} \right)^{1/2}$ $\dots = \frac{(ad - bc)}{\sqrt{(a+b)(a+c)(b+d)(c+d)}}$
54	Pearson & Heron I	$S = \frac{(ad - bc)}{\sqrt{(a+b)(a+c)(b+d)(c+d)}}$
55	Pearson & Heron II	$S = \cos \left( \frac{f \sqrt{bc}}{\sqrt{ad} + \sqrt{bc}} \right)$
56	Sokal & Sneath III	$S = \frac{a+d}{b+c}$
57	Sokal & Sneath V	$S = \frac{ad}{(a+b)(a+c)(b+d)(c+d)^{0.5}}$
58	Cole	$S = \frac{\sqrt{2}(ad - bc)}{\sqrt{(ad - bc)^2 - (a+b)(a+c)(b+d)(c+d)}}$

59	Stiles	$S = \log_{10} \frac{n \left(  ad - bc  - \frac{n}{2} \right)^2}{(a+b)(a+c)(b+d)(c+d)}$
60	Ochiai II	$S = \frac{ad}{\sqrt{(a+b)(a+c)(b+d)(c+d)}}$
61	Yuleq	$D = \frac{2bc}{ad + bc}$
62	Yuleq	$S = \frac{ad - bc}{ad + bc}$
63	Yulew	$S = \frac{\sqrt{ad} - \sqrt{bc}}{\sqrt{ad} + \sqrt{bc}}$
64	Kulczynski I	$S = \frac{a}{b+c}$
65	Tanimoto	$S = \frac{a}{(a+b)+(a+c)-a}$
66	Dipersion	$S = \frac{ad - bc}{(a+b+c+d)^2}$
67	Hamann	$S = \frac{(a+d)-(b+c)}{a+b+c+d}$
68	Michael	$S = \frac{4(ad - bc)}{(a+d)^2 + (b+c)^2}$
69	Goodman & Kruskal	$S = \frac{\dagger - \dagger'}{2n - \dagger'}$ $\dagger = \max(a, b) + \max(c, d) + \max(a, c) + \max(b, d)$ $\dagger' = \max(a+c, b+d) + \max(a+b, c+d)$
70	Anderberg	$S = \frac{\dagger - \dagger'}{2n}$

71	Baroni Urbani & Buser I	$S = \frac{\sqrt{ad} + a}{\sqrt{ad} + a + b + c}$
72	Baroni Urbani & Buser II	$S = \frac{\sqrt{ad} + a - (b + c)}{\sqrt{ad} + a + b + c}$
73	Pierce	$S = \frac{ab + bc}{ab + 2bc + cd}$
74	Eyraud	$S = \frac{n^2(na - (a + b)(a + c))}{(a + b)(a + c)(b + d)(c + d)}$
75	Tarantula	$S = \frac{\frac{a}{(a+b)}}{\frac{c}{(c+d)}} = \frac{a(c+d)}{c(a+b)}$
76	Ample	$S = \left  \frac{\frac{a}{(a+b)}}{\frac{c}{(c+d)}} \right  = \left  \frac{a(c+d)}{c(a+b)} \right $

Kronologis algoritma pengukuran similarity biner dan pengukuran jarak (distance) berdasarkan tahun ditemukannya disajikan pada gambar 3.1.



Gambar 3.1 Tabel kronologis algoritma pengukuran similarity biner dan pengukuran jarak (distance) berdasarkan urutan tahun

### 3.3 Deteksi Tajwid menggunakan Algoritma Similarity Biner Kulczynski

Untuk menerapkan algoritma Kulczynski pada program, maka terlebih dahulu harus membangun algoritma *Operational Taxonomic Units* (OTUs). Implementasi algoritma *Operational Taxonomic Units* (OTUs) yang dibangun adalah :

```

if           (t_edge[cac_y,                               cac_x]=1)
AND(templating1[ini_y,ini_x]=1)                      then
a[1]:=a[1]+1;

if           (t_edge[cac_y,                               cac_x]=0)
AND(templating1[ini_y,ini_x]=1)                      then
b[1]:=b[1]+1;
  
```

```

if           (t_edge[cac_y,                  cac_x]=1)
AND(templating1[ini_y,ini_x]=0)          then
c[1]:=c[1]+1;

if           (t_edge[cac_y,                  cac_x]=0)
AND(templating1[ini_y,ini_x]=0)          then
d[1]:=d[1]+1;

```

kode program untuk algoritma Kulczynski adalah :

```

//total  dan hitung KULCZYNSKI
kulczynski_1:=  a[1]/(b[1]+c[1]);
kulczynski_2:=  a[2]/(b[2]+c[2]);

if cac_1>=(luasframe-sensitif) then
if (kulczynski_1>0) then
begin
//label3.caption:=inttostr(cac_1);
//maka catat koordinat awal dan akhir frame
koor_x_atas_waj:=pos_x;
koor_y_atas_waj:=pos_y;
koor_x_baw_waj:=koor_x_atas_waj+      X_baw-
X_atas ;
koor_y_baw_waj:=pos_y+ Y_baw-Y_atas;

//frame

image5.Canvas.Brush.Style:=bsclear;
image5.Canvas.Pen.Width:=5;
image5.Canvas.Pen.Color:=clblue;
image5.Canvas.Ellipse(koor_x_atas_waj,
koor_y_atas_waj,                      koor_x_baw_waj,
koor_y_baw_waj);
cac_1:=0;
end; // end    if cac_1

if cac_2>=(luasframe-sensitif) then
if (kulczynski_2>0) then
begin
//label5.caption:=inttostr(cac_2);

```

```

//maka catat koordinat awal dan ahir frame
koor_x_atas_waj:=pos_x;
koor_y_atas_waj:=pos_y;
koor_x_baw_waj:=koor_x_atas_waj+           X_baw-
X_atas ;
koor_y_baw_waj:=pos_y+ Y_baw-Y_atas;

//frame

image5.Canvas.Brush.Style:=bsclear;
image5.Canvas.Pen.Width:=5;
image5.Canvas.Pen.Color:=clgreen;
image5.Canvas.Ellipse(koor_x_atas_waj,
koor_y_atas_waj,                      koor_x_baw_waj,
koor_y_baw_waj);
cac_2:=0;
end; // end      if cac_2
end; //end for awal

```

dan kode program seluruhnya adalah :

```

unit Unit1;

interface

uses
  Windows,    Messages,    SysUtils,    Variants,
Classes,    Graphics,    Controls,    Forms,
  Dialogs,    ExtCtrls,    StdCtrls,    ExtDlgs,    Menus,
jpeg;

type
  TForm1 = class(TForm)
    Button1: TButton;
    Button2: TButton;
    OpenDialog1: TOpenDialog;
    SaveDialog1: TSaveDialog;
    Image1: TImage;
    Image2: TImage;

```

```
Image3: TImage;
Button3: TButton;
Image4: TImage;
Button4: TButton;
MainMenu1: TMainMenu;
Preprocessing1: TMenuItem;
OReGreyKon1: TMenuItem;
Latih1: TMenuItem;
Lokalisir1: TMenuItem;
GroupBox1: TGroupBox;
Button5: TButton;
Label1: TLabel;
Button6: TButton;
Label2: TLabel;
Button7: TButton;
Button8: TButton;
Refleksil: TMenuItem;
Edit2: TEdit;
Edit3: TEdit;
Edit4: TEdit;
Edit5: TEdit;
UpdatedKoordinat1: TMenuItem;
Button9: TButton;
GroupBox2: TGroupBox;
Button10: TButton;
Button11: TButton;
Edit6: TEdit;
Edit7: TEdit;
Edit8: TEdit;
Image5: TImage;
Label3: TLabel;
Label4: TLabel;
Button12: TButton;
Button13: TButton;
Edit9: TEdit;
Edit10: TEdit;
Button14: TButton;
Button15: TButton;
Edit1: TEdit;
```

```

GroupBox3: TGroupBox;
Memo1: TMemo;
StaticText1: TStaticText;
StaticText2: TStaticText;
Button16: TButton;
Label5: TLabel;
Label6: TLabel;
Label7: TLabel;
procedure Button1Click(Sender: TObject);
procedure Button2Click(Sender: TObject);
procedure Button3Click(Sender: TObject);
procedure Button4Click(Sender: TObject);
procedure OReGreyKon1Click(Sender: TObject);
procedure Lokalisir1Click(Sender: TObject);
procedure FormActivate(Sender: TObject);
procedure Button5Click(Sender: TObject);
procedure Button6Click(Sender: TObject);
procedure Button7Click(Sender: TObject);
procedure Button8Click(Sender: TObject);
procedure RefleksilClick(Sender: TObject);
procedure UpdatedKoordinat1Click(Sender:
TObject);
procedure Button9Click(Sender: TObject);
procedure Button10Click(Sender: TObject);
procedure Button11Click(Sender: TObject);
procedure Button12Click(Sender: TObject);
procedure Button13Click(Sender: TObject);
procedure Button14Click(Sender: TObject);
procedure Button15Click(Sender: TObject);
procedure Button16Click(Sender: TObject);

private
  { Private declarations }
public
  { Public declarations }
end;

var

```

```

Form1: TForm1;
      citral,citra2, citra3, citra4 : TBitmap;
      citra5, citra6                  : TBitmap;
      sw,sh                           : word;
      X_atas, Y_atas, X_baw, Y_baw  : integer;
      t_edge                           : array
[0..1000, 0..1000] of byte;
      Fi                               : Textfile;
      NamaArsip                         : String;
      koor_x_atas_waj                  : integer;
      koor_y_atas_waj                  : integer;
      koor_x_baw_waj                  : integer;
      koor_y_baw_waj                  : integer;
      luasframe                         : integer;
      tampungkoordinat                : string;
      sensitif                          : integer;
      templating1                      : array
[0..1000, 0..1000] of byte;
      templating2                      : array
[0..1000, 0..1000] of byte;

implementation

{$R *.dfm}

procedure TForm1.Button1Click(Sender: TObject);
//load picture
begin
  opendialog1.filter := 'bitmaps | *bmp' ;
  if         opendialog1.execute      then
  citral.loadfromfile(opendialog1.filename);
  citral.pixelformat  := pf24bit; //in case
other format loaded

  Image1.Picture.Bitmap := citral;
  sw := citral.width;
  sh := citral.height;
  statictext1.caption := inttostr(sw);
  statictext2.caption := inttostr(sh);

```

```

end;

function floor(a : single) : word;
begin
  result := trunc(a);
end;

function ceiling(a : single) : word;
begin
  result := trunc(a);
  if frac(a) > 0.0001 then inc(result);
end;

procedure TForm1.Button2Click(Sender: TObject);
//reduce or enlarge
var sx1,sy1,sx2,sy2 : single;      //source field
positions
  x,y : word;                      //dest field
pixels
  destR,destG,destB : single;      //destination
colors
  sR,sG,sB : byte;                //source colors
  destWidth, destheight : word;
  f,fi2 : single;
  i,j : word;
  dx,dy,PC : single;
  color : longInt;
begin
  destwidth := strtoint(edit1.text);
  f := sw / destwidth;
  fi2 := 1/f;
  fi2 := fi2*fi2;
  //destheight := trunc(citra1.height/f);
  destheight := 600;
  with citra2 do
    begin
      width := destwidth;
      height := destheight;
    end;

```

```

//---
for y := 0 to destheight-1 do
//vertical destination pixels
begin
sy1 := f * y;
sy2 := sy1 + f;
for x := 0 to destwidth-1 do
//horizontal destination pixels
begin
sx1 := f * x;
sx2 := sx1 + f;
destR := 0; destG := 0; destB := 0;
//clear colors
for j := floor(sy1) to ceiling(sy2)-1 do
//vertical source pixels
begin
dy := 1;
if sy1 > j then begin
dy := dy-(sy1-j);
end;
if sy2 < j+1 then begin
dy := dy-(j+1-sy2);
end;
for i := floor(sx1) to ceiling(sx2)-1 do
//horizontal source pixels
begin
dx := 1;
if sx1 > i then begin
dx := dx-(sx1-i);
end;
if sx2 < i+1 then begin
dx := dx-(i+1-sx2);
end;
color := citral.canvas.pixels[i,j];
sR := color and $ff;
sG := (color shr 8) and $ff;
sB := (color shr 16) and $ff;
PC := dx*dy*fi2;
destR := destR + sR*PC;

```

```

        destG := destG + sG*PC;
        destB := destB + sB*PC;
    end;//for i
    end;//for j
    citra2.Canvas.Pixels[x,y] := RGB(trunc(destR),trunc(destG),trunc(destB));
end;//for x
end;//for y

Image2.Picture.Bitmap := citra2;
citra2.SaveToFile('resizinghasil.bmp');
end;

procedure TForm1.Button3Click(Sender: TObject);
var temp : PByteArray;
    i,j : integer;
    x : byte;
    digit : byte;
begin
    citra3 := TBitmap.Create;
    citra3.LoadFromFile('resizinghasil.bmp');

    for j:=0 to citra3.Height-1 do
    begin
        temp := citra3.ScanLine[j];
        i:=0;
        repeat
            x
            :=round((0.11*temp[i])+(0.59*temp[i+1])+(0.3*temp[i+2]));
            for digit:=0 to 2 do
                temp[i+digit]:=x;
            inc(i,3);
        until i >= 3*citra3.Width-1;
    end;
    Image3.Picture.Bitmap := citra3;
    citra3.SaveToFile('greyscale.bmp');
    citra4:=citra3;
end;

