

# IDENTIFIKASI REAL TIME GERAKAN TANGAN PADA LEAP MOTION MENGGUNAKAN METODE DECISION TREE (J48)

Arda Surya Editya<sup>1</sup>

<sup>1</sup>Teknik Komputer, <sup>2</sup>Politeknik NSC Surabaya  
<sup>1</sup>ardasurya@gmail.com

## Abstrak

Kemajuan teknologi saat ini membuat banyak bermunculan produk teknologi dimana sebagian besar dapat membantu pekerjaan manusia. Salah satu produk teknologi saat ini adalah *Leap Motion*. *Leap Motion* adalah sebuah produk teknologi yang memanfaatkan *image processing* untuk memudahkan pengguna untuk menjadikan laptop atau komputer kita menjadi *touch screen* namun dengan medium udara. Pada penelitian ini akan dilakukan pengujian membuat sebuah gerakan baru diluar gerakan dasar leapmotion untuk menciptakan sebuah kontrol baru secara *realtime* dengan menggunakan metode *decision tree*. Percobaan ini menggunakan 3 gerakan baru yakni menggenggam, menguncup dan menggunting setelah diujicobakan maka didapatkan hasil sebagai berikut. Untuk gerakan menggenggam dari 10 kali didapatkan presentase keberhasilan deteksi 90%, untuk gerakan menggunting didapatkan presentase keberhasilan 70%, dan gerakan menguncup didapatkan presentase keberhasilan 80%. Dari data diatas maka dapat ditarik kesimpulan bahwa metode *decision tree* sangat cocok digunakan untuk klasifikasi *realtime* pada *leap motion*.

**Keyword: Klasifikasi, Leap Motion, J48**

## 1. PENDAHULUAN

*Leap Motion (Hand Motion tracking)* merupakan istilah untuk perekaman gerakan tangan yang digunakan menjadi model digital dan merupakan perangkat tambahan yang dapat dihubungkan ke komputer dan kemudian dapat digunakan untuk menggantikan fungsi *mouse* maupun *keyboard*. Fungsi dari alat yang bernama *Leap Motion* ini, dapat membantu penggunanya mengendalikan atau menggantikan tugas *mouse* maupun *keyboard* pada komputer hanya dengan gerakan tangan dan jari. Bentuk *Leap Motion* ini berukuran kotak yang cukup kecil sehingga mudah dibawa kemanapun.

Cara kerja *Leap Motion* ini adalah pengguna menghubungkan perangkat ke komputer, lalu cukup meletakkannya di depan monitor atau di dekat komputer yang pasti jangan sampai diluar jangkauan kabelnya. Kemudian *Leap Motion* akan mendeteksi keberadaan tangan maupun jari yang selanjutnya pengguna dapat menggunakan gerakan tangan atau jari yang diinginkan serta gerakan kombinasi yang sudah diatur. Interpretasi visual dari gerakan tangan secara alami telah menjadi bagian penting untuk berinteraksi dengan mesin dalam kehidupan sehari hari, seperti contohnya popularnya penggunaan *mouse* dan *keyboard* sebagai alat bantu dalam mencapai kemudahan dalam berinteraksi dengan komputer. Namun pada masa sekarang, interaksi tersebut masih dianggap terbatas. Disisi lain perkembangan dan pemanfaatan teknologi untuk media *virtual* telah diwujudkan sebagai sarana informasi dan komunikasi sensor secara *realtime*.

Dengan fungsinya yang memiliki banyak manfaat maka peneliti tertarik untuk meneliti bagaimana *leap motion* dapat mengenali gerakan-gerakan baru yang diciptakan sehingga dengan adanya penelitian ini dapat dijadikan sebuah landasan dalam pembuatan gerakan baru menggunakan *leap motion*.

## 2. TINJAUAN PUSTAKA

Penelitian yang sedang dilaksanakan ini sebelumnya sudah pernah dilaksanakan oleh beberapa peneliti diantara lain <sup>[1]</sup> *Digital Image Processing*, yang merupakan dasar dari pemrosesan gambar digital. Lebih lanjut Yu H Wang [2] meneliti tentang klasifikasi video menjadi identifikasi objek sehingga dapat diklasifikasikan dengan tepat.

