# BAB IV IMPLEMENTASI

Pada bab ini akan dibahas mengenai implementasi dari perancangan sistem. Implementasi berupa kode sumber dengan bahasa pemrograman C# dengan menggunakan Kinect SDK. Berikut ini akan diperkenalkan lingkungan yang digunakan untuk mengimplementasikan perangkat lunak ini.



## 4.1 Lingkungan Pembangunan

Dalam membangun aplikasi ini digunakan beberapa perangkat pendukung, baik perangkat keras maupun perangkat lunak. Lingkungan pembangunan aplikasi ini dijelaskan sebagai berikut.

### 4.1.1 Lingkungan Pembangunan Perangkat Keras

Perangkat keras yang digunakan dalam pembuatan aplikasi ini adalah sebuah perangkat laptop dengan spesifikasi sebagai berikut:

* Prosesor Intel(R) Core(TM) i7-6700HQ CPU @ 2.60GHz
* Memori (RAM) 16,00 GB

Dengan tambahan perangkat lunak Kinect Sensor Xbox One.

### 4.1.2 Lingkungan Pembangunan Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan untuk membuat aplikasi ini sebaga berikut.

* Microsoft Visual Studio 2015
* Windows 10 64 bit sebagai sistem operasi
* Kinect SDK

## 4.2 Implementasi Antarmuka

Seperti yang telah dijelaskan pada Subbab 3.2.5, modul pengenalan bahasa isyarat yang akan dibuat hanya akan memiliki satu window utama yang sudah mencakup semua fungsionalitas perangkat lunak yang dibutuhkan. Tampilan antarmuka perangkat dapat dilihat pada Gambar 4.1. Sedangkan kode sumber untuk antarmuka perangkat lunak dapat dilihat pada Kode Sumber 4.1.