```

```

procedure TForm1.Button4Click(Sender: TObject);
const sobel : array[0..1,0..2,0..2] of smallint
=
  ((((-1,0,1),(-2,0,2),(-1,0,1)),
    ((-1,-2,-1),(0,0,0),(1,2,1)));
   prewitt     : array[0..1,0..2,0..2]     of
smallint =
  ((((-1,0,1),(-1,0,1),(-1,0,1)),
    ((-1,-1,-1),(0,0,0),(1,1,1)));

var row           : array[0..8] of pbytearray;
    col            : pbytearray;
    x,y            : smallint;
    i,j,k,p       : smallint;
    citra_tampung : tbitmap;
    sum,jum        : longint;
    pos_x,pos_y   : integer;

begin
  P:=-100;
  pos_y:=0;
  citra_tampung := tbitmap.Create;
  citra_tampung.Assign(citra3);
  for y:=1 to citra4.Height-2 do
  begin
    for i:=-1 to 1 do
      row[i+1]:= citra_tampung.ScanLine[y+i];
      col := citra4.ScanLine[y];
      x:=3;
      inc(pos_y,1);
      pos_x:=1;
      repeat
        sum := 0;
        for i:=-1 to 1 do
          for j:=-1 to 1 do
            sum:=sum+(sobel[0,i+1,j+1]*row[i+1,x+j*3]);
            jum:=0;

```

```

        for i:=-1 to 1 do
            for j:=-1 to 1 do

jum:=jum+(sobel[1,i+1,j+1]*row[i+1,x+j*3]);
        sum := (sum + jum)+p;
        if sum>255 then sum:=255;
        if sum<0 then sum:=0;
        for k:=0 to 2 do col[x+k]:=sum;
        t_edge[pos_y,pos_x]:=sum;
        if      t_edge[pos_y,pos_x]<20      then
t_edge[pos_y,pos_x]:=0 else
            if      t_edge[pos_y,pos_x]>=20      then
t_edge[pos_y,pos_x]:=1;
//if t_edge[pos_y,pos_x]<>0 then
//memo1.Lines.Add('['+
inttostr(pos_y) + ','+inttostr(pos_x)+ '] '+'='+'+
'+inttostr( t_edge[pos_y,pos_x]));

        inc(x,3);
        inc(pos_x,1);
        until x>=3*(citra4.Width-4);
    end;
Image4.Picture.bitmap := citra4;
citra4.SaveToFile('convolusi.bmp');
citra_tampung.free;
edit6.Text:=inttostr(pos_x);
edit7.Text:=inttostr(pos_y);

end;

procedure      TForm1.OReGreyKon1Click(Sender:
TObject);
begin
button1.click;
button2.click;
button3.click;
button4.click;
Image5.Picture.bitmap := citra2;

```

```

end;

procedure TForm1.Lokalisir1Click(Sender:
TObject);
begin
Image4.Picture.bitmap := citra4;
//-----melokalisir-----
image4.Canvas.Brush.Style:=bsClear;
image4.Canvas.Pen.Width:=3;
image4.Canvas.Pen.Color:=clRed;
image4.Canvas.Rectangle(X_atas, Y_atas, X_baw,
Y_baw);
//-----melokalisir-----

end;

procedure TForm1.FormActivate(Sender: TObject);
begin
X_atas:=34;
Y_atas:=45;
X_baw:=74;
Y_baw:=85;
//----- BOBOT
NamaArsip:='koordinat.txt';
AssignFile(Fi,NamaArsip);
reset(Fi);
//-----
while not Eof(Fi) do
begin
begin
Readln(Fi,tampungkoordinat);
X_atas:=strToInt(tampungkoordinat);
Readln(Fi,tampungkoordinat);
Y_atas:=strToInt(tampungkoordinat);
Readln(Fi,tampungkoordinat);
X_baw:=strToInt(tampungkoordinat);
Readln(Fi,tampungkoordinat);
Y_baw:=strToInt(tampungkoordinat);

end;

```

```

CloseFile(Fi);
luasframe:=(Y_baw-Y_atas)*(X_baw-X_atas);
edit9.Text:=inttostr(luasframe);
sensitif:=strToInt(edit10.Text);
end;

procedure TForm1.Button5Click(Sender: TObject);
begin
X_atas:=X_atas+2;
X_baw:=X_baw+2;
lokalisir1.click;
Updatedkoordinat1.Click;

end;

procedure TForm1.Button6Click(Sender: TObject);
begin
X_atas:=X_atas-2;
X_baw:=X_baw-2;
lokalisir1.click;
Updatedkoordinat1.Click;
end;

procedure TForm1.Button7Click(Sender: TObject);
begin
Y_atas:=Y_atas+2;
Y_baw:=Y_baw+2;
lokalisir1.click;
Updatedkoordinat1.Click;
end;

procedure TForm1.Button8Click(Sender: TObject);
begin
Y_atas:=Y_atas-2;
Y_baw:=Y_baw-2;
lokalisir1.click;
Updatedkoordinat1.Click;
end;

```

```

procedure TForm1.RefleksilClick(Sender: TObject);
begin
//-----melokalisir-----
image2.Canvas.Brush.Style:=bsClear;
image2.Canvas.Pen.Width:=3;
image2.Canvas.Pen.Color:=clRed;
image2.Canvas.Rectangle(X_atas, Y_atas, X_baw,
Y_baw);
//-----melokalisir-----
end;

procedure TForm1.UpdatedKoordinat1Click(Sender: TObject);
begin
edit2.Text:=IntToStr(X_atas);
edit3.Text:=IntToStr(Y_atas);
edit4.Text:=IntToStr(X_baw);
edit5.Text:=IntToStr(Y_baw);
//Arsip
NamaArsip:='koordinat.txt';
AssignFile(Fi,NamaArsip);
Rewrite(Fi);

writeln(Fi,IntToStr(X_atas));
writeln(Fi,IntToStr(Y_atas));
writeln(Fi,IntToStr(X_baw));
writeln(Fi,IntToStr(Y_baw));

CloseFile(Fi);
end;

procedure TForm1.Button9Click(Sender: TObject);
begin
Image2.Picture.Bitmap := citra2;
lokalisir1.click;
refleksil.click;
Updatedkoordinat1.Click;
end;

```

```

procedure TForm1.Button10Click(Sender: TObject);
var
  i,j           : integer;
  x             : byte;
  pos_x,pos_y  : integer;

begin
  //plating
  pos_y:=0;

  //Arsip
  NamaArsip:='bit.txt';
  AssignFile(Fi,NamaArsip);
  rewrite(Fi);

  for j:=Y_atas to Y_baw-1 do
    begin

      i:=X_atas;
      inc(pos_y,1);
      pos_x:=1;
      repeat
        x:=t_edge[j,i];
        //templating adalah template base yg
        akan dimatch dgn t_edge d detek
        templating1[pos_y,pos_x]:=x;

      writeln(Fi,inttostr(templating1[pos_y,pos_x]));
      inc(i,1);
      inc(pos_x,1);
      until i >= X_baw-1;
      //writeln(Fi);
    end;
  CloseFile(Fi);
  edit6.Text:=inttostr(pos_x);
  edit7.Text:=inttostr(pos_y);

  //-----

```

```

end;

procedure TForm1.Button11Click(Sender: TObject);
var pos_x,pos_y : integer;
    cac_x,cac_y : integer;
    ini_x,ini_y : integer;
    cac_1, cac_2 : integer;
    baca : string;
    i,j : integer;
    jarak_bc_weed_1 : real;
    tampung_bc_baw : integer;
    tampung_bc_atas : integer;
    tampung_bc_bag1 : real;
    jarak_bc_weed_2 : real;
    tampung_bc_baw2 : integer;
    tampung_bc_atas2 : integer;
    tampung_bc_bag2 : real;
    kulczynski_1, kulczynski_2 : real;
    a,b,c,d : array [1..100]
of integer;
    setclear : integer;

begin
  Image5.Picture.bitmap := citra2;
  for setclear:=1 to 100 do
  begin
    a[setclear]:=0;
    b[setclear]:=0;
    c[setclear]:=0;
    d[setclear]:=0;
  end;
  //-----loading plating 1
  pos_y:=0;
  //Arsip
  NamaArsip:='bit.txt';
  AssignFile(Fi,NamaArsip);
  reset(Fi);

  for j:=Y_atas to Y_baw-1 do

```

```

begin

    i:=X_atas;
    inc(pos_y,1);
    pos_x:=1;
    repeat
        readln(Fi,baca);

templating1[pos_y,pos_x]:=strtoint(baca);
    inc(i,1);
    inc(pos_x,1);
until i >= X_baw-1;

end;
CloseFile(Fi);

//-----
//-----loading plating 2
pos_y:=0;
//Arsip
NamaArsip:='bit2.txt';
AssignFile(Fi,NamaArsip);
reset(Fi);

for j:=Y_atas to Y_baw-1 do
begin

    i:=X_atas;
    inc(pos_y,1);
    pos_x:=1;
    repeat
        readln(Fi,baca);

templating2[pos_y,pos_x]:=strtoint(baca);
    inc(i,1);
    inc(pos_x,1);
until i >= X_baw-1;

```

```

    end;
    CloseFile(Fi);

    //-----
    for pos_y:=1 to citra2.height-40 do
        for pos_x:=1 to (citra2.width-40) do
            //if pos_y mod 2 = 0 then
            begin

                ini_y:=0;
                cac_1:=0;
                cac_2:=0;
                //-----set to 0
                jarak_bc_weed_1:=0;
                tampung_bc_atas:=0;
                tampung_bc_baw:=0;
                jarak_bc_weed_2:=0;
                tampung_bc_atas2:=0;
                tampung_bc_baw2:=0;

                for cac_y:=pos_y to (pos_y+Y_baw-Y_atas-1)
do
begin
inc(ini_y,1);
ini_x:=0;
for cac_x:=pos_x to (pos_x+X_baw-X_atas-1)
do
begin
inc(ini_x,1);

//bagian vip pembanding pola

if t_edge[cac_y,
cac_x]=templatting1[ini_y,ini_x] then
inc(cac_1,1);

```

```

if                               t_edge[cac_y,
cac_x]=templating2[ini_y,ini_x]      then
inc(cac_2,1);

//hitung OTUs ciri 1 Operational Taxonomic
Units

if      (t_edge[cac_y,      cac_x]=1)      AND
(templating1[ini_y,ini_x]=1) then a[1]:=a[1]+1;
  if      (t_edge[cac_y,      cac_x]=0)      AND
(templating1[ini_y,ini_x]=1) then b[1]:=b[1]+1;
    if      (t_edge[cac_y,      cac_x]=1)      AND
(templating1[ini_y,ini_x]=0) then c[1]:=c[1]+1;
      if      (t_edge[cac_y,      cac_x]=0)      AND
(templating1[ini_y,ini_x]=0) then d[1]:=d[1]+1;

//hitung OTUs ciri 2 Operational Taxonomic
Units

if      (t_edge[cac_y,      cac_x]=1)      AND
(templating2[ini_y,ini_x]=1) then a[2]:=a[2]+1;
  if      (t_edge[cac_y,      cac_x]=0)      AND
(templating2[ini_y,ini_x]=1) then b[2]:=b[2]+1;
    if      (t_edge[cac_y,      cac_x]=1)      AND
(templating2[ini_y,ini_x]=0) then c[2]:=c[2]+1;
      if      (t_edge[cac_y,      cac_x]=0)      AND
(templating2[ini_y,ini_x]=0) then d[2]:=d[2]+1;

end; //end for cac_x
end; //end for cac_y

//total dan hitung KULCZYNSKI
kulczynski_1:= a[1]/(b[1]+c[1]);
kulczynski_2:= a[2]/(b[2]+c[2]);

if cac_1>=(luasframe-sensitif) then
if (kulczynski_1>0) then
begin
//label3.caption:=inttostr(cac_1);

```

```

        //maka catat koordinat awal dan ahir frame
        koor_x_atas_waj:=pos_x;
        koor_y_atas_waj:=pos_y;
        koor_x_baw_waj:=koor_x_atas_waj+           X_baw-
X_atas ;
        koor_y_baw_waj:=pos_y+ Y_baw-Y_atas;

        //frame

        image5.Canvas.Brush.Style:=bsclear;
        image5.Canvas.Pen.Width:=5;
        image5.Canvas.Pen.Color:=clblue;
        image5.Canvas.Ellipse(koor_x_atas_waj,
        koor_y_atas_waj,                      koor_x_baw_waj,
        koor_y_baw_waj);
        cac_1:=0;
        end; // end    if cac_1

        if cac_2>=(luasframe-sensitif) then
        if (kulczynski_2>0) then
        begin
        //label5.caption:=inttostr(cac_2);
        //maka catat koordinat awal dan ahir frame
        koor_x_atas_waj:=pos_x;
        koor_y_atas_waj:=pos_y;
        koor_x_baw_waj:=koor_x_atas_waj+           X_baw-
X_atas ;
        koor_y_baw_waj:=pos_y+ Y_baw-Y_atas;

        //frame

        image5.Canvas.Brush.Style:=bsclear;
        image5.Canvas.Pen.Width:=5;
        image5.Canvas.Pen.Color:=clgreen;
        image5.Canvas.Ellipse(koor_x_atas_waj,
        koor_y_atas_waj,                      koor_x_baw_waj,
        koor_y_baw_waj);
        cac_2:=0;
        end; // end    if cac_2

```

```

end; //end for awal
//if           cac_1>1500      then
edit8.Text:=inttostr(cac_1);

end;

procedure TForm1.Button12Click(Sender: TObject);
begin
X_atas:=X_atas-1;
Y_atas:=Y_atas-1;
X_baw:=X_baw+1;
Y_baw:=Y_baw+1;
luasframe:=(Y_baw-Y_atas)*(X_baw-X_atas);
edit9.Text:=inttostr(luasframe);
end;

procedure TForm1.Button13Click(Sender: TObject);
begin
X_atas:=X_atas+1;
Y_atas:=Y_atas+1;
X_baw:=X_baw-1;
Y_baw:=Y_baw-1;
luasframe:=(Y_baw-Y_atas)*(X_baw-X_atas);
edit9.Text:=inttostr(luasframe);
end;

procedure TForm1.Button14Click(Sender: TObject);
begin
sensitif:=sensitif-1;
edit10.Text:=inttostr(sensitif);
end;

procedure TForm1.Button15Click(Sender: TObject);
begin
sensitif:=sensitif+1;
edit10.Text:=inttostr(sensitif);
end;

procedure TForm1.Button16Click(Sender: TObject);