### 2.1 Leap Motion

*Leap Motion (Hand Motion tracking)* merupakan istilah untuk perekaman gerakan tangan yang digunakan menjadi model digital dan merupakan perangkat tambahan yang dapat dihubungkan ke komputer dan kemudian dapat digunakan untuk menggantikan fungsi *mouse* maupun *keyboard*. Fungsi dari alat yang bernama *Leap Motion* ini, dapat membantu penggunanya mengendalikan atau menggantikan tugas *mouse* maupun *keyboard* pada komputer hanya dengan gerakan tangan dan jari. Bentuk *Leap Motion* ini berukuran kotak yang cukup kecil sehingga mudah dibawa kemanapun. Cara kerja *Leap Motion* ini adalah pengguna menghubungkan perangkat ke komputer, lalu cukup meletakkannya di depan monitor atau di dekat komputer yang pasti jangan sampai diluar jangkauan kabelnya. Kemudian *Leap Motion* akan mendeteksi keberadaan tangan maupun jari yang selanjutnya pengguna dapat menggunakan gerakan tangan atau jari yang diinginkan serta gerakan kombinasi yang sudah diatur.

### 2.2 Motion Tracking

*Motion Tracking* adalah istilah yang digunakan untuk mendeskripsikan perekaman gerakan dan pengertian gerakan tersebut menjadi model digital. *Motion tracking* disimulasi sebagai alat analisis *photogrammetric* dalam penelitian *biomechanics* pada tahun 1970-an dan 1980-an, serta meluas ke ranah edukasi, latihan, olahraga, dan baru

saja ke ranah animasi komputer untuk televisi, sinema, dan *video games*. Istilah *Hand Motion Tracking* merupakan salah satu istilah untuk salah satu implementasi teknik *tracking* dan *motion*, dimana proses *tracking* dilakukan pada objek tangan manusia, *Hand tracking* dapat di implementasikan dalam banyak hal dan bidang, seperti dapat digunakan untuk mengenal bahasa isyarat, dan dapat dipakai menjadi teknik interaksi antara manusia dan komputer berbasis visi dan lain sebagainya. Tangan manusia adalah sebuah struktur mekanik kompleks yang terdiri atas beberapa segmen tulang, ligame-ligamen yang menghubungkan antar segmen tulang secara leluasa, otot-otot yang berperan sebagai motor gerak, tendon yang berperan untuk menghubungkan otot dengan tulang, dan kulit serta saraf-saraf halus yang menyelubungi otot dan tulang. Tulang-tulang saling terhubung pada persendian dan tidak berubah ukurannya. Otot-otot menghasilkan tenaga penggerak dan menggerakkan sendi-sendi. Berdasarkan tipe gerakan dan rotasi yang mungkin dilakukan, sendi-sendi pada tangan manusia dapat diklasifikasikan sebagai *flexion*, *twist*, *directive*, atau *spherical*. Contoh sendi bertipe *flexion* dengan 1 DOF adalah lutut dan siku, sedangkan contoh sendi bertipe *twist* dengan 1 DOF adalah sendi pronation dari lengan bawah. Gerakan direktif dari sendi dengan 2 DOF menghasilkan gerakan *flexion* dengan arah lebih dari 2. Sendi *spherical*, sebagaimana pada sendi bahu, memiliki 3 DOF dan dapat melakukan gerakan *directive* dan *twist* secara simultan.

### 2.3 Gesture Understanding

Interaksi dan komunikasi Manusia dengan mesin melalui penggunaan *gesture* dapat menjadi salah satu cara untuk meningkatkan kenyamanan dalam proses interaksi antara manusia dengan mesin, terlebih lagi bila model interaksi yang disajikan berbasis ruang 3 dimensi. *Gesture understanding* untuk model gerakan tangan dapat dikelompokkan menjadi 2 definisi umum, yang pertama gerakan *gesture* (tangan) sebagai representasi pergerakan tokoh dalam permainan, kedua adalah *gesture* sebagai pengontrol dan pengguna terhadap sebuah aplikasi / sistem. Sebagai contohnya ketika seseorang ingin memulai sebuah sistem atau aplikasi dengan menggunakan *mouse* atau *keyboard*, proses ini dapat digantikan dalam gerakan *gesture* yang memberikan arti yang sama, salah satu contohnya adalah penggunaan *Tapgesture*, *swipe gesture* dan lain sebagainya. sehingga hal ini tergantung pada definisi yang diterapkan pada sebuah sistem. Model Interaksi *gesture* dapat dibagi menjadi dua kelompok, yakni,