Gambar 4.1 Antarmuka Perangkat Lunak

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63 | <Window x:Class="KinectHandTracking.MainWindow" xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"  xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"  Title="Kinect 2 hand tracking" Height="738.5" Width="1200" Loaded="Window\_Loaded" Closed="Window\_Closed">  <Grid Margin="0,0,2,0">  <Border BorderBrush="Black" BorderThickness="1" HorizontalAlignment="Left" Height="129" Margin="398,561,0,0" VerticalAlignment="Top" Width="563"/>  <Border BorderBrush="Black" BorderThickness="1" HorizontalAlignment="Left" Height="171" Margin="976,335,0,0" VerticalAlignment="Top" Width="171"/>  <Border BorderBrush="Black" BorderThickness="1" HorizontalAlignment="Left" Height="133" Margin="966,26,0,0" VerticalAlignment="Top" Width="189"/>  <Border BorderBrush="Black" BorderThickness="1" HorizontalAlignment="Left" Height="65" Margin="966,186,0,0" VerticalAlignment="Top" Width="189"/>  <Border BorderBrush="Black" BorderThickness="1" HorizontalAlignment="Left" Height="333" Margin="966,300,0,0" VerticalAlignment="Top" Width="189"/>  <Border BorderBrush="Black" BorderThickness="1" HorizontalAlignment="Left" Height="72" Margin="973,546,0,0" VerticalAlignment="Top" Width="175"/>  <Viewbox Width="940" Height="700" Margin="10,10,0,0" HorizontalAlignment="Left" VerticalAlignment="Top">  <Grid>  <Image Name="camera" Width="1920" Height="1080" />  <Canvas Name="canvas" Width="1920" Height="1080" ClipToBounds="True" Margin="-10,10,10,-10" />  <Border BorderBrush="Black" BorderThickness="1" HorizontalAlignment="Left" Height="976" VerticalAlignment="Top" Width="382" Margin="758,104,0,0"/>  </Grid>  </Viewbox>  <Label Name="label3" Content="Testing Data" HorizontalAlignment="Left" Margin="969,186,0,0" VerticalAlignment="Top" Width="152" FontSize="16"/>  <Button Name="OneTestButton" Content="Start Testing" HorizontalAlignment="Left" Margin="975,222,0,0" VerticalAlignment="Top" Width="75" Click="OneTestButton\_Click"/>  <Label Name="label5" Content="Create New Gesture" HorizontalAlignment="Left" Margin="969,23,0,0" VerticalAlignment="Top" Width="152" FontSize="16"/>  <Label Name="label6" Content="Gesture Name:" HorizontalAlignment="Left" Margin="973,49,0,0" VerticalAlignment="Top" Width="115" FontSize="16"/>  <TextBox Name="fileName" HorizontalAlignment="Left" Height="28" Margin="979,84,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="146" FontSize="16"/>  <Button Name="createButton" Content="Create File" HorizontalAlignment="Left" Margin="979,131,0,0" VerticalAlignment="Top" Width="75" Click="createButton\_click"/>  <Label Name="ambil" Content="-" HorizontalAlignment="Left" Height="47" Margin="1069,120,62,0" VerticalAlignment="Top" Width="59" FontSize="24"/>  <Label Name="label13" Content="/15" HorizontalAlignment="Left" Height="47" Margin="1096,117,0,0" VerticalAlignment="Top" Width="59" FontSize="24"/>  <Label Name="label7" Content="Output Image" HorizontalAlignment="Left" Margin="969,299,0,0" VerticalAlignment="Top" Width="152" FontSize="16"/>  <Border HorizontalAlignment="Left" Height="171" Margin="976,336,0,0" VerticalAlignment="Top" Width="171" >  <Image x:Name="outputImage" Stretch="UniformToFill" Margin="0,-3,0.2,2.2"/>  </Border>  <Label Name="label8" Content="Output Text" HorizontalAlignment="Left" Margin="969,514,0,0" VerticalAlignment="Top" Width="152" FontSize="16"/>  <Label Name="outputText" Content="" HorizontalAlignment="Left" Margin="973,555,0,0" VerticalAlignment="Top" Height="48" Width="174" FontSize="26.667" RenderTransformOrigin="0.981,2.1"/>  <Label Name="label9" Content="Status:" HorizontalAlignment="Left" Margin="6,560,0,0" VerticalAlignment="Top" Width="100" FontSize="24" Height="47"/>  <Label Name="statusDetail" Content="Kinect Not Connected" HorizontalAlignment="Left" Height="47" Margin="82,560,0,0" VerticalAlignment="Top" Width="341" FontSize="24"/>  <Label Name="label10" Content="Left Hand:" HorizontalAlignment="Left" Margin="399,559,0,0" VerticalAlignment="Top" Width="119" FontSize="24" Height="47"/>  <Label Name="tblLeftHandState" Content="Not Detected" HorizontalAlignment="Left" Height="47" Margin="515,559,0,0" VerticalAlignment="Top" Width="187" FontSize="24"/>  <Label Name="labelxL" Content="X" HorizontalAlignment="Left" Height="42" Margin="404,606,0,0" VerticalAlignment="Top" FontSize="16"/>  <Label Name="labelyL" Content="Y" HorizontalAlignment="Left" Height="42" Margin="404,648,0,-11" VerticalAlignment="Top" FontSize="16"/>  <Label Name="xL" Content="-" HorizontalAlignment="Left" Height="42" Margin="428,606,0,0" VerticalAlignment="Top" FontSize="16"/>  <Label Name="yL" Content="-" HorizontalAlignment="Left" Height="42" Margin="428,641,0,0" VerticalAlignment="Top" FontSize="16"/>  <Label Name="label11" Content="Right Hand:" HorizontalAlignment="Left" Margin="671,559,0,0" VerticalAlignment="Top" Width="144" FontSize="24" Height="47"/>  <Label Name="tblRightHandState" Content="Not Detected" HorizontalAlignment="Left" Height="47" Margin="803,559,0,0" VerticalAlignment="Top" Width="187" FontSize="24"/>  <Label Name="labelxR" Content="X" HorizontalAlignment="Left" Height="42" Margin="671,606,0,0" VerticalAlignment="Top" FontSize="16"/>  <Label Name="labelyR" Content="Y" HorizontalAlignment="Left" Height="42" Margin="671,641,0,0" VerticalAlignment="Top" FontSize="16"/>  <Label Name="xR" Content="-" HorizontalAlignment="Left" Height="42" Margin="695,606,0,0" VerticalAlignment="Top" FontSize="16"/>  <Label Name="yR" Content="-" HorizontalAlignment="Left" Height="42" Margin="695,641,0,0" VerticalAlignment="Top" FontSize="16"/>  <Label Name="label12" Content="Frame:" HorizontalAlignment="Left" Margin="981,251,0,0" VerticalAlignment="Top" Width="144" FontSize="24" Height="47"/>  <Label Name="ambilData" Content="Done" HorizontalAlignment="Left" Height="47" Margin="1069,251,-66,0" VerticalAlignment="Top" Width="187" FontSize="24"/>  <Button Name="stopButton" Content="Stop" HorizontalAlignment="Left" Margin="981,663,0,0" VerticalAlignment="Top" Width="75" Click="stopButton\_Click"/>  </Grid>  </Window> |

Kode Sumber 4.1 Tampilan Perangkat Lunak

## 4.3 Implementasi Perangkat Lunak

Pada subbab ini akan dibahas mengenai implementasi perangkat lunak dari kasus penggunaan ke dalam baris kode. Dijelaskan juga dengan fungsi yang dibutuhkan untuk menunjang perangkat lunak ini agar dapat berjalan sebagaimana mestinya. Implementasi ini dilakukan menggunakan Microsoft Visual Studio 2015 dengan bahasa pemrograman C#.