```

```

var
  i,j : integer;
  x : byte;
  pos_x,pos_y : integer;

begin
  //plating
  pos_y:=0;

  //Arsip
  NamaArsip:='bit2.txt';
  AssignFile(Fi,NamaArsip);
  rewrite(Fi);

  for j:=Y_atas to Y_baw-1 do
    begin

      i:=X_atas;
      inc(pos_y,1);
      pos_x:=1;
      repeat
        x:=t_edge[j,i];
        //templating adalah template base yg
        akan dimatch dgn t_edge d detek
        templating2[pos_y,pos_x]:=x;

      writeln(Fi,inttostr(templating2[pos_y,pos_x]));
      inc(i,1);
      inc(pos_x,1);
      until i >= X_baw-1;
      //writeln(Fi);
    end;
  CloseFile(Fi);
  edit6.Text:=inttostr(pos_x);
  edit7.Text:=inttostr(pos_y);

  //-----

```

```
end;
initialization

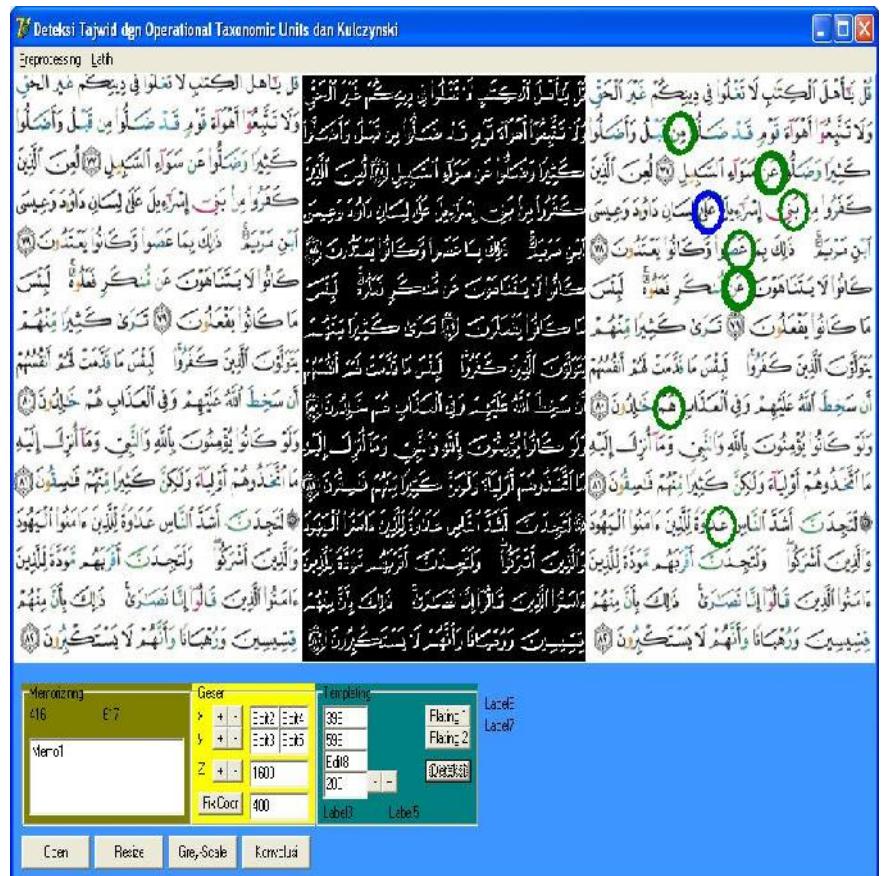
  citral := Tbitmap.create;
  with citral do
    begin
      pixelformat := pf24bit;
      width := 100; height := 100;
    end;
  citra2 := TBitmap.create;
  with citra2 do
    begin
      pixelformat := pf24bit;
      width := 100; height := 100;
    end;
  sw := 100; sh := 100;

finalization

  citral.free;
  citra2.free;

end.
```

Contoh tampilan form keluaran programnya adalah sebagai berikut



# BAB 4

## TRANSFORMASI INTEGRAL

### 4.1 Transformasi Fourier

Transformasi Fourier pada sinyal 1 dimensi  $f(x) = \{f(0), f(1), f(2), \dots, f(N-1)\}$  berukuran N, dengan indeks x bernilai dari 0 hingga N-1, akan menghasilkan sinyal 1 dimensi  $F(u) = \{F(0), F(1), F(2), \dots, F(N-1)\}$ , dengan  $F(u)$  dinyatakan sebagai :

$$F(u) = \int_{-\infty}^{\infty} f(t) e^{-jut} dt$$

dimana  $F(u)$  adalah fungsi dalam domain frekwensi dan  $u$  adalah frekwensi radial  $0 \leq u \leq f$  atau dapat dituliskan bahwa  $u = 2\pi f$  dan  $j = \sqrt{-1}$  merupakan bilangan kompleks.

Dengan memanfaatkan persamaan Euler :

$$e^{j\theta} = \cos \theta + j \sin \theta$$

maka persamaan Fourier dapat dinyatakan dalam bentuk trigonometri sebagai berikut :

$$F(u) = \frac{1}{N} \left( \sum_{x=0}^{N-1} f(x) \cos(2\pi xu / N) - j \sum_{x=0}^{N-1} f(x) \sin(2\pi xu / N) \right)$$

atau

$$F(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) \left[ \cos\left(\frac{2\pi fx}{N}\right) - j \sin\left(\frac{2\pi fx}{N}\right) \right]$$

Hasil dari Transformasi Fourier mengandung nilai riil  $R(u)$  dan imajiner  $I(u)$ , dan untuk menghilangkan nilai imajiner tersebut maka perlu perhitungan Magnitudo (Spektrum) dan sudut (phase). Adapun Magnitudo dapat dihitung dengan cara :

$$|F(u)| = \sqrt{R(u)^2 + I(u)^2}$$

Sedangkan phase dapat dihitung dengan persamaan :

$$\phi(u) = \tan^{-1} \frac{I(u)}{R(u)}.$$

Sebagai contoh, ambil kasus, diketahui sebuah sinyal  $f(x) = (3, 4, 4, 5)$ , maka Transformasi Fourier Diskrit – nya dapat dihitung sebagai berikut :

$$F(0), N=4, u=0$$

$3 \left[ \cos\left(\frac{2\pi f_0}{4}\right) - j \sin\left(\frac{2\pi f_0}{4}\right) \right]$	3
$4 \left[ \cos\left(\frac{2\pi f_1}{4}\right) - j \sin\left(\frac{2\pi f_1}{4}\right) \right]$	4
$4 \left[ \cos\left(\frac{2\pi f_2}{4}\right) - j \sin\left(\frac{2\pi f_2}{4}\right) \right]$	4
$5 \left[ \cos\left(\frac{2\pi f_3}{4}\right) - j \sin\left(\frac{2\pi f_3}{4}\right) \right]$	5

$$F(0) = \frac{1}{4} (3 + 4 + 4 + 5) = 4$$

$F(1), N=4, u=0$

$3 \left[ \cos\left(\frac{2f_{1.0}}{4}\right) - j \sin\left(\frac{2f_{1.0}}{4}\right) \right]$	3
$4 \left[ \cos\left(\frac{2f_{1.1}}{4}\right) - j \sin\left(\frac{2f_{1.1}}{4}\right) \right]$	-4j
$4 \left[ \cos\left(\frac{2f_{1.2}}{4}\right) - j \sin\left(\frac{2f_{1.2}}{4}\right) \right]$	-4
$5 \left[ \cos\left(\frac{2f_{1.3}}{4}\right) - j \sin\left(\frac{2f_{1.3}}{4}\right) \right]$	5j

$$F(1) = \frac{1}{4} (3 - 4j - 4 + 5j) = -0,25 + 0,25j$$

$F(2), N=4, u=0$

$3 \left[ \cos\left(\frac{2f_{2.0}}{4}\right) - j \sin\left(\frac{2f_{2.0}}{4}\right) \right]$	3
$4 \left[ \cos\left(\frac{2f_{2.1}}{4}\right) - j \sin\left(\frac{2f_{2.1}}{4}\right) \right]$	-4
$4 \left[ \cos\left(\frac{2f_{2.2}}{4}\right) - j \sin\left(\frac{2f_{2.2}}{4}\right) \right]$	4
$5 \left[ \cos\left(\frac{2f_{2.3}}{4}\right) - j \sin\left(\frac{2f_{2.3}}{4}\right) \right]$	-5

$$F(2) = \frac{1}{4} (3 - 4 + 4 - 5) = -0,5$$

$$F(3), N=4, u=0$$

$3 \left[ \cos\left(\frac{2f_3}{4}\right) - j \sin\left(\frac{2f_3}{4}\right) \right]$	3
$4 \left[ \cos\left(\frac{2f_{3.1}}{4}\right) - j \sin\left(\frac{2f_{3.1}}{4}\right) \right]$	$4j$
$4 \left[ \cos\left(\frac{2f_{3.2}}{4}\right) - j \sin\left(\frac{2f_{3.2}}{4}\right) \right]$	-4
$5 \left[ \cos\left(\frac{2f_{3.3}}{4}\right) - j \sin\left(\frac{2f_{3.3}}{4}\right) \right]$	-5j

$$F(3) = \frac{1}{4}(3 + 4j - 4 - 5j) = -0,25 - 0,25j$$

Spektrum atau magnitudo dari hasil di atas berturut-turut adalah :

$$|F(u)| = \sqrt{R(u)^2 + I(u)^2}$$

$$|F(0)| = 4$$

$$|F(1)| = \sqrt{-0,25^2 + 0,25^2} = 0,35$$

$$|F(2)| = 0,5$$

$$|F(3)| = \sqrt{-0,25^2 + -0,25^2} = 0,35$$

dan phase-nya adalah :

$$\varphi(u) = \tan^{-1} \frac{I(u)}{R(u)}$$

$$\varphi(0) = \tan^{-1} \frac{0}{4} = \tan^{-1}(0)$$

$$\text{„ (1)} = \tan^{-1} \frac{0,25}{-0,25} = \tan^{-1}(-1)$$

$$\text{„ (2)} = \tan^{-1} \frac{0}{-0,5} = \tan^{-1}(0)$$

$$\text{„ (3)} = \tan^{-1} \frac{-0,25}{-0,25} = \tan^{-1}(1)$$

## 4.2 Transformasi Sinyal lainnya

Beberapa pendekatan transformasi integral yang dapat digunakan untuk memanipulasi sinyal dari sebuah domain ke domain lainnya, disajikan dalam tabel berikut :

Transformasi	Formula
Kosinus Diskrit	$C(u) = \sqrt{\frac{2}{N}} r(u) \sum_{x=0}^{N-1} f(x) \cos\left[\frac{f(2x+1)u}{2N}\right]$ <p style="text-align: center;">ketentuan :</p> $r(u) = \begin{cases} \frac{1}{\sqrt{2}} & \text{untuk } u = 0 \\ 1 & \text{untuk } u \neq 0 \end{cases}$
Hartley Diskrit	$H(u) = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} f(x) \left[ \cos\left(\frac{2fux}{N}\right) - \sin\left(\frac{2fux}{N}\right) \right]$
Sinus Diskrit	$S(u) = \sqrt{\frac{2}{N+1}} \sum_{x=0}^{N-1} f(x) \sin\left[\frac{f(u+1)(x+1)}{N+1}\right]$

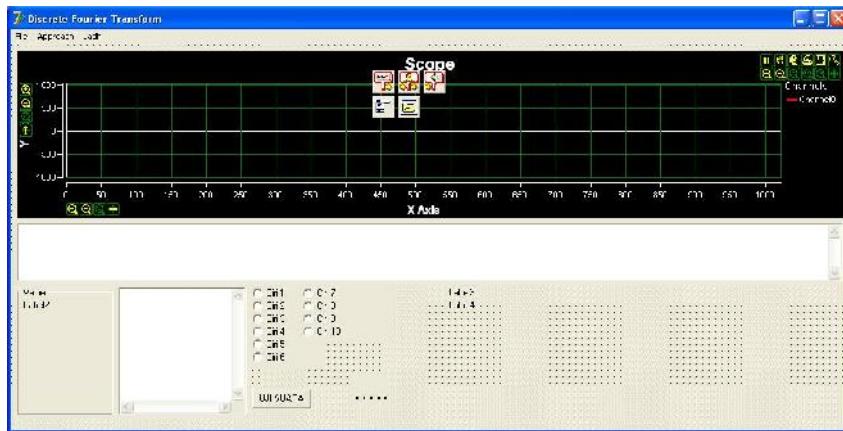
Mellin	$\{ (s) = \int_{x=0}^{N-1} x^{s-1} f(x) dx$
Laplace	$I(u) = \int_{x=0}^{N-1} f(x) e^{-ux} dt$
Weierstrass	$\check{S}(u) = \frac{1}{\sqrt{4f}} \int_{x=0}^{N-1} f(x) e^{\frac{(u-x)^2}{4}} dx$
Fourier Sinus Diskrit	$F(u) = \sum_{x=0}^{N-1} f(x) [\sin(2\pi f u x)]$
Fourier Cosinus Diskrit	$F(u) = \sum_{x=0}^{N-1} f(x) [\cos(2\pi f u x)]$
Fourier	$F(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) [\cos(\frac{2\pi f u x}{N}) - j \sin(\frac{2\pi f u x}{N})]$

### 4.3 Terapan Sistem

Sehari-hari kita tidak terlepas dari suara, misalkan suara bacaan ayat-ayat suci Al Qur'an yang selalu dikumandangkan setiap hari di masjid sebelum adzan. Permasalahannya adalah tidak semua manusia mampu mengetahui ayat berapa dan surah apa dari bacaan Qur'an yang dikumandangkan, oleh karena permasalahan itu maka diperlukan sebuah sistem yang robust yang mampu mengenal ayat-ayat Al Qur'an tersebut. Untuk membangun sebuah sistem maka diperlukan sebuah metode atau algoritma, yang dalam kasus kali ini kita pilih Transformasi Fourier Diskrit. Sistem yang dibangun dapat berupa waktu-nyata atau tidak. Untuk memudahkan pemahaman dalam membangun sistem

maka langkah awal kita adalah membangun sistem yang mampu bekerja secara tidak waktu-nyata.

