

# PENDETEKSIAN GERAK TANGAN MANUSIA SEBAGAI INPUT PADA KOMPUTER

Wikaria Gazali<sup>1</sup>; Haryono Soeparno<sup>2</sup>

<sup>1</sup> Jurusan Matematika, Fakultas Sains dan Teknologi, Universitas Bina Nusantara  
Jln. K.H. Syahdan No. 9, Palmerah, Jakarta Barat 11480  
[wikaria@binus.edu](mailto:wikaria@binus.edu)

<sup>2</sup> Jurusan Teknik Informatika, Fakultas Ilmu Komputer, Universitas Bina Nusantara,  
Jl. K.H. Syahdan No. 9, Palmerah, Jakarta Barat 11480  
[haryono@binus.edu](mailto:haryono@binus.edu)

## ABSTRACT

*Hand Recognizer is one of the results of science and technology development. Recognition capability which is used to be difficult to do by a computer engine is not impossible any more today. This research aims to produce a hand recognizer application program that can recognize human's hand (movement) and recast it in numbers or called Motion Detection. The process of introduction and training conducted in this research uses Image Processing and Backpropagation. The application program designed in this research uses new hand shapes in the form of numbers as the training data which is able to recognize back the hands moved in accordance to the expected targets and the level of accuracy depending on the quality of the in-training data and length of training. Training can be continued until the introduction of hands or the application program achieves the required level of accuracy.*

**Keywords:** *hands recognizer, motion detection, histogram, backpropagation, artificial neural network*

## ABSTRAK

*Pengenalan tangan (Hands Recognizer) merupakan salah satu hasil dari perkembangan ilmu pengetahuan dan teknologi. Kemampuan pengenalan tangan yang dahulu terasa sulit untuk dilakukan oleh suatu mesin komputer sekarang bukanlah lagi hal yang mustahil. Makalah ini bertujuan untuk menghasilkan sebuah program aplikasi Hands Recognizer yang dapat mengenali tangan (gerakan) yang dilakukan manusia dan menampilkannya kembali ke dalam bentuk angka atau bisa juga disebut dengan Motion Detection. Proses pengenalan maupun training yang dilakukan pada makalah ini menggunakan Image Processing dan Backpropagation. Program aplikasi yang dirancang dalam makalah ini menggunakan bentuk-bentuk tangan berupa angka sebagai data training dan mampu mengenali kembali tangan-tangan baru yang digerakkan sesuai target yang diharapkan dengan tingkat akurasi yang tergantung pada kualitas data yang di-training dan lamanya training. Training dapat terus dilanjutkan sampai pengenalan tangan atau program aplikasi mencapai tingkat akurasi yang diinginkan.*

**Kata kunci:** *pengenalan tangan, motion detection, histogram, backpropagation, jaringan saraf tiruan*

## PENDAHULUAN

Kecanggihan teknologi sekarang ini berkembang dengan begitu pesatnya sehingga saat ini pengguna (user) bisa berinteraksi dengan komputer melalui cara-cara yang semakin revolusioner. Sekarang bukan zamannya lagi pengguna hanya dapat berinteraksi dengan komputer menggunakan modus teks. Sebuah komputer pribadi bukan lagi sebuah mesin yang kaku dan menyusahkan penggunanya untuk menyesuaikan diri dan mempelajarinya. Bila diamati dengan seksama maka akan terlihat perkembangan teknologi komputer terkini yang sudah mengarah bukan hanya pada kemudahan pemakaian komputer, namun juga disertai dengan efek-efek menarik yang membuat pengguna komputer merasa terhibur ketika mengoperasikannya.

Namun sayangnya sampai sekarang pemakalah tidak banyak menemukan sebuah aplikasi yang dapat mengenali tangan manusia yang dapat digunakan untuk mengendalikan sebuah komputer. Maka dari itu pemakalah mencoba membuat aplikasi tersebut yang dapat mengenali tangan manusia dengan menggunakan sebuah webcam (sebuah kamera yang banyak digunakan oleh pengguna komputer untuk mengambil video).

Perlu diadakan banyak penelitian dan pengembangan untuk mewujudkan suatu sistem *hands recognition* (pengenal tangan) ini. Makalah ini merupakan salah satu alat yang diharapkan dapat berkontribusi terhadap kemajuan teknologi di Indonesia umumnya dan teknologi pengenalan tangan pada khususnya.