### 4.3.1 Implementasi Pendeteksian *Skeleton* Pengguna

Untuk menjalankan perangkat lunak ini diperlukan bantuan untuk mendeteksi *skeleton* pengguna, sehingga dibutuhkan bantuan dari perangkat keras Kinect 2.0. Sebelum mendeteksi *skeleton,* Kinect 2.0 harus diintegrasikan dengan program terlebih dahulu.

Kode sumber proses integrasi Kinect 2.0 dapat dilakukan seperti pada Kode Sumber 4.2. Untuk melakukan ekstraksi fitur, perangkat lunak mendeteksi tubuh pengguna terlebih dahulu. Ketika tubuh pengguna sudah terdeteksi, perangkat lunak kemudian menggambarkan *skeleton* pengguna secara keseluruhan termasuk 11 *skeleton joints* yang akan digunakan dalam proses ekstraksi fitur statis dan dinamis serta menentukan posisi gerakan. Kode sumber untuk mendeteksi *skeleton* pengguna dapat dilihat pada Kode Sumber 4.3.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | \_sensor = KinectSensor.GetDefault();  if (\_sensor != null)  {  \_sensor.Open();  statusDetail.Content = "Idle";  \_reader = \_sensor.OpenMultiSourceFrameReader(FrameSourceTypes.Color | FrameSourceTypes.Depth | FrameSourceTypes.Infrared | FrameSourceTypes.Body);  \_reader.MultiSourceFrameArrived += Reader\_MultiSourceFrameArrived;  } |

Kode Sumber 4.2 Kode Sumber Integrasi Kinect

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75 | async void Reader\_MultiSourceFrameArrived(object sender, MultiSourceFrameArrivedEventArgs e)  {  var reference = e.FrameReference.AcquireFrame();  #region Acquire Frame Color  // Color  using (var frame = reference.ColorFrameReference.AcquireFrame())  {  if (frame != null)  {  camera.Source = frame.ToBitmap();  }  }  #endregion  using (var frame = reference.BodyFrameReference.AcquireFrame())  {  if (frame != null)  {  canvas.Children.Clear();  \_bodies = new Body[frame.BodyFrameSource.BodyCount];  frame.GetAndRefreshBodyData(\_bodies);  var csv = new StringBuilder();  string filePath = "E:\\Kuliah\\TUGAS AKHIR\\TABI\\Data\\DataSet\\dataori.csv";  string imagePath = "";  foreach (var body in \_bodies)  {  if (body != null)  {  if (body.IsTracked)  {  Joint handRight = body.Joints[JointType.HandRight];  Joint handLeft = body.Joints[JointType.HandLeft];  Joint wristRight = body.Joints[JointType.WristRight];  Joint wristLeft = body.Joints[JointType.WristLeft];  Joint elbowRight = body.Joints[JointType.ElbowRight];  Joint elbowLeft = body.Joints[JointType.ElbowLeft];  Joint shoulderRight = body.Joints[JointType.ShoulderRight];  Joint shoulderLeft = body.Joints[JointType.ShoulderLeft];  Joint spineMid = body.Joints[JointType.SpineMid];  Joint spineShoulder = body.Joints[JointType.SpineShoulder];  Joint neck = body.Joints[JointType.Neck];  foreach (Joint joint in body.Joints.Values)  {  if (joint.TrackingState == TrackingState.Tracked)  {  CameraSpacePoint jointPosition = joint.Position;  Point point = new Point();  ColorSpacePoint colorPoint = \_sensor.CoordinateMapper.MapCameraPointToColorSpace(jointPosition);  point.X = float.IsInfinity(colorPoint.X) ? 0 : colorPoint.X;  point.Y = float.IsInfinity(colorPoint.Y) ? 0 : colorPoint.Y;  Ellipse ellipse = new Ellipse  {  Fill = Brushes.Yellow,  Width = 30,  Height = 30  };  Canvas.SetLeft(ellipse, point.X - ellipse.Width / 2);  Canvas.SetTop(ellipse, point.Y - ellipse.Height / 2);  canvas.Children.Add(ellipse);  }  }  }  }  }  }  }  } |