Desain formulir sebagaimana desain berikut :



Lalu tuliskan kode program berikut :

```
unit Unit1;

interface

uses

  Windows, Messages, SysUtils, Variants, Classes,
  Graphics, Controls, Forms,

  Dialogs, ALCommonFilter, ALBasicGenericFilter,
  ALGenericFilter, Math,

  StdCtrls, Menus, ALBasicAudioOut, ALAudioOut,
  LPComponent,

  ALCommonPlayer, ALWavePlayer,
  SLComponentCollection, LPDrawLayers, Xpman,
  SLScope;

type

TForm1 = class(TForm)
  ALWavePlayer1: TALWavePlayer;
  ALGenericFilter1: TALGenericFilter;
  ALAudioOut1: TALAudioOut;
```

```
Memo1: TMemo;
MainMenu1: TMainMenu;
File1: TMenuItem;
Open1: TMenuItem;
OpenDialog1: TOpenDialog;
SLScope1: TSLScope;
GroupBox2: TGroupBox;
Memo2: TMemo;
ransformasi1: TMenuItem;
RadioButton1: TRadioButton;
RadioButton2: TRadioButton;
Button1: TButton;
Label1: TLabel;
Matriks1: TMenuItem;
Label2: TLabel;
DCT: TMenuItem;
raininig1: TMenuItem;
PreTrain1: TMenuItem;
Label3: TLabel;
Label4: TLabel;
RadioButton3: TRadioButton;
RadioButton4: TRadioButton;
RadioButton5: TRadioButton;
RadioButton6: TRadioButton;
RadioButton7: TRadioButton;
RadioButton8: TRadioButton;
RadioButton9: TRadioButton;
RadioButton10: TRadioButton;
procedure ALGenericFilter1ProcessData(Sender:
TObject;
```

```

    InBuffer: IALAudioBuffer;    var OutBuffer:
IALAudioBuffer;

    var SendOutputData: Boolean);
procedure Open1Click(Sender: TObject);

procedure RadioButton1Click(Sender: TObject);
procedure RadioButton2Click(Sender: TObject);
procedure Button1Click(Sender: TObject);
procedure Matriks1Click(Sender: TObject);
procedure FormActivate(Sender: TObject);

procedure vek2Click(Sender: TObject);

procedure PreTrain1Click(Sender: TObject);
procedure RadioButton3Click(Sender: TObject);
procedure RadioButton4Click(Sender: TObject);
procedure RadioButton5Click(Sender: TObject);
procedure RadioButton6Click(Sender: TObject);
procedure RadioButton7Click(Sender: TObject);
procedure RadioButton8Click(Sender: TObject);
procedure RadioButton9Click(Sender: TObject);
procedure RadioButton10Click(Sender: TObject);

private
  { Private declarations }
public
  { Public declarations }
end;

```

```

var
  Form1: TForm1;
  namafile           : string;
  fsinyal            : textfile;
  SINYAL             : array[0..19999,0..1000]
of integer;
  x_DCT              : array[0..19999,0..1000]
of real;
  N                  : integer;
  X                  : integer;
  x1                : real;
implementation
{$R *.dfm}
procedure
TForm1.ALGenericFilter1ProcessData(Sender:
TObject;
  InBuffer: IALAUDIOBUFFER; var OutBuffer:
IALAUDIOBUFFER;
  var SendOutputData: Boolean);
var
  I:integer;
begin
  namafile := 'sinyalasli.txt';
  assignfile(fsinyal,namafile);
  rewrite(fsinyal);

  for I:=0 to InBuffer.getsize()-1 do
  begin
    write(fsinyal, IntToStr(InBuffer[I,0])+' ');
    SINYAL[X,I]:= InBuffer[I,0];
  end;

```

```

closefile(fsinyal);
Matriks1.Click;
N:= InBuffer.getsize()-1;
inc(X,1); //penanda buffer N buff
end;

procedure TForm1.Open1Click(Sender: TObject);
begin
X:=0;
if not OpenDialog1.Execute then exit else
begin
    alWavePlayer1.FileName           :=
OpenDialog1.FileName;
    alWavePlayer1.Start;
end;
label1.Caption:='....';
end;

procedure      TForm1.RadioButton1Click(Sender:
TObject);
begin
radiobutton1.Enabled:=false;
//simpan peta ciri 1
    namafile := 'kernel_1.txt';
    assignfile(fsinyal,namafile);
    rewrite(fsinyal);
    writeln(fsinyal, floatToStr(x1));

    closefile(fsinyal);
end;

```

```

procedure TForm1.RadioButton2Click(Sender: TObject);
begin
  radiobutton2.Enabled:=false;
  //simpan peta ciri 2
  namafile := 'kernel_2.txt';
  assignfile(fsinyal,namafile);
  rewrite(fsinyal);
  writeln(fsinyal, floatToStr(x1));
  closefile(fsinyal);
end;

procedure TForm1.Button1Click(Sender: TObject);
var xx_uji_1, xx_uji_2 : real;
    x_uji_1, x_uji_2 : string;
    jvc_1, jvc_2 : real;
    xx_uji_3, xx_uji_4 : real;
    x_uji_3, x_uji_4 : string;
    jvc_3, jvc_4 : real;
    xx_uji_5, xx_uji_6 : real;
    x_uji_5, x_uji_6 : string;
    jvc_5, jvc_6 : real;
    xx_uji_7, xx_uji_8 : real;

```

```

x_uji_7, x_uji_8 : string;
jvc_7, jvc_8 : real;
xx_uji_9, xx_uji_10 : real;
x_uji_9, x_uji_10 : string;
jvc_9, jvc_10 : real;

begin
label1.Caption:='....';

DCT.Click;

//loading ciri 1
namafile := 'kernel_1.txt';
assignfile(fsinyal,namafile);
reset(fsinyal);
readln (fsinyal, x_uji_1);

closefile(fsinyal);

//mapping progress
xx_uji_1:=strtofloat(x_uji_1);

//hitung jarak vektor
jvc_1:=abs(abs(xx_uji_1)-abs(x1));

```

```

//loading ciri 2
    namafile := 'kernel_2.txt';
    assignfile(fsinyal,namafile);
    reset(fsinyal);
    readln (fsinyal, x_uji_2);

//mapping progress
xx_uji_2:=strtofloat(x_uji_2);

//hitung jarak vektor
jvc_2:=abs(abs(xx_uji_2)-abs(x1));

//loading ciri 3
    namafile := 'kernel_3.txt';
    assignfile(fsinyal,namafile);
    reset(fsinyal);
    readln (fsinyal, x_uji_3);

//mapping progress
xx_uji_3:=strtofloat(x_uji_3);

//hitung jarak vektor
jvc_3:=abs(abs(xx_uji_3)-abs(x1));

//loading ciri 4
    namafile := 'kernel_4.txt';
    assignfile(fsinyal,namafile);
    reset(fsinyal);
    readln (fsinyal, x_uji_4);

```

```

//mapping progress
xx_uji_4:=strtofloat(x_uji_4);

//hitung jarak vektor
jvc_4:=abs(abs(xx_uji_4)-abs(x1));

//loading ciri 5
  namofile := 'kernel_5.txt';
  assignfile(fsinyal,namofile);
  reset(fsinyal);
  readln (fsinyal, x_uji_5);

//mapping progress
xx_uji_5:=strtofloat(x_uji_5);

//hitung jarak vektor
jvc_5:=abs(abs(xx_uji_5)-abs(x1));

//loading ciri 6
  namofile := 'kernel_6.txt';
  assignfile(fsinyal,namofile);
  reset(fsinyal);
  readln (fsinyal, x_uji_6);

//mapping progress
xx_uji_6:=strtofloat(x_uji_6);

//hitung jarak vektor
jvc_6:=abs(abs(xx_uji_6)-abs(x1));

```

```

//loading ciri 7
    namafile := 'kernel_7.txt';
    assignfile(fsinyal,namafile);
    reset(fsinyal);
    readln (fsinyal, x_uji_7);

//mapping progress
xx_uji_7:=strtofloat(x_uji_7);

//hitung jarak vektor
jvc_7:=abs(abs(xx_uji_7)-abs(x1));

//loading ciri 8
    namafile := 'kernel_8.txt';
    assignfile(fsinyal,namafile);
    reset(fsinyal);
    readln (fsinyal, x_uji_8);

//mapping progress
xx_uji_8:=strtofloat(x_uji_8);

//hitung jarak vektor
jvc_8:=abs(abs(xx_uji_8)-abs(x1));

//loading ciri 9
    namafile := 'kernel_9.txt';
    assignfile(fsinyal,namafile);
    reset(fsinyal);
    readln (fsinyal, x_uji_9);

```

```

//mapping progress
xx_uji_9:=strtofloat(x_uji_9);

//hitung jarak vektor
jvc_9:=abs(abs(xx_uji_9)-abs(x1));

//loading ciri 10
  namofile := 'kernel_10.txt';
  assignfile(fsinyal,namofile);
  reset(fsinyal);
  readln (fsinyal, x_uji_10);

//mapping progress
xx_uji_10:=strtofloat(x_uji_10);

//hitung jarak vektor
jvc_10:=abs(abs(xx_uji_10)-abs(x1));

//uji vektor untuk menentukan kedekatan ciri
if jvc_1<jvc_2 then
if jvc_1<jvc_3 then
if jvc_1<jvc_4 then
if jvc_1<jvc_5 then
if jvc_1<jvc_6 then
if jvc_1<jvc_7 then
if jvc_1<jvc_8 then
if jvc_1<jvc_9 then
if jvc_1<jvc_10 then label1.Caption:='Dikenal
sebagai Ayat 1';

if jvc_2<jvc_1 then

```

```

        if jvc_2<jvc_3  then
        if jvc_2<jvc_4  then
        if jvc_2<jvc_5  then
        if jvc_2<jvc_6  then
        if jvc_2<jvc_7  then
        if jvc_2<jvc_8  then
        if jvc_2<jvc_9  then
        if jvc_2<jvc_10 then  label1.Caption:='Dikenal
sebagai Ayat 2';

        if jvc_3<jvc_1  then
        if jvc_3<jvc_2  then
        if jvc_3<jvc_4  then
        if jvc_3<jvc_5  then
        if jvc_3<jvc_6  then
        if jvc_3<jvc_7  then
        if jvc_3<jvc_8  then
        if jvc_3<jvc_9  then
        if jvc_3<jvc_10 then  label1.Caption:='Dikenal
sebagai Ayat 3';

        if jvc_4<jvc_1  then
        if jvc_4<jvc_3  then
        if jvc_4<jvc_2  then
        if jvc_4<jvc_5  then
        if jvc_4<jvc_6  then
        if jvc_4<jvc_7  then
        if jvc_4<jvc_8  then
        if jvc_4<jvc_9  then
        if jvc_4<jvc_10 then  label1.Caption:='Dikenal
sebagai Ayat 4';

```

```
if jvc_5<jvc_1  then
if jvc_5<jvc_3  then
if jvc_5<jvc_4  then
if jvc_5<jvc_2  then
if jvc_5<jvc_6  then
if jvc_5<jvc_7  then
if jvc_5<jvc_8  then
if jvc_5<jvc_9  then
if jvc_5<jvc_10 then  label1.Caption:='Dikenal
sebagai Ayat 5';

if jvc_6<jvc_1  then
if jvc_6<jvc_3  then
if jvc_6<jvc_4  then
if jvc_6<jvc_5  then
if jvc_6<jvc_2  then
if jvc_6<jvc_7  then
if jvc_6<jvc_8  then
if jvc_6<jvc_9  then
if jvc_6<jvc_10 then  label1.Caption:='Dikenal
sebagai Ayat 6';

if jvc_7<jvc_1  then
if jvc_7<jvc_3  then
if jvc_7<jvc_4  then
if jvc_7<jvc_5  then
if jvc_7<jvc_2  then
if jvc_7<jvc_6  then
if jvc_7<jvc_8  then
if jvc_7<jvc_9  then
```

```

if jvc_7<jvc_10    then   label1.Caption:='Dikenal
sebagai Ayat 7');

if jvc_8<jvc_1  then
if jvc_8<jvc_3  then
if jvc_8<jvc_4  then
if jvc_8<jvc_5  then
if jvc_8<jvc_2  then
if jvc_8<jvc_7  then
if jvc_8<jvc_6  then
if jvc_8<jvc_9  then
if jvc_8<jvc_10  then   label1.Caption:='Dikenal
sebagai Ayat 8');

if jvc_9<jvc_1  then
if jvc_9<jvc_3  then
if jvc_9<jvc_4  then
if jvc_9<jvc_5  then
if jvc_9<jvc_2  then
if jvc_9<jvc_7  then
if jvc_9<jvc_8  then
if jvc_9<jvc_6  then
if jvc_9<jvc_10  then   label1.Caption:='Dikenal
sebagai Ayat 9');

if jvc_10<jvc_1  then
if jvc_10<jvc_3  then
if jvc_10<jvc_4  then
if jvc_10<jvc_5  then
if jvc_10<jvc_2  then
if jvc_10<jvc_7  then

```

```

if jvc_10<jvc_8  then
if jvc_10<jvc_9  then
if jvc_10<jvc_6  then    label1.Caption:='Dikenal
sebagai Ayat 10');

label3.Caption:=floattostr(jvc_1);
label4.Caption:=floattostr(jvc_2);
end;

procedure TForm1.Matriks1Click(Sender: TObject);
begin
Memo1.Lines.LoadFromFile(namafайл);
end;

procedure TForm1.FormActivate(Sender: TObject);
begin
X:=0;
end;

procedure TForm1.vek2Click(Sender: TObject);
var u, Y, Z      : integer;
    ENERGI       : real;
begin
ENERGI:=0;

for Y:=0 to X do
  for u:=0 to N do  x_DCT[Y,u]:=0;
  namafайл := 'spektrum.txt';
  assignfile(fsinyal,namafайл);
  rewrite(fsinyal);