- Model interaksi *gesture* berdasarkan pola bentuk *gesture*, dimana sistem bekerja dengan menangkap gambar dan mengenali *gesture* dari identifikasi bentuk, salah satu contohnya pada *gesture* berdasarkan pola tangan dimana pendeteksian gerakan tangan dengan pola tangan yang ditangkap oleh *leap motion controller* sesuai dengan pola gerakan tangan pengguna
- Pola gerakan *gesture*, dimana sebuah *gesture* ditangkap dengan membaca sinyal yang dibangkitkan

dari sensor yang telah terpasang, salah satu contohnya dideskripsikan pada *gesture* berdasarkan gerakan dimana gerakan tangan pengguna dibaca berdasarkan arah dan pola gerakannya.

### 2.3 Software WEKA

WEKA adalah sebuah software yang dikeluarkan oleh Waikato University. Software ini berfungsi sebagai menerapkan metode *artificial intelligence* untuk mengolah data dalam jumlah besar.

### 3. METODE PENELITIAN

Kami mengambil 3 jenis gerakan untuk dijadikan sebagai dataset yakni gerakan menggenggam, menguncup dan menepuk. Seperti ditunjukkan pada gambar dibawah :



Gambar 1. Genggam

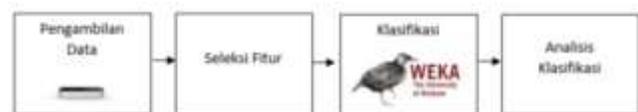


Gambar 2. Kuncup



Gambar 3. Gunting

Skema pengambilan data gerak dapat digambarkan pada skema dibawah ini :



Pengambilan data dilakukan menggunakan *leap motion* dimana dapat merekam pergerakan tangan dengan *tool* yang sudah disediakan oleh *leap motion developer* sehingga didapatkan data x,y,z dari setiap *gesture*. Data ini

akan dilabeli terlebih dahulu yang kemudian akan diolah weka.

Data yang diambil merupakan data 3 gerakan yang diambil oleh 2 orang yakni arda dan soffa setelah data yang diperoleh dari 3 gerakan dari 2 orang tersebut dijadikan satu untuk dijadikan dataset. Proses penggabungan dan pelabelan ini menggunakan software microsoft excel.



Gambar 4. Data Mentah

Setelah melalui pembentukan dataset dan pelabelan maka berikutnya data akan diolah menggunakan *machine learning* dengan metode *Decision Tree* (j48).

### 3. HASIL DAN PEMBAHASAN

Setelah kita mengadakan ujicoba maka dapat didapatkan hasil sebagai berikut :

Untuk deteksi dataset responden pertama didapatkan hasil sebagai berikut :

```

Time taken to build model: 0.32 seconds

=== Evaluation on test set ===
=== Summary ===

Correctly Classified Instances      1917      43.2048 %
Incorrectly Classified Instances    2520      56.7951 %
Kappa statistic                    0.1494
Mean absolute error                0.3704
Root mean squared error            0.4153
Relative absolute error            55.0835 %
Root relative squared error        130.0512 %
Total Number of Instances         4437

=== Detailed Accuracy By Class ===

      TP Rate  FP Rate  Precision  Recall  F-Measure  ROC Area  Class
      0.228   0.003   0.978   0.228   0.349   0.412  genggan
      0.991   0.804   0.358   0.991   0.524   0.593  kuncup
      0.129   0.019   0.774   0.129   0.221   0.555  gunting
Weighted Avg.  0.432   0.258   0.716   0.432   0.348   0.587

=== Confusion Matrix ===

  a  b  c  <-- classified as
356 1160 47 | a = genggan
 4 1369  9 | b = kuncup
 4 1296 192 | c = gunting
    
```