Kode Sumber 4.3 Kode Sumber Deteksi *Skeleton* Pengguna

### 4.3.2 Implementasi Proses Ekstraksi Fitur

Pada Kode Sumber 4.4 dijelaskan mengenai proses ekstraksi fitur statis dan dinamis pada setiap data serta menentukan posisi gerakan tangan yang dilakukan oleh pengguna berdasarkan rancangan pada Subbab 3.2.6.1. Penjelasan kode sumber tersebut adalah sebagai berikut:

1. Variabel *i* menunjukkan data/*frame* ke-*i* yang sedang diolah.
2. Variabel *Ny* menunjukkan koordinat *y* *skeleton joints* N.
3. Variabel *SMy* menunjukkan koordinat *y* *skeleton joints* SM.
4. Variabel *SS* menunjukkan koordinat *skeleton joints* SS.
5. Variabel *SL* menunjukkan koordinat *skeleton joints* SL.
6. Variabel *EL* menunjukkan koordinat *skeleton joints* EL.
7. Variabel *WL* menunjukkan koordinat *skeleton joints* WL.
8. Variabel *HL* menunjukkan koordinat *skeleton joints* HL.
9. Variabel *SR* menunjukkan koordinat *skeleton joints* SR.
10. Variabel *ER* menunjukkan koordinat *skeleton joints* ER.
11. Variabel *WR* menunjukkan koordinat *skeleton joints* WR.
12. Variabel *HR* menunjukkan koordinat *skeleton joints* HR.
13. Variabel *tanganKanan* menunjukkan posisi tangan kanan.
14. Variabel *tanganKiri* menunjukkan posisi tangan kiri.
15. Variabel *deltaHLx* menunjukkan tangan kiri.
16. Variabel *deltaHLy* menunjukkan tangan kiri.
17. Variabel *alphaHL* menunjukkan tangan kiri.
18. Variabel *deltaHRx* menunjukkan tangan kanan.
19. Variabel *deltaHRy* menunjukkan tangan kanan.
20. Variabel *alphaHR* menunjukkan tangan kanan.
21. Variabel *kuantKiri* menunjukkan hasil kuantisasi fitur dinamis tangan kiri.
22. Variabel *kuantKanan* menunjukkan hasil kuantisasi fitur dinamis tangan kanan.
23. Variabel *SRER* menunjukkan vektor bahu menuju siku tangan kanan.
24. Variabel *ERWR* menunjukkan vektor siku menuju pergelangan tangan kanan.
25. Variabel *WRHR* menunjukkan vektor pergelangan menuju telapak tangan kanan.
26. Variabel *SLEL* menunjukkan vektor bahu menuju siku tangan kiri.
27. Variabel *ELWL* menunjukkan vektor siku menuju pergelangan tangan kiri.
28. Variabel *WLHL* menunjukkan vektor pergelangan menuju telapak tangan kiri.
29. Variabel *HRHL* menunjukkan vektor telapak tangan kanan menuju telapak tangan kiri.
30. Variabel *SSSRER* menunjukkan sudut pada bahu tangan kanan.
31. Variabel *SRERWR* menunjukkan sudut pada siku tangan kanan.
32. Variabel *ERWRHR* menunjukkan sudut pada pergelangan tangan kanan.
33. Variabel *SSSLEL* menunjukkan sudut pada bahu tangan kiri.
34. Variabel *SLELWL* menunjukkan sudut pada siku tangan kiri.
35. Variabel *ELWLHL* menunjukkan sudut pada pergelangan tangan kiri.
36. Variabel DisHRHL menunjukkan jarak antara telapak tangan kanan dan telapak tangan kiri.