```

```

for Y:=0 to X do
    for u:=0 to N do
        begin

            for Z:=0 to N do
                begin

x_DCT[Y,u]:=x_DCT[Y,u]+(SINYAL[Y,Z]*(1/(N))*(cos(2
*3.14*(Z+1)*u/(2*N))-
(sin(2*3.14*(Z+1)*u/(2*N))));

end; //end    for Z:=0 to N do

if      x_DCT[Y,u]<>0      then      writeln(fsinyal,
floattoStr(x_DCT[Y,u])+ ' ');

ENERGI:=ENERGI+ x_DCT[Y,u];
end; //end    for u:=0 to N do
closefile(fsinyal);
Memo2.Lines.LoadFromFile(namafile);

x1:=ENERGI/(X*N);
label2.Caption:=floattostr(x1);
end;

procedure TForm1.PreTrain1Click(Sender: TObject);
begin
radiobutton1.Enabled:=true;
radiobutton2.Enabled:=true;
radiobutton3.Enabled:=true;
radiobutton4.Enabled:=true;
radiobutton5.Enabled:=true;

```

```

radiobutton6.Enabled:=true;
radiobutton8.Enabled:=true;
radiobutton7.Enabled:=true;
radiobutton9.Enabled:=true;
radiobutton10.Enabled:=true;
end;

procedure TForm1.RadioButton3Click(Sender:
TObject);
begin
radiobutton3.Enabled:=false;
//simpan peta ciri 3
  namofile := 'kernel_3.txt';
  assignfile(fsinyal,namofile);
  rewrite(fsinyal);
  writeln(fsinyal, floatToStr(x1));
  closefile(fsinyal);
end;

procedure TForm1.RadioButton4Click(Sender:
TObject);
begin
radiobutton4.Enabled:=false;
//simpan peta ciri 4
  namofile := 'kernel_4.txt';
  assignfile(fsinyal,namofile);
  rewrite(fsinyal);
  writeln(fsinyal, floatToStr(x1));
  closefile(fsinyal);
end;

```

```

procedure TForm1.RadioButton5Click(Sender:
TObject);
begin
radiobutton5.Enabled:=false;
//simpan peta ciri 5
    namafile := 'kernel_5.txt';
    assignfile(fsinyal,namafile);
    rewrite(fsinyal);
    writeln(fsinyal, floatToStr(x1));
    closefile(fsinyal);
end;

procedure TForm1.RadioButton6Click(Sender:
TObject);
begin
radiobutton6.Enabled:=false;
//simpan peta ciri 6
    namafile := 'kernel_6.txt';
    assignfile(fsinyal,namafile);
    rewrite(fsinyal);
    writeln(fsinyal, floatToStr(x1));
    closefile(fsinyal);
end;

procedure TForm1.RadioButton7Click(Sender:
TObject);
begin
radiobutton7.Enabled:=false;
//simpan peta ciri 7
    namafile := 'kernel_7.txt';
    assignfile(fsinyal,namafile);

```

```

        rewrite(fsinyal);
        writeln(fsinyal, floattoStr(x1));
        closefile(fsinyal);

end;

procedure TForm1.RadioButton8Click(Sender: TObject);
begin
  radiobutton8.Enabled:=false;
  //simpan peta ciri 8
  namafile := 'kernel_8.txt';
  assignfile(fsinyal,namafile);
  rewrite(fsinyal);
  writeln(fsinyal, floattoStr(x1));
  closefile(fsinyal);
end;

procedure TForm1.RadioButton9Click(Sender: TObject);
begin
  radiobutton9.Enabled:=false;
  //simpan peta ciri 9
  namafile := 'kernel_9.txt';
  assignfile(fsinyal,namafile);
  rewrite(fsinyal);
  writeln(fsinyal, floattoStr(x1));
  closefile(fsinyal);
end;

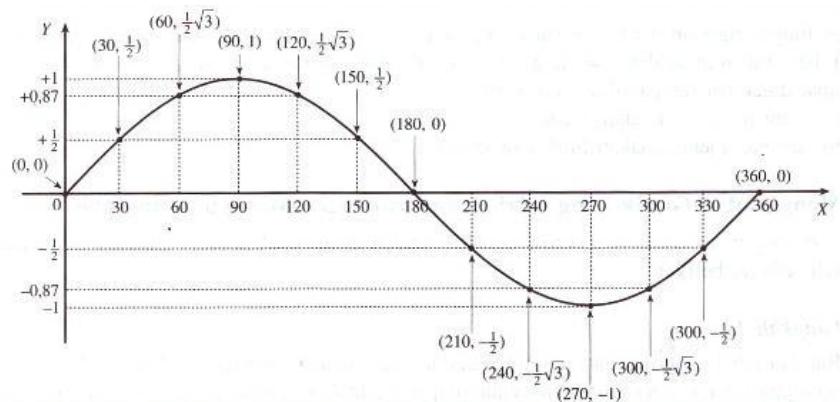
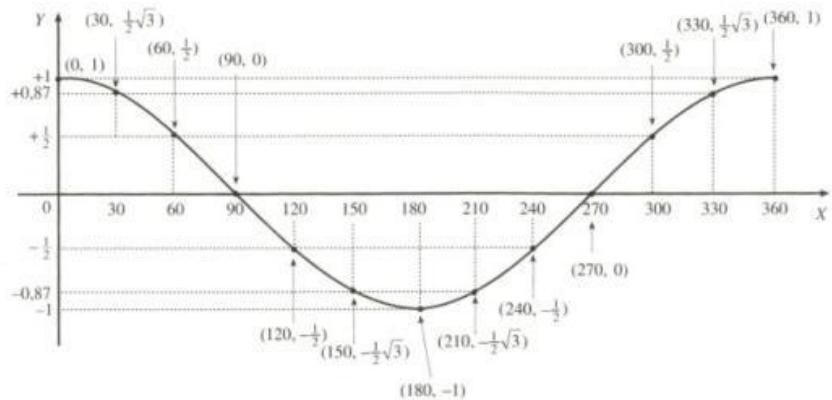
procedure TForm1.RadioButton10Click(Sender: TObject);

```

```

begin
radiobutton10.Enabled:=false;
//simpan peta ciri 10
  namafile := 'kernel_10.txt';
  assignfile(fsinyal,namafile);
  rewrite(fsinyal);
  writeln(fsinyal, floatToStr(x1));
  closefile(fsinyal);
end;
end.

```



# **BAB 5**

## **SISTEM WAKTU NYATA PENGUJIAN HAFALAN AYAT- AYAT SUCI AL QUR’AN**

### **5.1 Pendahuluan**

Tidak dipungkiri lagi, bahwa peminatan terhadap hafalan Al Qur'an semakin banyak secara kuantitas dan mengalami perkembangan secara pesat dalam metode pendekatan hafalannya. hal ini dapat dilihat dengan tumbuh suburnya dengan mudah lembaga-lembaga tahfidz Qur'an. Setiap lembaga tahfidz Qur'an memiliki cara-cara sendiri dalam menerapkan praktek hafalan Qur'an kepada para santrinya, dan pastinya mereka dibimbing oleh seorang guru yang sudah menguasai Qur'an secara hafalan. Dengan peserta hafalan Qur'an yang cukup besar dan keterbatasan kemampuan pelayanan guru dalam waktu yang singkat, maka telah munculah permasalahan baru yang menarik untuk diangkat dan dicari solusi terapannya secara efisien dan hemat waktu. Maka penelitian yang diajukan adalah bagaimana membangun sebuah sistem pengujian hafalan Qur'an yang mengadopsi kemampuan hafalan guru untuk diterapkan sebagai pengawasan dan pengujian hafalan para peserta hafalan Qur'an.

Untuk membangun sebuah sistem hafalan Qur'an yang handal secara waktu-nyata maka diperlukan sebuah atau lebih algoritma.

Dalam penelitian yang diajukan pada kesempatan ini, peneliti menggunakan dua pendekatan (1) Transformasi Mellin, dan (2) Transformasi Walsh. Adapun alasan pemilihan kedua algoritma ini adalah Transformasi Mellin lebih mewakili algoritma yang efisien yang bersifat eksponensial dan Transformasi Walsh mewakili algoritma yang berbasis non-sinusoidal, atau memiliki fungsi basis -1 dan 1.

### **Transformasi Mellin**

Transformasi bertujuan mengubah sinyal-sinyal digital dari domain waktu menjadi sinyal-sinyal pada domain frekuensi dengan formulasi berikut :

$$\{ (s) = \int_{x=0}^{N-1} x^{s-1} f(x) dx$$

dimana

$\{ (s)$  = Transformasi Mellin

$s$  = indeks sinyal pada domain frekuensi

$N$  = total cacah sinyal

$x$  = indeks sinyal pada domain waktu

$f(x)$  = nilai sinyal.

### **Transformasi Walsh**

Tidak seperti transformasi yang dibahas sebelumnya yang memiliki fungsi basis bilangan pecahan antara selang -1 hingga 1 (sin

dan cos), Transformasi Walsh memiliki sifat yang non-sinusoidal, atau dengan kata lain Transformasi Walsh memiliki fungsi basis -1 atau 1.

Transformasi Walsh pada sinyal  $f(x)$  dinyatakan sebagai :

$$W(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) \prod_{i=0}^{n-1} (-1)^{b_i(x)b_{n-1-i}(u)}$$

dengan  $u=0, 1, 2, \dots, N-1$  dan  $x=0, 1, 2, \dots, N-1$ , dan nilai n mengikuti aturan

$$N = 2^n.$$

Sebagai contoh jika  $N=8$  maka  $n=3$ .

$b_i(x)$  menyatakan bit ke i dari bentuk biner x, misal  $x=4$  atau secara biner =100, maka  $b_0(x) = 0$ ,  $b_1(x) = 0$ , dan  $b_2(x) = 1$ . Demikian juga berlaku untuk  $b_i(u)$ .

Fungsi basis Transformasi Walsh dinyatakan dengan formula berikut :

$$g(x,u) = \prod_{i=0}^{n-1} (-1)^{b_i(x)b_{n-1-i}(u)}$$

dan dari persamaan fungsi basis (kernel) Transformasi Walsh untuk  $N=8$  akan dihasilkan nilai-nilai kernel berikut :

x u	0	1	2	3	4	5	6	7
0	+	+	+	+	+	+	+	+
1	+	+	+	+	-	-	-	-
2	+	+	-	-	+	+	-	-
3	+	+	-	-	-	-	+	+
4	+	-	+	-	+	-	+	-
5	+	-	+	-	-	+	-	+
6	+	-	-	+	+	-	-	+
7	+	-	-	+	-	+	+	-

Untuk mendapatkan sebuah nilai pada kernel, maka dengan cara perhitungan berikut, misalkan ambil kasus  $u=1$  dan  $x=4$ , karena  $N=8$  maka  $n=3$ .

biner  $u=1$  adalah 001

biner  $x=4$  adalah 100

maka

$$b_0(u) = 1, b_1(u) = 0, \text{ dan } b_2(u) = 0$$

$$b_0(x) = 0, b_1(x) = 0, \text{ dan } b_2(x) = 1$$

$$g(x, u) = \prod_{i=0}^{n-1} (-1)^{b_i(x)b_{n-1-i}(u)}$$

$$g(4,1) = \prod_{i=0}^2 (-1)^{b_i(4)b_{n-l-i}(1)}$$

$$g(4,1) = (-1)^{(0)(0)}(-1)^{(0)(0)}(-1)^{(1)(1)} = (-1)^1 = -1$$

Kasus lain misalkan u=4 dan x=6,

biner u=4 adalah 100

biner x=6 adalah 110

maka

$$b_0(u) = 0, b_1(u) = 0, \text{ dan } b_2(u) = 1$$

$$b_0(x) = 0, b_1(x) = 1, \text{ dan } b_2(x) = 1$$

$$g(x,u) = \prod_{i=0}^{n-1} (-1)^{b_i(x)b_{n-l-i}(u)}$$

$$g(6,4) = \prod_{i=0}^2 (-1)^{b_i(6)b_{n-l-i}(4)}$$

$$g(6,4) = (-1)^{(0)(1)}(-1)^{(1)(0)}(-1)^{(1)(0)} = (-1)^0 = 1$$

Hitung Transformasi Walsh dari sinyal  $f(x)=1, 1, 1, 1, 5, 5, 5, 5$

Penyelesaian :

$$W(0) = \frac{1}{8} (1 + 1 + 1 + 1 + 5 + 5 + 5 + 5) = 3$$

$$W(1) = \frac{1}{8} (1 + 1 + 1 + 1 - 5 - 5 - 5 - 5) = -2$$

$$W(2) = \frac{1}{8} (1 + 1 - 1 - 1 + 5 + 5 - 5 - 5) = 0$$

$$W(3) = \frac{1}{8}(1+1-1-1-5-5+5+5)=0$$

$$W(4) = \frac{1}{8}(1-1+1-1+5-5+5-5)=0$$

$$W(5) = \frac{1}{8}(1-1+1-1-5+5-5+5)=0$$

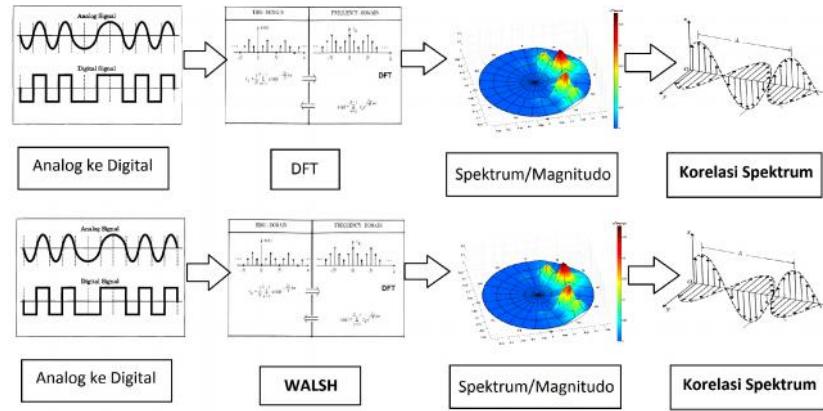
$$W(6) = \frac{1}{8}(1-1-1+1+5-5-5+5)=0$$

$$W(7) = \frac{1}{8}(1-1-1+1-5+5+5-5)=0$$

maka Transformasi Walsh dari sinyal  $f(x)=1, 1, 1, 1, 5, 5, 5, 5$  adalah  
 $W(u)=3, -2, 0, 0, 0, 0, 0, 0$ .