Untuk deteksi dataset responden 2 didapatkan hasil sebagai berikut :

```

Time taken to build model: 0.32 seconds

=== Evaluation on test set ===
=== Summary ===

Correctly Classified Instances      1917      43.2048 %
Incorrectly Classified Instances    2520      56.7951 %
Kappa statistic                    0.1494
Mean absolute error                0.3704
Root mean squared error            0.4153
Relative absolute error            55.0835 %
Root relative squared error        130.0512 %
Total Number of Instances         4437

=== Detailed Accuracy By Class ===

      TP Rate  FP Rate  Precision  Recall  F-Measure  ROC Area  Class
      0.228   0.003   0.978   0.228   0.349   0.412  genggan
      0.991   0.804   0.358   0.991   0.524   0.593  kuncup
      0.129   0.019   0.774   0.129   0.221   0.555  gunting
Weighted Avg.  0.432   0.258   0.716   0.432   0.348   0.587

=== Confusion Matrix ===

  a  b  c  <-- classified as
356 1160 47 | a = genggan
 4 1369  9 | b = kuncup
 4 1296 192 | c = gunting
    
```

Untuk Deteksi gabungan kedua dataset didapatkan hasil sebagai berikut :

```

Time taken to build model: 0.23 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances      9050      97.7958 %
Incorrectly Classified Instances    204       2.2048 %
Kappa statistic                    0.9669
Mean absolute error                0.0147
Root mean squared error            0.1212
Relative absolute error            3.3099 %
Root relative squared error        25.7288 %
Total Number of Instances         9254

=== Detailed Accuracy By Class ===

      TP Rate  FP Rate  Precision  Recall  F-Measure  ROC Area  Class
      0.974   0.011   0.974   0.974   0.975   0.961  genggan
      0.974   0.011   0.977   0.974   0.975   0.961  kuncup
      0.985   0.011   0.981   0.985   0.983   0.997  gunting
Weighted Avg.  0.976   0.011   0.978   0.978   0.979   0.993

=== Confusion Matrix ===

  a  b  c  <-- classified as
2824  43  34 | a = genggan
 49 2941  30 | b = kuncup
 21  27 3228 | c = gunting
    
```

Untuk deteksi berikutnya gabungan data 2 responden dengan metode data split 66% didapatkan hasil sebagai berikut:

```

Time taken to build model: 0.25 seconds

--- Evaluation on test-split ---
--- Summary ---

Currently Classified Instances      3067      97.488%
Incorrectly Classified Instances    79        2.511%
Mappa statistic                    0.9623
Mean absolute error                 0.0167
Root mean squared error             0.1294
Relative absolute error             1.7705%
Root relative squared error         27.443%
Total Number of Instances          3146

--- Detailed Accuracy By Class ---

      TP Rate  FP Rate  Precision  Recall  F-Measure  ROC Area  Class
      -----  -----  -
      0.975    0.011    0.977    0.975    0.976    0.982    gonggan
      0.968    0.016    0.967    0.968    0.967    0.978    kuncup
      0.981    0.011    0.98    0.981    0.981    0.985    gantung
Weighted Avg.  0.975    0.012    0.975    0.975    0.975    0.981

--- Confusion Matrix ---

  a  b  c  c-- classified as
  377 17  8 |  a = gonggan
  19 891 14 |  b = kuncup
  4  17 1089 |  c = gantung
    
```

#### 4. KESIMPULAN DAN SARAN

Berdasarkan penelitian diatas maka dapat disimpulkan sebagai berikut :

1. Untuk ujicoba dengan skenario 1 menunjukkan 43,20% data yang berhasil diidentifikasi.
2. Untuk ujicoba dengan skenario 2 menunjukkan 97,79% data yang berhasil diidentifikasi.
3. Untuk ujicoba dengan skenario 3 dan ke 4 menunjukkan 97,48% data yang berhasil diidentifikasi.

#### 5. REFERENSI

- Rafael C. Gonzales and Richard E. Woods, "Digital Image Processing," Prentice Hall Press, 2002, ISBN 0-201-18075-8.
- Yu, H., Wang, J., Huang, Z., Yang, Y., & Xu, W. (2016). Video paragraph captioning using hierarchical recurrent neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 4584-4593).