Fitur statis dan dinamis serta posisi gerakan tangan digabungkan menjadi fitur data yang kemudian disimpan ke dalam sebuah berkas berekstensi .csv dengan pembagian fitur data berdasarkan Tabel 3.6. Sebelum disimpan, seluruh atribut yang ada di dalam fitur data diubah terlebih dahulu menjadi atribut tipe string. Kode sumber penyimpanan fitur data dapat dilihat pada Kode Sumber 4.5.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  108  109  110  111  112  113  114  115  116  117  118  119  120  121  123  124  125  126  127  128  129  130  131  132  133  134  135 | if (i < 40 && statusAmbil != 0)  {  if (i < 38)  {  ambilData.Content = (i + 1).ToString();  }  else ambilData.Content = "Done";  if (i == 1)  {  Ny = (neck.Position.Y);  SMy = (spineMid.Position.Y);  }  if (i == 20)  {  if (HL.Y > Ny) tanganKiri = "Kepala";  else if (HL.Y < SMy) tanganKiri = "Perut";  else tanganKiri = "Dada";  if (HR.Y > Ny) tanganKanan = "Kepala";  else if (HR.Y < SMy) tanganKanan = "Perut";  else tanganKanan = "Dada";  }  // tangan kanan  if (deltaHRx == 0 && deltaHRy == 0)  {  alphaHR = -1;  }  else if (deltaHRx >= 0 && deltaHRy >= 0)  {  alphaHR = (Math.Atan(deltaHRy / deltaHRx)) \* (180 / Math.PI);  }  else if (deltaHRx < 0)  {  alphaHR = (Math.Atan(deltaHRy / deltaHRx)) \* (180 / Math.PI) + 180;  }  else  {  alphaHR = (Math.Atan(deltaHRy / deltaHRx)) \* (180 / Math.PI) + 360;  }  // tangan kiri  if (deltaHLx == 0 && deltaHLy == 0)  {  alphaHL = -1;  }  else if (deltaHLx >= 0 && deltaHLy >= 0)  {  alphaHL = (Math.Atan(deltaHLy / deltaHLx)) \* (180 / Math.PI);  }  else if (deltaHRx < 0)  {  alphaHL = (Math.Atan(deltaHLy / deltaHLx)) \* (180 / Math.PI) + 180;  }  else  {  alphaHL = (Math.Atan(deltaHLy / deltaHLx)) \* (180 / Math.PI) + 360;  }  // tangan kanan  if (alphaHR >= 315) { kuantKanan[i] = 8; }  else if (alphaHR >= 270) { kuantKanan[i] = 7; }  else if (alphaHR >= 225) { kuantKanan[i] = 6; }  else if (alphaHR >= 180) { kuantKanan[i] = 5; }  else if (alphaHR >= 135) { kuantKanan[i] = 4; }  else if (alphaHR >= 90) { kuantKanan[i] = 3; }  else if (alphaHR >= 45) { kuantKanan[i] = 2; }  else if (alphaHR >= 0) { kuantKanan[i] = 1; }  else if (alphaHR < 0) { kuantKanan[i] = 0; }  // tangan kiri  if (alphaHL >= 315) { kuantKiri[i] = 8; }  else if (alphaHL >= 270) { kuantKiri[i] = 7; }  else if (alphaHL >= 225) { kuantKiri[i] = 6; }  else if (alphaHL >= 180) { kuantKiri[i] = 5; }  else if (alphaHL >= 135) { kuantKiri[i] = 4; }  else if (alphaHL >= 90) { kuantKiri[i] = 3; }  else if (alphaHL >= 45) { kuantKiri[i] = 2; }  else if (alphaHL >= 0) { kuantKiri[i] = 1; }  else if (alphaHL < 0) { kuantKiri[i] = 0; }  if (1 < i && i < 38)  {  SRER += ER - SR;  ERWR += WR - ER;  WRHR += HR - WR;  SLEL += EL - SL;  ELWL += WL - EL;  WLHL += HL - WL;  HRHL += HL - HR;  Vector v1, v2;  double res;  //SS-SR-ER  v1 = SS - SR;  v2 = ER - SR;  res = Vector.AngleBetween(v1, v2);  SSSRER += (double)res \* Math.PI / 180;  //SR-ER-WR  v1 = SR - ER;  v2 = WR - ER;  res = Vector.AngleBetween(v1, v2);  SRERWR += (double)res \* Math.PI / 180;  //ER-WR-HR  v1 = ER - WR;  v2 = HR - WR;  res = Vector.AngleBetween(v1, v2);  ERWRHR += (double)res \* Math.PI / 180;  //SS-SL-EL  v1 = SS - SL;  v2 = EL - SL;  res = Vector.AngleBetween(v1, v2);  SSSLEL += (double)res \* Math.PI / 180;  //SL-EL-WL  v1 = SL - EL;  v2 = WL - EL;  res = Vector.AngleBetween(v1, v2);  SLELWL += (double)res \* Math.PI / 180;  //EL-WL-HL  v1 = EL - WL;  v2 = HL - WL;  res = Vector.AngleBetween(v1, v2);  ELWLHL += (double)res \* Math.PI / 180;  //Distance HR - HL  DisHRHL += Math.Sqrt((HR.X - HL.X) \* (HR.X - HL.X) + (HR.Y - HL.Y) \* (HR.Y - HL.Y)); // sqrt from dotproduct = length  }  } |