## 5.2 Skema Sistem

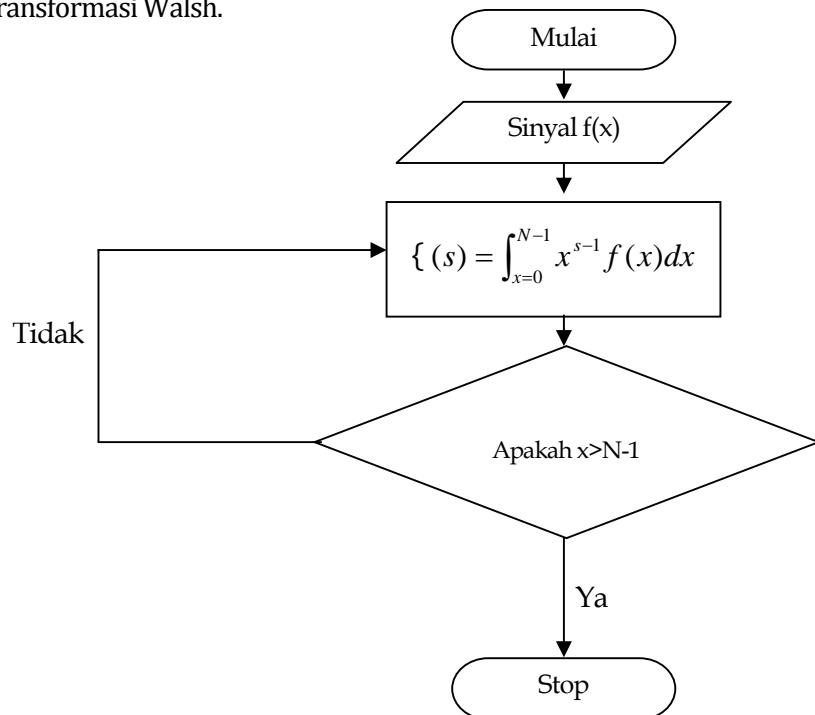
Dalam penelitian yang diajukan untuk membangun sebuah sistem yang robust, maka diperlukan alur sistem atau skema. Skema sistem yang dibangun dilustrasikan pada gambar 5.1.



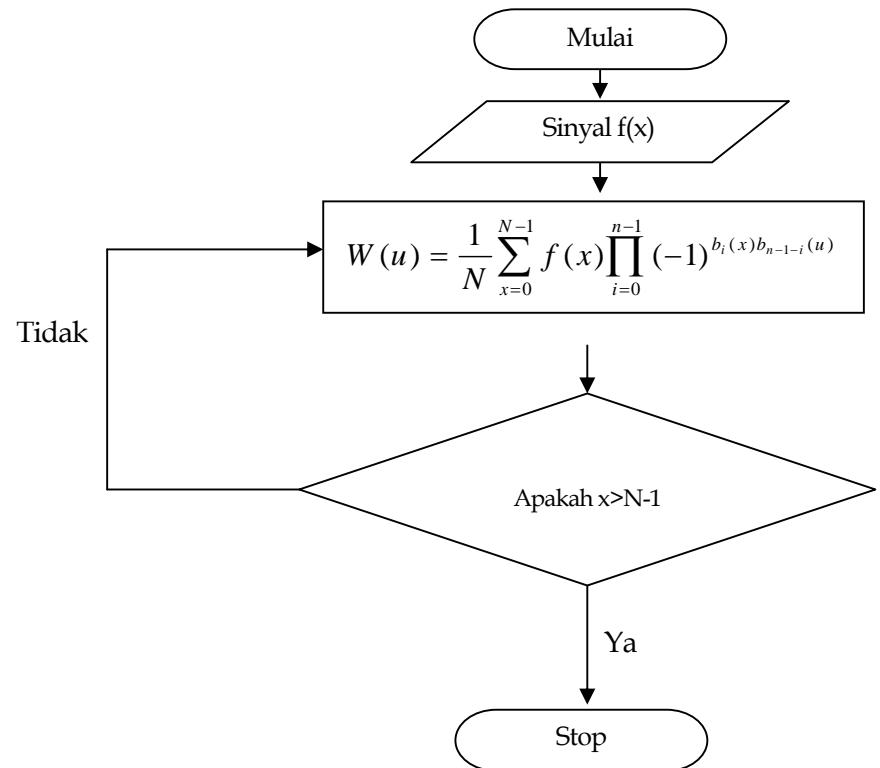
Gambar 5.1 Skema sistem pengujian pola suara

Proses sistem diawali dengan mengambil suara analog dan direkam menjadi suara digital pada domain waktu, selanjutnya DFT menggeser sinyal-sinyal pada domain waktu tersebut menjadi sinyal-sinyal pada domain frekuensi, yang selanjutnya dilakukan perhitungan spektrum atau magnitudo. Nilai-nilai spektrum inilah yang akan diajukan nilai referensi yang akan dikorelasikan dengan nilai spektrum suara pengujian.

Untuk mempertegas operasional dari kerja pada blok skema sistem, maka diperlukan diagram operasional flowchart. Berturut-turut gambar 5.2 dan 5.3 mengilustrasikan flowchart Transformasi Mellin dan Transformasi Walsh.



Gambar 5.2 Diagram alir Transformasi Mellin



Gambar 5.3 Diagram alir Transformasi Walsh

### 5.3 Hasil

Pengukuran unjuk kerja sistem dilakukan dengan berbagai tingkatan jumlah pelatihan untuk kedua transformasi. Berbagai hasil unjuk kerja sistem disajikan pada tabel 5.1 dan tabel 5.2.

Tabel 5.1 Hasil unjuk kerja sistem menggunakan Transformasi Mellin

<b>Jumlah Suara Pelatihan</b>	<b>Jumlah Suara Pengujian</b>	<b>Jumlah Pendeteksian yang benar</b>	<b>False Positive</b>	<b>Detection Rate</b>
25	100	60	0,4	0,6
50	100	67	0,33	0,67
75	100	85	0,15	0,85
100	100	90	0,1	0,9

Tabel 5.2 Hasil unjuk kerja sistem menggunakan Transformasi Walsh

<b>Jumlah Suara Pelatihan</b>	<b>Jumlah Suara Pengujian</b>	<b>Jumlah Pendeteksia n yang benar</b>	<b>False Positive</b>	<b>Detection Rate</b>
25	100	50	0,5	0,5
50	100	59	0,41	0,59
75	100	70	0,3	0,7
100	100	82	0,18	0,82

Dari hasil unjuk kerja yang ditampilkan pada tabel 5.1 dan 5.2, menunjukkan bahwa Transformasi Mellin memiliki nilai rate 0,9 dan Transformasi Walsh memiliki nilai rate 0,82. Dengan kata lain Transformasi Mellin sedikit lebih unggul dalam keakuratan unjuk kerjanya dibandingkan Transformasi Walsh.

#### 5.4 Implementasi Sistem

Secara teknis langkah-langkah yang diperlukan dalam membangun sistem adalah menyiapkan 5 form, mendesainnya, dan mengisi koding untuk masing-masing formulir sebagaimana dilustrasikan di bawah ini :



```
unit Unit1;
interface
uses
Windows, Messages, SysUtils, Variants, Classes,
Graphics, Controls, Forms, Dialogs, Menus,
ALBasicAudioOut, ALAudioOut, LPComponent,
ALCommonPlayer, ALWavePlayer, jpeg, ExtCtrls;
type
```

```

TForm1 = class(TForm)
  MainMenul: TMainMenu;
  ALWavePlayer1: TALWavePlayer;
  ALAudioOut1: TALAudioOut;
  N12: TMenuItem;
  Halaman2: TMenuItem;
  Halaman11: TMenuItem;
  Halaman31: TMenuItem;
  Image1: TImage;
  procedure Halaman2Click(Sender: TObject);
  procedure Halaman11Click(Sender: TObject);
  procedure Halaman31Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form1: TForm1;
  PENANDA_AYAT : Integer;
implementation
uses Unit2, Unit3, Unit4, Unit5;

{$R *.dfm}

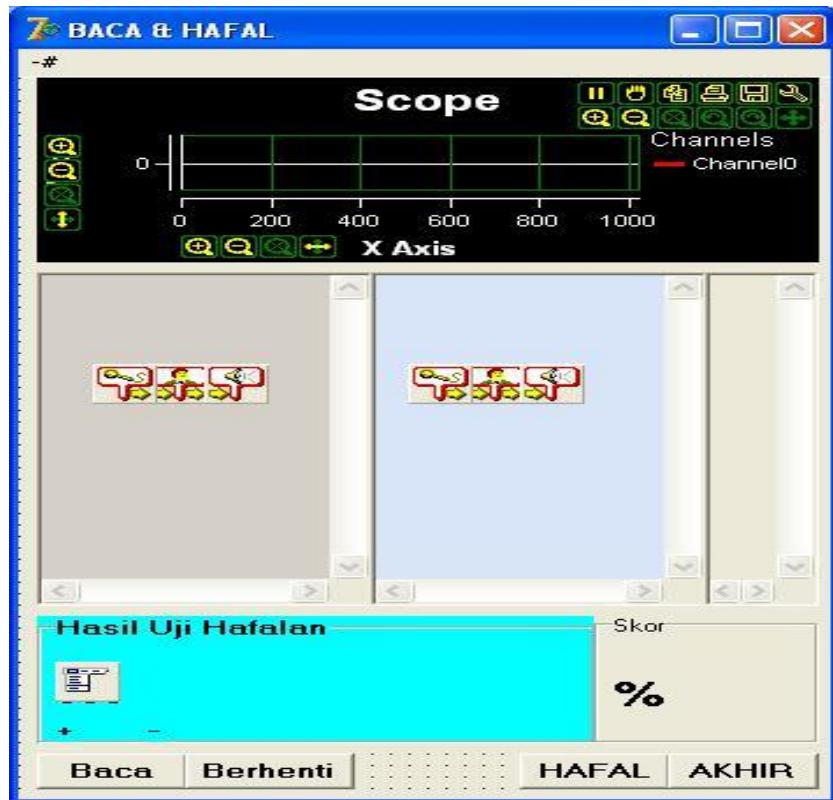
procedure TForm1.Halaman2Click(Sender: TObject);
begin
  Alwaveplayer1.Enabled:=False;
  PENANDA_AYAT:=2;
  Form2.Show;
  Form4.show;

```

```
end;

procedure TForm1.Halaman11Click(Sender: TObject);
begin
  Alwaveplayer1.Enabled:=False;
  PENANDA_AYAT:=1;
  Form2.show;
  Form3.show;
end;

procedure TForm1.Halaman31Click(Sender: TObject);
begin
  Alwaveplayer1.Enabled:=False;
  PENANDA_AYAT:=3;
  Form2.show;
  Form5.show;
end;
end.
```



```
unit Unit2;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes,
  Graphics, Controls, Forms,
  Dialogs, ExtCtrls, jpeg, LPComponent,
  ALCommonPlayer, ALWavePlayer,
  ALBasicAudioOut, ALAudioOut, ALCommonFilter,
  ALBasicGenericFilter,
  ALGenericFilter, StdCtrls,
  SLComponentCollection, LPDrawLayers, SSLScope,
  xpman,
  ALAudioIn, Menus;
type
  TForm2 = class(TForm)
```

```

        Button1: TButton;
        ALGenericFilter1: TALGenericFilter;
        ALAudioOut1: TALAudioOut;
        Button2: TButton;
        ALAudioIn1: TALAudioIn;
        ALGenericFilter2: TALGenericFilter;
        ALAudioOut2: TALAudioOut;
        Memo1: TMemo;
        Button3: TButton;
        ALAudioIn2: TALAudioIn;
        Button4: TButton;
        Memo2: TMemo;
        Memo3: TMemo;
        GroupBox1: TGroupBox;
        Label1: TLabel;
        Label2: TLabel;
        Label3: TLabel;
        MainMenu1: TMainMenu;
        N1: TMenuItem;
        N031: TMenuItem;
        N041: TMenuItem;
        N051: TMenuItem;
        N061: TMenuItem;
        SLScope1: TSLScope;
        GroupBox2: TGroupBox;
        Label4: TLabel;
        SLScope2: TSLScope;
        procedure Button1Click(Sender: TObject);

        procedure ALGenericFilter1ProcessData(Sender: TObject;

```

```

    InBuffer: IALAudioBuffer; var OutBuffer:
IALAudioBuffer;

    var SendOutputData: Boolean);

procedure Button2Click(Sender: TObject);

procedure ALGenericFilter2ProcessData(Sender:
TObject;

    InBuffer: IALAudioBuffer; var OutBuffer:
IALAudioBuffer;

    var SendOutputData: Boolean);

procedure Button3Click(Sender: TObject);
procedure Button4Click(Sender: TObject);
procedure N031Click(Sender: TObject);
procedure N041Click(Sender: TObject);
procedure N051Click(Sender: TObject);
procedure N061Click(Sender: TObject);
procedure FormActivate(Sender: TObject);

private
{ Private declarations }

public
{ Public declarations }

end;

var
Form2: TForm2;
M_sebaran, M_sebaranH : array[0..100000] of Real;
M,N, NH, Total, L, TL : Integer;
Banding, cek : Real;
Alpha : Real;
namafile, tampung : string;
fsinyal : textfile;

```

```

implementation
uses Unit3, Unit1;
{$R *.dfm}
procedure TForm2.Button1Click(Sender: TObject);
begin
    //AlwavePlayer1.Enabled:=True;
    S1scope1.Visible:=True;
    S1scope2.Visible:=False;
    Mem1.Clear;
    ALAudioIn1.Start();
    N:=0;
    Label1.Caption:='...';
    Label4.Caption:='...';
    //Struktur File
    if PENANDA_AYAT=1 then namafile := '1.txt';
    if PENANDA_AYAT=2 then namafile := '2.txt';
    if PENANDA_AYAT=3 then namafile := '3.txt';
    if PENANDA_AYAT=4 then namafile := '4.txt';
    assignfile(fsinyal,namafile);
    rewrite(fsinyal);
end;

procedure
TForm2.ALGenericFilter1ProcessData(Sender:
TObject;
    InBuffer: IALAudioBuffer; var OutBuffer:
IALAudioBuffer;
    var SendOutputData: Boolean);
var
I : Integer;
begin

```

```

inc(N,1);
Total:=0;
for I := 0 to InBuffer.GetSize() - 1 do
begin
  Total:= Total+InBuffer[ I, 0 ];
  inc(M,1);
end;

M_sebaran[N]:=Total/(InBuffer.GetSize() - 1);

//save
writeln(fsinyal, floattoStr(M_sebaran[N]));
//-----
if M_sebaran[N]<>0 then
  Memo1.Lines.Add(inttostr(N)+'
'+floattostr(M_sebaran[N]));
end;

procedure TForm2.Button2Click(Sender: TObject);
begin

  S1scope1.Visible:=False;
  S1scope2.Visible:=True;
  ALAudioIn2.Start();
  NH:=0; L:=0; TL:=0;
  Memo2.Clear;
  Memo3.Clear;
  Label1.Caption:='...';
  Label4.Caption:='...';

//loading

```

```

        if PENANDA_AYAT=1 then namafile := '1.txt';
        if PENANDA_AYAT=2 then namafile := '2.txt';
        if PENANDA_AYAT=3 then namafile := '3.txt';
        if PENANDA_AYAT=4 then namafile := '4.txt';

        assignfile(fsinyal,namafile);
        reset(fsinyal);
        N:=0;
        while not Eof(fsinyal) do
        begin
        inc(N,1);
        readln (fsinyal, tampung);
        M_sebaran[N]:=strtofloat(tampung);
        end;
        closefile(fsinyal);
        end;

procedure
TForm2.ALGenericFilter2ProcessData(Sender:
 TObject;
        InBuffer: IALAudioBuffer; var OutBuffer:
 IALAudioBuffer;
        var SendOutputData: Boolean);
var
        I : Integer;
begin
        inc(NH,1);
        Total:=0;

        for I := 0 to InBuffer.GetSize() - 1 do
        begin
        Total:= Total+InBuffer[ I, 0 ];

```

```

inc(M,1);
end;

M_sebaranH[NH]:=Total/(InBuffer.GetSize() - 1);
if M_sebaran[N]<>0 then
Memo2.Lines.Add(inttostr(NH) +
'+floattosrt(M_sebaranH[NH]));
;

//BANDINGKAN
if NH<=N then
begin
if M_sebaran[NH]<>0 then
if M_sebaranH[NH]<>0 then
if M_sebaran[NH]>M_sebaranH[NH] then
banding:=M_sebaranH[NH]/M_sebaran[NH]
else
banding:=M_sebaran[NH]/M_sebaranH[NH];

if abs(banding)>0.5 then begin inc(L,1);
Memo3.Lines.Add( '+') end
else begin inc(TL,1);
Memo3.Lines.Add( '-') end
;

end;
end;

procedure TForm2.Button3Click(Sender: TObject);
Var u,x : integer;
Real_T_MELLIN, imajiner_T_MELLIN, spektrum : real;
N_spektrum : integer;

```

```

        T_spektrum          :;
real;

        Real_T_MELLINH,   imajiner_T_MELLINH,   spektrumH
: real;

        N_spektrumH          :;
integer;

        T_spektrumH,   Energi          :;
real;

begin
ALAudioIn2.Stop();
Label2.Caption:='+:= '+inttostr(L);
Label3.Caption:='-:= '+inttostr(TL);
//cek:=1;

if L>TL then cek:=TL/L;
if cek<Alpha then Label1.Caption:='LULUS'
else Label1.Caption:='TIDAK LULUS';
//-----Transformasi Sinyal T_MELLIN

N_spektrum :=0;
T_spektrum :=0;
N_spektrumH:=0;
T_spektrumH:=0;

for u:=0 to N-1 do
  for x:=0 to N-1 do
begin
  Real_T_MELLIN:=(1/N)*x*M_sebaran[x];
  //Imajiner_T_MELLIN:=-(1/N)* M_sebaran[x] * sin
((2*PI*u*x)/N);
  //spektrum T_MELLIN

```

```

spektrum:=sqrt(sqr(real_T_MELLIN));
N_spektrum:=N_spektrum+1;
T_spektrum:=T_spektrum+spektrum ;

//---tahap II

Real_T_MELLINH:=(1/N)* M_sebaranH[x] * cos
((2*PI*u*x)/N);
//Imajiner_T_MELLINH:=- (1/N)* M_sebaranH[x] * sin
((2*PI*u*x)/N);

//spektrum T_MELLIN

spektrumH:=sqrt(sqr(real_T_MELLINH));
N_spektrumH:=N_spektrumH+1;
T_spektrumH:=T_spektrumH+spektrumH ;
//-----akhir Transformasi
end;
//-----Analisa Energi Sinyal
if T_spektrumH<T_spektrum then Energi:=
T_spektrumH/T_spektrum else
Energi:=
T_spektrum/T_spektrum;
//if cek<Alpha then
label4.Caption:=inttostr(round(Energi)*100) +
'+'%;
if cek<Alpha then
label4.Caption:=floattostr(round((L/(L+TL))*100)) +
'+'%;
end;
procedure TForm2.Button4Click(Sender: TObject);
begin
ALAUDIOIn1.Stop();

```

```
closefile(fsinyal);
end;

procedure TForm2.N031Click(Sender: TObject);
begin
Alpha:=0.3;
end;

procedure TForm2.N041Click(Sender: TObject);
begin
Alpha:=0.4;
end;

procedure TForm2.N051Click(Sender: TObject);
begin
Alpha:=0.5;
end;

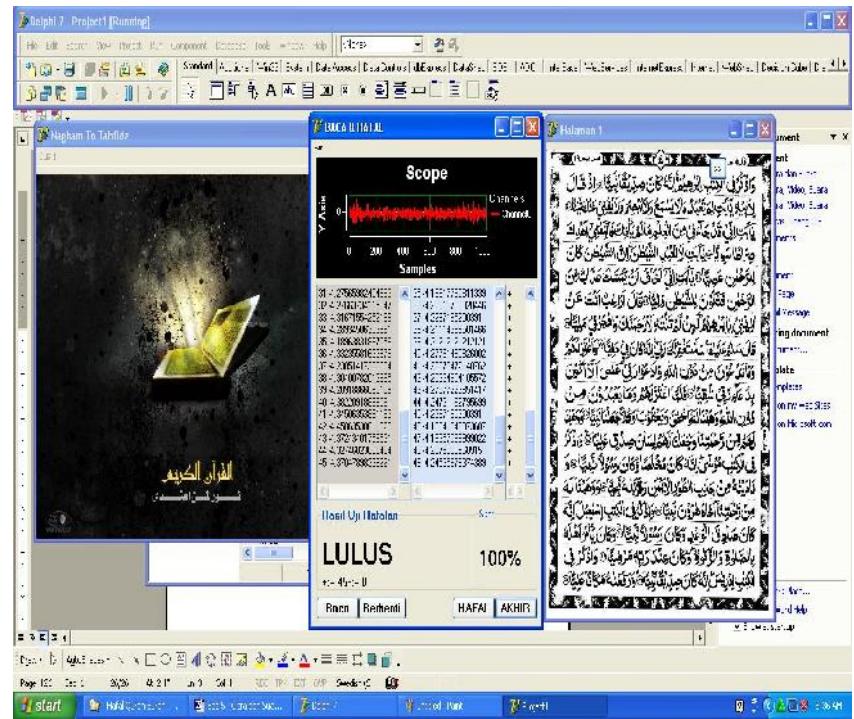
procedure TForm2.N061Click(Sender: TObject);
begin
Alpha:=0.6;
end;

procedure TForm2.FormActivate(Sender: TObject);
begin
Alpha:=0.0000;
end;
end.
```

وَإِذْ كُرِّرَ فِي الْكِتَابِ إِبْرَاهِيمَ إِنَّهُ كَانَ صَدِيقًا لِيَسَّارًا إِذْ سَأَلَ  
وَلَكِبِيرِهِ بِأَيْمَانِهِ تَعْبِدُ مَا لَا يُسْكِنُ وَلَا يَبْصِرُ وَلَا يَقْعُدُ عَنْ قِصْبَيْنِهِ  
يَا أَبْرَاهِيمَ إِنِّي قَدْ يَحْأَلُ فِي مِنْ الْعِلْمِ مَا مَلِكَ بِأَيْمَانِكَ فَإِنْ تَعْنَى هَذِهِكَ  
صَرَاطًا سَوِيًّا إِنِّي يَا يَسَّارًا لَا تَعْبِدِ الشَّيْطَانَ إِذْ الشَّيْطَانُ كَانَ  
لِلرَّجُلِينَ عَصَمِيًّا إِنِّي يَا يَسَّارًا إِذْ أَخَافُ أَنْ يَسْكُنَكَ عَذَابٌ مِنْ  
الرَّجُلِينَ فَتَكُونُ لِلشَّيْطَانِ قَدِيرًا هَذَا أَخَافُ أَنْتَ عَنِ  
الْفَقِيرِ يَا يَسَّارًا هَذِهِ لِرَبِّكَ لَمْ تَنْتَهُ كَلْمَحَيْكَ وَأَهْجَزَنِي مَلِيًّا  
عَلَى سَلْمَةِ عَلَيْكَ سَاسَسَغَيْرِكَ رَبِّكَ إِنَّهُ كَانَ فِي حَفْيَيْكَ وَأَعْتَرَنِكَمْ  
وَفَاتَنِي عَوْنَ صَنْ دُوقَنِي الْمُنْوَادِحَوَارِيَّ فِي عَسْنَى الْأَذْلَوْنَ  
يُبَدِّلُ عَلَيَّ رَقْبَيْكَ تَشَقِّيَّكَ إِذْ كَانَ اعْتَزَلَهُمْ وَمَا يَعْبُدُونَ وَصَنْ  
دُوقَنِ الْمُنْوَادِ وَهَمْنَالَةِ اسْمَاعِيلَ وَبِعِقَوبَ وَكَلَّاجَعْلَنَاتِيَّ وَوَهَبَنَا  
لَهُمْ قَنْ رَحْمَنَنَا وَجَعَلَنَا الْمُهَمَّ لِسَانَ حَسَدَقَ عَلِيَّنَا إِذْ كَرَّ  
فِي الْكِتَابِ مُؤْنَسِي إِنِّي كَانَ مَعْلَمَصَا وَكَانَ رَسْوَالَكَجِيَّا إِذْ وَهَمْنَالَةِ  
قَادِيَّيَّهُ مِنْ بَحَارِبِ الظُّلُمُو الْأَدْعَمِنَ وَقَرْبَنَةِ تَجِيَّا وَوَهَمْنَالَةِ  
مِنْ رَحْمَنَنَا أَشَاءَهُ طَرْقَوْتِ تَجِيَّا وَإِذْ كَرَّ فِي الْكِتَابِ إِسْطَعِيلَ إِنِّي  
كَانَ صَلَادَقَ الْوَعِيدِ وَكَانَ رَسْوَالَكَجِيَّا وَكَانَ يَا مُرَاهَدَهُ  
يَا الْمَلْوَقَ وَالْمَلْوَقَ وَكَانَ عَنْدَ رَتِيَّهُ مَرْضِيَّا وَإِذْ كَرَّ فِي  
الْكِتَابِ إِدْرِيسَ إِنِّي كَانَ صَدِيقًا لِيَسَّارًا وَرَقْعَنَةِ مَكَانَاتِي عَلَيْكَ

أَوْلَيَّاتِ الْقَرْبَتِ أَنْكَهَ اللَّهَ بِكَبِيرِهِ مِنَ الْمُهَمَّنَ صَنْ دُوكَ دَمَرَ  
وَسَقَتْ كَحَلَنَا مَعْ نُوچَ وَمَنْ دُرِيَّ إِبْرَاهِيمَ وَانْرَأَيِّنَ وَمَنْ  
هَدَنَنَا وَاحْجَدَنَا إِذَا اتَّشَلَ عَلَيْهِمُ الْمُرْجَنَ خَرْنَوْ وَاسْجَدَ  
وَبِكِيَّا إِذْ كَلَفَهُنْ بِعِدَ وَهُرْ كَلَفَ أَصْسَاعُوا الصَّلَوةَ طَائِحَوَا  
الشَّكَوْتِ قَسْوَقَ يَلْقَوْنَ عَيَّاهُ الْأَمْنَ ثَابَتْ وَامْنَ وَعِوَلَ  
صَنَالْخَافَ وَلِيَكَ يُبَدِّلَ خَلَوْنَ الْجَنَّةَ وَلَا يَقْلَمَنَونَ شَيْئَانِيَّ بِعَذَنَ  
الْأَعْيَ وَعَدَ الرَّجُلِنَ عَمَّا دَأَدَهُ يَا عَيْبَ إِنِّي كَانَ وَعَدَ كَمَنْيَيَّهَا لَا  
يَسْمَعُونَ وَفِيهَا الْقَوَّالِسَلَّهَا وَلَهُمْ رَوْقَمَ وَفِيهَا بَكْرَهُ وَعَنْبَيَّا  
وَلِيَكَ الْجَنَّةَ الْقَوِّيَّ تَقْرِيَّثَ مِنْ عَيَّابَوَا أَصَنَ كَانَ تَقِيَّا وَمَا  
تَنْتَزَلَ الْأَيْمَرِيَّاتِ لِلْمَدَابِنَ إِبْرَاهِيمَ وَمَا لَخَفَنَا وَعَابَنَ ذَلِكَ  
وَمَا كَانَ رَئِيكَ تَسِيَّا إِذْ السَّكَوْتِ وَالْأَكْضِ وَعَابَنَهَا فَأَعْمَدَهُ  
وَاصْطَطَرَ لِعِيَادَتِهِ هَلْ تَقْلَمَلَهُ سَمِيَّا وَيَقْوَلُ الْأَنْسَانَ  
إِذَا مَاهَمَتْ لَتَسْوَقَ أَخْيَجَ حَيَّيَهُ أَوْلَادَهُ كَوَالِيَّانَ أَلَّا يَحْفَنَهُ  
مِنْ قَبْلَ وَلَهُ يَكَ شَيْئَانَ قَوْرَتَلَكَ لَتَخَشَرَهُمْ وَالشَّيْطَانَ  
لَتَخَشَرَهُمْ كَوَلَهُ كَوَلَهُ جَيْبَيَانَ لَمَّا لَتَبْرَعَنَ مِنْ كَلْلَ  
شَيْئَيَّهُ أَيْهُمْ أَشَدَّ عَلَى الرَّجُلِنَ شَيْئَانَ لَمَّا لَتَبْرَعَنَ أَعْلَمَ بِالْذَّوَافِ  
هَذَا قَلَ بِهَا صَلَيَّا وَلَنْ شَكَمَ الْأَفَارِدَهَا مَاهَنَ عَلَى دَوَافِهَا سَعَيَّا  
مَقْضَيَيَّاتِ لَمَّا لَتَجَسَّ الدَّرَبَنَ التَّقَوَا وَإِذْ الْمَلِيَّاتِ فِيهَا حَيَّيَّا

Untuk form3 dan form4, atau form berikutnya tidak memerlukan kode program. Dan berikut beberapa hasil eksekusi sistem.