Kode Sumber 4.4 Kode Sumber Ekstraksi Fitur Statis, Dinamis dan Menentukan Posisi Gerakan Tangan dari *Skeleton* Pengguna

|  |  |
| --- | --- |
| 1  2  3  4  5  6  8  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90 | var stringSRER = SRER.ToString();  var stringERWR = ERWR.ToString();  var stringWRHR = WRHR.ToString();  var stringSLEL = SLEL.ToString();  var stringELWL = ELWL.ToString();  var stringWLHL = WLHL.ToString();  var stringHRHL = HRHL.ToString();  var stringSSSRER = SSSRER.ToString();  var stringSRERWR = SRERWR.ToString();  var stringERWRHR = ERWRHR.ToString();  var stringSSSLEL = SSSLEL.ToString();  var stringSLELWL = SLELWL.ToString();  var stringELWLHL = ELWLHL.ToString();  var stingDisHRHL = DisHRHL.ToString();  var stringkuant4 = kuantKiri[4].ToString();  var stringkuant6 = kuantKiri[6].ToString();  var stringkuant8 = kuantKiri[8].ToString();  var stringkuant10 = kuantKiri[10].ToString();  var stringkuant12 = kuantKiri[12].ToString();  var stringkuant14 = kuantKiri[14].ToString();  var stringkuant16 = kuantKiri[16].ToString();  var stringkuant18 = kuantKiri[18].ToString();  var stringkuant20 = kuantKiri[20].ToString();  var stringkuant22 = kuantKiri[22].ToString();  var stringkuant24 = kuantKiri[24].ToString();  var stringkuant26 = kuantKiri[26].ToString();  var stringkuant28 = kuantKiri[28].ToString();  var stringkuant30 = kuantKiri[30].ToString();  var stringkuant32 = kuantKiri[32].ToString();  var stringkuant34 = kuantKiri[34].ToString();  var stringkuant36 = kuantKiri[36].ToString();  var stringkuant38 = kuantKiri[38].ToString();  var stringkuant43 = kuantKanan[4].ToString();  var stringkuant45 = kuantKanan[6].ToString();  var stringkuant47 = kuantKanan[8].ToString();  var stringkuant49 = kuantKanan[10].ToString();  var stringkuant51 = kuantKanan[12].ToString();  var stringkuant53 = kuantKanan[14].ToString();  var stringkuant55 = kuantKanan[16].ToString();  var stringkuant57 = kuantKanan[18].ToString();  var stringkuant59 = kuantKanan[20].ToString();  var stringkuant61 = kuantKanan[22].ToString();  var stringkuant63 = kuantKanan[24].ToString();  var stringkuant65 = kuantKanan[26].ToString();  var stringkuant67 = kuantKanan[28].ToString();  var stringkuant69 = kuantKanan[30].ToString();  var stringkuant71 = kuantKanan[32].ToString();  var stringkuant73 = kuantKanan[34].ToString();  var stringkuant75 = kuantKanan[36].ToString();  var stringkuant77 = kuantKanan[38].ToString();  var stringtangankiri = tanganKiri;  var stringtangankanan = tanganKanan;  var stringnamagerakan = namaGerakan;  if (statusAmbil == 1)  {  var newLine = string.Format("{0},{1},{2},{3},{4},{5},{6},{7},{8},{9},{10},{11},{12},{13},{14},{15},{16},{17},{18},{19},{20},{21},{22},{23},{24},{25},{26},{27},{28},{29},{30},{31},{32},{33},{34},{35},{36},{37},{38},{39},{40},{41},{42},{43},{44},{45},{46},{47},{48},{49},{50},{51},{52}",  stringkuant4, stringkuant6,  stringkuant8, stringkuant10,  stringkuant12, stringkuant14,  stringkuant16, stringkuant18,  stringkuant20, stringkuant22,  stringkuant24, stringkuant26,  stringkuant28, stringkuant30,  stringkuant32, stringkuant34,  stringkuant36, stringkuant38,  stringkuant43, stringkuant45,  stringkuant47, stringkuant49,  stringkuant51, stringkuant53,  stringkuant55, stringkuant57,  stringkuant59, stringkuant61,  stringkuant63, stringkuant65,  stringkuant67, stringkuant69,  stringkuant71, stringkuant73,  stringkuant75, stringkuant77,  stringtangankiri, stringtangankanan,  stringSRER, stringERWR, stringWRHR,  stringSLEL, stringELWL, stringWLHL,  stringHRHL,  stringSSSRER, stringSRERWR, stringERWRHR,  stringSSSLEL, stringSLELWL, stringELWLHL,  stingDisHRHL,  stringnamagerakan  );    csv.AppendLine(newLine);  } |

Kode Sumber 4.5 Kode Sumber Menyimpan Hasil Ekstraksi Fitur Statis, Dinamis dan Posisi Gerak Tangan dari *Skeleton* Pengguna

### 4.3.4 Implementasi Proses *Testing* Data

Hasil dari *training* data yang dapat berupa *tree* yang dipisahkan dalam 2 bagian, yaitu *tree* statis dan dinamis yangdilihat pada Gambar 3.8 dan Gambar 3.9. Model *tree* tersebut kemudian diimplentasikan ke dalam perangkat lunak seperti yang dapat dilihat pada Kode Sumber 4.6. Keluaran dari kode sumber tersebut berupa prediksi gerakan yang dilakukan oleh pengguna ketika melakukan proses *testing* baik itu dalam bentuk tulisan maupun dalam bentuk gambar.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  108  109  110  111  112  113  114  115  116  117  118  119  120  121  122  123  124  125  126  127  128  129  130  131  132  133  134  135  136  137  138  139  140  141  142  143  144  145  146  147  148  149  150  151  152  153  154  155  156  157  158  159  160  161  162  163  164  165  166  167  168  169  170  171  172  173  174  175  176  177  178  179  180  181  182  183  184  185  186  187  188  189  190  191  192  193  194  195  196  197  198  199  200  201  202  203  204  205  206  207  208  209  210  211  212  213  214  215  216  217  218  219  220  221  222  223  224  225  226  227  228  229  230  231  232  233  234  235  236  237  238  239  240  241  242  243  244  245  246  247  248  249  250  251  252  253  254  255  256  257  258  259  260  261  262  263  264  265  266  267  268  269  270  271  272  273  274  275  276  277  278  279  280  281  282  283  284  285  286  287  288  289  290  291  292  293  294  295  296  297  298  299  300  301  302  303  304  305  306  307  308  309  310  311  312  313  314  315  316  317  318  319  320  321  322  323  324  325  326  327  328  329  330  331  332  333  334  335  336  337  338  339  340  341  342  343  344  345  346  347  348  349  350  351  352  353  354  355  356  357  358  359  360  361  362  363  364 | else if (statusAmbil == 2)  {  // Statis or Dinamis  flag4 = 0;  for (j = 2; j < 39; j = j+2)  {  if (kuantKanan[j] == 0)  {  flag4++;  }  }  if (flag4 > 9)  {  // Statis  if (ELWL.X < 0.03)  {  if (SRER.X < 0.11)  {  if (SLEL.X < -0.02)  {  if (DisHRHL < 0.47)  {  outputText.Content = "Hamba";  }  else if (DisHRHL >= 0.47)  {  outputText.Content = "Jendral";  }  }  else if (SLEL.X >= -0.02)  {  outputText.Content = "Wadah";  }  }  else if (SRER.X >= 0.11)  {  if (SLEL.X < -0.04)  {  if (SRERWR < -0.84)  {  outputText.Content = "Hai";  }  else if (SRERWR >= -0.84)  {  outputText.Content = "Hormat";  }  }  else if (SLEL.X >= -0.04)  {  if (SRERWR < -0.84)  {  outputText.Content = "Hai";  }  else if (SRERWR >= -0.84)  {  outputText.Content = "Ketua";  }  }  }  }  else if (ELWL.X >= 0.03)  {  if (ERWR.Y < -0.03)  {  if (SLEL.X < -0.1)  {  outputText.Content = "Ada";  }  else if (SLEL.X >= -0.1)  {  outputText.Content = "Gang";  }  }  else if (ERWR.Y >= -0.03)  {  if (SRER.Y < -0.12)  {  outputText.Content = "Botol";  }  else if (SRER.Y >= -0.12)  {  outputText.Content = "Geledeg";  }  }  }  }  else if (flag4 <= 9)  {  // Dinamis  if (kuantKiri[20] < 6)  {  if (kuantKiri[24] < 5.5)  {  if (kuantKiri[34] < 0.5)  {  if (kuantKanan[18] < 2)  {  if (kuantKanan[8] < 0.5)  {  outputText.Content = "Samping";  }  else if (kuantKanan[8] >= 0.5)  {  outputText.Content = "Sempit";  }  }  else if (kuantKanan[18] >= 2)  {  if (kuantKanan[26] < 4.5)  {  if (kuantKanan[36] < 2)  {  if (kuantKanan[18] < 6.5)  {  outputText.Content = "Topeng";  }  else if (kuantKanan[18] >= 6.5)  {  outputText.Content = "Samping";  }  }  else if (kuantKanan[36] >= 2)  {  if (kuantKiri[32] < 0.5)  {  if (kuantKiri[18] < 