# BAB 6

## SISTEM WAKTU NYATA PENGUJIAN MENGGUNAKAN TRANSFORMASI NON- SINUSOIDAL

### 6.1 Transformasi Hadamard

Sama seperti Transformasi Hadamard, Transformasi Hadamard memiliki sifat yang non-sinusoidal, atau dengan kata lain Transformasi Hadamard memiliki fungsi basis -1 atau 1.

Transformasi Hadamard pada sinyal  $f(x)$  dinyatakan sebagai :

$$H(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x)(-1)^{\sum_{i=0}^{n-1} b_i(x)b_i(u)}$$

dengan  $u=0, 1, 2, \dots, N-1$  dan  $x=0, 1, 2, \dots, N-1$ , dan nilai n mengikuti aturan  $N = 2^n$ .

Sebagai contoh jika  $N=8$  maka  $n=3$ .

$b_i(x)$  menyatakan bit ke i dari bentuk biner x, misal x=4 atau secara biner =100, maka  $b_0(x) = 0$ ,  $b_1(x) = 0$ , dan  $b_2(x) = 1$ . Demikian juga berlaku untuk  $b_i(u)$ .

Fungsi basis Transformasi Hadamard dinyatakan dengan formula berikut :

$$g(x, u) = (-1)^{\sum_{i=0}^{n-1} b_i(x)b_i(u)}$$

dan dari persamaan fungsi basis (kernel) Transformasi Hadamard untuk N=8 akan dihasilkan nilai-nilai kernel berikut :

x \ u	0	1	2	3	4	5	6	7
0	+	+	+	+	+	+	+	+
1	+	-	+	-	+	-	+	-
2	+	+	-	-	+	+	-	-
3	+	-	-	+	+	-	-	+
4	+	+	+	+	-	-	-	-
5	+	-	+	-	-	+	-	+
6	+	+	-	-	-	-	+	+
7	+	-	-	+	-	+	+	-

Untuk mendapatkan sebuah nilai pada kernel, maka dengan cara perhitungan berikut, misalkan ambil kasus  $u=1$  dan  $x=4$ , karena  $N=8$  maka  $n=3$ .

biner  $u=1$  adalah 001

biner  $x=4$  adalah 100

maka

$$b_0(u) = 1, b_1(u) = 0, \text{ dan } b_2(u) = 0$$

$$b_0(x) = 0, b_1(x) = 0, \text{ dan } b_2(x) = 1$$

$$g(x, u) = (-1)^{\sum_{i=0}^{n-1} b_i(x)b_i(u)}$$

$$g(4,1) = (-1)^{\sum_{i=0}^{n-1} b_i(4)b_i(1)}$$

$$g(4,1) = (-1)^{(1)(0)}(-1)^{(0)(0)}(-1)^{(0)(1)} = (-1)^0 = 1$$

Kasus lain misalkan  $u=4$  dan  $x=6$ ,

biner  $u=4$  adalah 100

biner  $x=6$  adalah 110

maka

$$b_0(u) = 0, b_1(u) = 0, \text{ dan } b_2(u) = 1$$

$$b_0(x) = 0, b_1(x) = 1, \text{ dan } b_2(x) = 1$$

$$g(x,u) = (-1)^{\sum_{i=0}^{n-1} b_i(x)b_i(u)}$$

$$g(6,4) = (-1)^{\sum_{i=0}^{n-1} b_i(6)b_i(4)}$$

$$g(6,4) = (-1)^{(0)(0)}(-1)^{(1)(0)}(-1)^{(1)(1)} = (-1)^1 = -1$$

Hitung Transformasi Hadamard dari sinyal  $f(x)=1, 1, 1, 1, 5, 5, 5, 5$

Penyelesaian :

$$W(0) = \frac{1}{8}(1+1+1+1+5+5+5+5) = 3$$

$$W(1) = \frac{1}{8}(1-1+1-1+5-5+5-5) = 0$$

$$W(2) = \frac{1}{8}(1+1-1-1+5+5-5-5) = 0$$

$$W(3) = \frac{1}{8}(1-1-1+1+5-5-5+5) = 0$$

$$W(4) = \frac{1}{8}(1+1+1+1-5-5-5-5) = -2$$

$$W(5) = \frac{1}{8}(1 - 1 + 1 - 1 - 5 + 5 - 5 + 5) = 0$$

$$W(6) = \frac{1}{8}(1+1-1-1-5-5+5+5)=0$$

$$W(7) = \frac{1}{8}(1 - 1 - 1 + 1 - 5 + 5 + 5 - 5) = 0$$

maka Transformasi Hadamard dari sinyal  $f(x)=1, 1, 1, 1, 5, 5, 5, 5$  adalah  $W(u)=3, 0, 0, 0, -2, 0, 0, 0$ .

## 6.2 Transformasi Slant

Transformasi Slant dapat dinyatakan secara rekursif ke dalam bentuk matrik NxN berikut :

$$S_N = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & & & & \\ a_N & b_N & 0 & & & \\ & & -a_N & b_N & 0 & \\ & & & & & S_{N/2} \\ \hline 0 & I_{(N/2)-2} & 0 & I_{(N/2)-2} & & \\ \hline 0 & 1 & 0 & 0 & -1 & \\ -b_N & a_N & b_N & a_N & & 0 \\ \hline 0 & I_{(N/2)-2} & 0 & -I_{(N/2)-2} & & \\ & & & & & S_{N/2} \end{pmatrix}$$

Untuk Transformasi Slant orde 2 :

$$S_2 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

$$S_4 = \frac{1}{\sqrt{4}} \begin{bmatrix} 1 & 1 & 1 & 1 \\ \frac{3}{\sqrt{5}} & \frac{1}{\sqrt{5}} & \frac{-1}{\sqrt{5}} & \frac{-3}{\sqrt{5}} \\ 1 & -1 & -1 & 1 \\ \frac{1}{\sqrt{5}} & \frac{-3}{\sqrt{5}} & \frac{3}{\sqrt{5}} & \frac{-1}{\sqrt{5}} \end{bmatrix}.$$

Untuk menentukan matriks Transformasi Slant orde N, maka kita perlu mendefinisikan  $b_N$  dan  $a_N$ , adapun  $b_N$  dan  $a_N$  secara berturut-turut dihitung sebagai relasi rekursif berikut :

$$a_2 = 1$$

$$b_N = (1 + 4(a_{N/2})^2)^{1/2}$$

$$a_N = 2b_N a_{N/2}$$

atau

$$a_{2N} = \sqrt{\frac{3N^2}{4(N^2 - 1)}}$$

$$b_{2N} = \sqrt{\frac{N^2 - 1}{4(N^2 - 1)}}$$

$$S_N = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 & | & 0 & 0 & | & -1 & 0 & | & 0 & 0 \\ a_N & b_N & | & 0 & 0 & | & -a_N & b_N & | & 0 & 0 \\ \hline - & - & | & - & - & | & - & - & | & I_{(N/2)-2} & - \\ 0 & | & I_{(N/2)-2} & | & 0 & | & 0 & | & I_{(N/2)-2} & - \\ \hline - & - & | & - & - & | & - & - & | & - & - \\ 0 & 1 & | & 0 & 0 & | & 0 & -1 & | & 0 & 0 \\ -b_N & a_N & | & 0 & 0 & | & b_N & a_N & | & 0 & 0 \\ \hline - & - & | & - & - & | & - & - & | & -I_{(N/2)-2} & - \\ 0 & | & I_{(N/2)-2} & | & 0 & | & 0 & | & -I_{(N/2)-2} & - \\ \hline \end{bmatrix} \begin{bmatrix} S_{N/2} & | & 0 \\ & | & - \\ & | & - \\ & | & - \\ & | & - \\ & | & 0 & | & S_{N/2} \end{bmatrix}$$

$$S_8 = \frac{1}{\sqrt{8}} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & \times & 1/\sqrt{5} \\ 1 & -3 & 3 & -1 & 1 & -3 & 3 & -1 & \times & 1/\sqrt{5 \times 21} \\ 7 & -1 & -9 & -17 & 17 & 9 & 1 & -7 & \times & \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & \times & 1/\sqrt{21} \\ 7 & 5 & 3 & 1 & -1 & -3 & -5 & -7 & \times & \\ 1 & -3 & 3 & -1 & -1 & 3 & -3 & 1 & \times & 1/\sqrt{5} \\ 3 & 1 & -1 & -3 & -3 & -1 & 1 & 3 & \times & 1/\sqrt{5} \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & & \end{bmatrix}.$$

Contoh kasus, diketahui sinyal  $f(x) = 9, 7, 3, 5$ , maka hitung Transformasi Slant sinyal  $f(x)$ .

Jawab :

$$S(u) = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ \frac{3}{\sqrt{5}} & \frac{1}{\sqrt{5}} & \frac{-1}{\sqrt{5}} & \frac{-3}{\sqrt{5}} \\ 1 & -1 & -1 & 1 \\ \frac{1}{\sqrt{5}} & \frac{-3}{\sqrt{5}} & \frac{3}{\sqrt{5}} & \frac{-1}{\sqrt{5}} \end{bmatrix} \times \begin{bmatrix} 9 \\ 7 \\ 3 \\ 5 \end{bmatrix} = \begin{bmatrix} \frac{12}{\sqrt{5}} \\ \frac{8}{\sqrt{5}} \\ \frac{2}{\sqrt{5}} \\ -\frac{2}{\sqrt{5}} \end{bmatrix}$$

## **DAFTAR PUSTAKA**

Fadlisyah. 2007. *Computer Vision dan Pengolahan Citra*, Penerbit Andi. Yogyakarta.

Fadlisyah. 2008. *Pengolahan Citra*, Penerbit Graha Ilmu. Yogyakarta.

Fadlisyah dan Rizal. 2011. *Pemrograman Computer Vision pada Video*, Penerbit Graha Ilmu. Yogyakarta.

Fadlisyah, Bustami, dan Ikhwanus. 2013. *Pengolahan Suara*, Penerbit Graha Ilmu.Yogyakarta.

Fadlisyah. 2018. *Pengolahan Suara untuk DAK Qur'an*, Modul dan Bahan kuliah Universitas Malikussaleh.



## BIOGRAFI PENULIS



Lahir di Lhokseumawe Provinsi Aceh pada tanggal 28 Februari 1976, SD (Sekolah Dasar) pada tahun 1982 dan selesai pada tahun 1988, melanjutkan pendidikan ke Pasentren Bustanul Ulum yang berada di Desa Alue Pineng – Langsa pada tahun 1988 hingga selesai pada tahun 1991 dengan pendidikan MTSN No. 16 Langsa, kembali ke Lhokseumawe untuk melanjutkan pendidikan pada SMA Negeri Nomor 2 pada tahun 1991 dan selesai pada tahun 1994, kemudian berangkat menuju Kota Yogyakarta untuk melanjutkan Program Pendidikan Strata Satu (S1) di Jurusan Teknik Informatika Universitas Islam Indonesia pada tahun 1994 dan selesai pada tahun 1999 dengan menyandang gelar Sarjana Teknik (S.T) sambil menunggu pekerjaan yang tetap maka saya juga ikut mengajar di Universitas Ahmad Dahlan untuk waktu 1 tahun dan pada tahun 2001 kembali ke Kota Lhokseumawe untuk masuk menjadi Pegawai Negeri Sipil (PNS) sebagai Tenaga Pendidik (Dosen) di Universitas Malikussaleh yang baru saja di negerikan, jabatan pertama yang saya terima sebagai sekretaris LPPM, Ketua PSIK, Kepala UPT Pusat Komputer dan selanjutnya berangkat kuliah pada Program Strata Dua (S2) di Jurusan Teknik Informatika STMIK Eresha pada tahun 2011 dan selesai pada tahun 2014 dengan gelar Magister Komputer (M.Kom), pada saat itu di Universitas Malikussaleh menjabat sebagai Kepala UPT Perpustakaan dan melanjutkan pendidikan ke Program Doktor di Jurusan Ilmu Komputer Universitas Sumatera Utara pada tahun 2014 dan selesai pada tahun 2018 dengan menyandang gelar Doktor (Dr.), aktif dibeberapa organisasi baik yang berskala Nasional atau Internasional, aktif menulis Artikel diberbagai Seminar Nasional atau Internasional dan di Jurnal bereputasi (Scopus/WOS) dan sering memberikan Materi di berbagai Workshop atau Seminar.



PENGOLAHAN

---

# CITRA & SUARA

---



**SEFA BUMI PERSADA**  
Jl. Malikussaleh No. 3 Bayu - Aceh Utara  
email: [sefabumipersada@gmail.com](mailto:sefabumipersada@gmail.com)  
Telp. 085260363550

ISBN 978-623-7648-18-5

A standard linear barcode is positioned vertically on the right side of the page. It is used for book identification and tracking. Below the barcode, the ISBN number is repeated for reference.

9 786237 648185