0.5)  {  outputText.Content = "Topeng";  }  else if (kuantKiri[18] >= 0.5)  {  outputText.Content = "Bingung";  }  }  else if (kuantKiri[32] >= 0.5)  {  outputText.Content = "Bingung";  }  }  }  else if (kuantKanan[26] >= 4.5)  {  if (kuantKiri[16] < 3.5)  {  outputText.Content = "Sempit";  }  else if (kuantKiri[16] >= 3.5)  {  outputText.Content = "Bingkai";  }  }  }  }  else if (kuantKiri[34] >= 0.5)  {  if (tanganKiri == "Perut")  {  if (kuantKanan[26] < 4.5)  {  if (kuantKiri[30] < 0.5)  {  if (kuantKanan[32] < 2)  {  if (tanganKanan == "Kepala")  {  outputText.Content = "Topeng";  }  else if (tanganKanan == "Perut")  {  outputText.Content = "Samping";  }  else if (tanganKanan == "Dada")  {  outputText.Content = "Topeng";  }  }  else if (kuantKanan[32] >= 2)  {  outputText.Content = "Bingung";  }  }  else if (kuantKanan[30] >= 0.5)  {  if (kuantKanan[32] < 3)  {  outputText.Content = "Topeng";  }  else if (kuantKanan[32] >= 3)  {  outputText.Content = "Sempit";  }  }  }  else if (kuantKanan[26] >= 4.5)  {  if (kuantKiri[20] < 0.5)  {  if (kuantKiri[18] < 2)  {  outputText.Content = "Sempit";  }  else if (kuantKiri[18] >= 0.5)  {  outputText.Content = "Bingung";  }  }  else if (kuantKiri[20] >= 0.5)  {  outputText.Content = "Sempit";  }  }  }  else if (tanganKiri == "Dada")  {  outputText.Content = "Rujuk";  }  else if (tanganKiri == "Kepala")  {  outputText.Content = "Kijang";  }  }  }  else if (kuantKiri[24] >= 5.5)  {  if (kuantKiri[18] < 2.5)  {  if (kuantKanan[14] < 3)  {  outputText.Content = "Kijang";  }  else if (kuantKanan[14] >= 3)  {  outputText.Content = "Bola";  }  }  else if (kuantKanan[18] >= 2.5)  {  outputText.Content = "Badan";  }  }  }  else if (kuantKiri[20] >= 6)  {  if (kuantKanan[30] < 3)  {  if (kuantKanan[36] < 3)  {  if (kuantKanan[32] < 4.5)  {  if (kuantKanan[28] < 1.5)  {  if (kuantKanan[24] < 2)  {  outputText.Content = "Besar";  }  else if (kuantKanan[24] >= 2)  {  outputText.Content = "Kijang";  }  }  else if (kuantKanan[28] >= 1.5)  {  if (kuantKanan[6] < 1)  {  outputText.Content = "Kijang";  }  else if (kuantKanan[6] >= 1)  {  outputText.Content = "Besar";  }  }  }  else if (kuantKanan[32] >= 4.5)  {  outputText.Content = "Kijang";  }  }  else if (kuantKanan[36] >= 3)  {  if (kuantKanan[26] < 4)  {  outputText.Content = "Kijang";  }  else if (kuantKanan[26] >= 4)  {  if (kuantKiri[22] < 7.5)  {  outputText.Content = "Bingkai";  }  else if (kuantKiri[22] >= 7.5)  {  outputText.Content = "Badan";  }  }  }  }  else if (kuantKanan[30] >= 3)  {  if (kuantKiri[12] < 7.5)  {  if (kuantKanan[34] < 6.5)  {  if (kuantKiri[14] < 0.5)  {  outputText.Content = "Bingung";  }  else if (kuantKiri[14] >= 0.5)  {  if (kuantKiri[22] < 4)  {  outputText.Content = "Rujuk";  }  else if (kuantKiri[22] >= 4)  {  outputText.Content = "Bola";  }  }  }  else if (kuantKanan[34] >= 6.5)  {  if (kuantKanan[18] < 7.5)  {  outputText.Content = "Badan";  }  else if (kuantKanan[18] >= 7.5)  {  outputText.Content = "Bingung";  }  }  }  else if (kuantKiri[12] >= 7.5)  {  if (kuantKiri[14] < 5.5)  {  if (kuantKanan[20] < 7.5)  {  outputText.Content = "Badan";  }  else if (kuantKanan[20] >= 7.5)  {  outputText.Content = "Bola";  }  }  else if (kuantKiri[14] >= 5.5)  {  if (tanganKiri == "Perut")  {  outputText.Content = "Bola";  }  else if (tanganKiri == "Dada")  {  outputText.Content = "Bingkai";  }  else if (tanganKiri == "Kepala")  {  outputText.Content = "Kijang";  }  }  }  }  }  }  } |

Kode Sumber 4.6 Implementasi Decision Tree dari Hasil Klasifikasi

*[Halaman ini sengaja dikosongkan]*