Penelitian ini dibuat untuk menciptakan sebuah program aplikasi yang dapat digunakan untuk mengenali pergerakan tangan yang dilakukan oleh manusia, yang berupa simbol (misalnya angka satu, dua, tiga, dan seterusnya). Komputer diharapkan dapat memproses gerakan tersebut sebagai input untuk proses melakukan pekerjaan tertentu yang sudah diprogram sebelumnya.

Selain itu diharapkan juga bahwa program aplikasi ini akan membuka peluang bagi bermunculannya pengembangan yang lebih lanjut dalam penelitian yang lebih dalam mengenai masalah ini, terutama pada gerakan tangan yang lebih rumit.

Manfaat dari penelitian ini yaitu perkembangan lebih lanjut dari penelitian ini dapat membantu dalam pengendalian komputer, memungkinkan komputer dapat mengenali perintah, memberikan kontribusi dalam bidang robotik, maupun sebagai pelengkap sistem komputer rumah masa depan. Bila aplikasi *Hands Recognizer* ini semakin disempurnakan, akan banyak sekali bidang yang akan merasakan kontribusinya secara signifikan. Kehidupan manusia akan menjadi lebih mudah dan inovatif.

### Algoritma Back Propagation

Dalam algoritma *Backpropagation* terdapat proses *training* yang meliputi *feedforward*, *backward* serta perbaikan bobot dan bias. Sedangkan proses selanjutnya untuk mengenali pola yang telah dilatih, akan disebut dengan proses *recall* yang dilakukan hanya dengan mengulangi proses *feedforward*.

Berikut ini merupakan langkah-langkah yang digunakan dalam proses *training* dan *recall* menggunakan algoritma *Backpropagation*:

1. Inisialisasi bobot
2. Selama kondisi berhenti *false*, lakukan langkah 3 – 10
3. Untuk setiap pasangan data pelatihan, lakukan langkah 4 – 9

### Feed Forward (langkah maju)

4. Setiap *unit input* ( $X_i$ ,  $i = 1, \dots, n$ ) menerima sinyal *input*  $x_i$  dan meneruskan sinyal tersebut ke semua unit pada lapis di atasnya (*hidden layer*).
5. Setiap *unit tersembunyi* ( $Z_j$ ,  $j = 1, \dots, p$ ) menjumlahkan sinyal-sinyal *input* berbobot,

$$z\_in_j = v_{0j} + \sum_{i=1}^n x_i v_{ij}$$

Gunakan fungsi aktivasi untuk menghitung sinyal *output*-nya,  $z_j = f(z\_in_j)$

Lalu kirimkan sinyal tersebut ke semua *unit* di lapis atasnya (*unit-unit output*).

6. Setiap *unit output* ( $Y_k$ ,  $k = 1, \dots, m$ ) menjumlahkan sinyal-sinyal *input* berbobot,

$$y\_in_k = w_{0k} + \sum_{j=1}^p z_j w_{jk}$$

Gunakan fungsi aktivasi untuk menghitung sinyal *output*-nya,  $y_k = f(y\_in_k)$

### Backward (langkah mundur)

7. Tiap-tiap *unit output* ( $Y_k$ ,  $k = 1, \dots, m$ ) menerima pola target yang berhubungan dengan pola input pembelajaran, hitung informasi *error*-nya,  $\delta_k = (t_k - y_k) f'(y\_in_k)$

Hitung koreksi bobot (yang nantinya akan digunakan untuk memperbaiki nilai  $w_{jk}$ ),

$$\Delta w_{jk} = \alpha \delta_k z_j$$

Hitung juga koreksi bias (yang nantinya akan digunakan untuk memperbaiki nilai  $w_{0k}$ ),

$$\Delta w_{0k} = \alpha \delta_k$$

Dan kirimkan nilai  $\delta_k$  ke *unit-unit* yang ada di lapis bawahnya.

8. Tiap-tiap *hidden unit* ( $z_j$ ,  $j = 1, \dots, p$ ) menjumlahkan delta *input*-nya (dari *unit* yang berada pada lapis atasnya),

$$\delta\_in_j = \sum_{k=1}^m \delta_k w_{jk}$$

Kalikan nilai ini dengan turunan dari fungsi aktivasinya untuk menghitung informasi *error*-nya,

$$\delta_j = \delta\_in_j f'(z\_in_j)$$

Kemudian hitung koreksi bobot (yang nantinya digunakan untuk memperbaiki nilai  $v_{ij}$ ),

$$\Delta v_{ij} = \alpha \delta_j x_i$$

Hitung juga koreksi bias (yang nantinya digunakan untuk memperbaiki nilai  $v_{0j}$ ),  $\Delta v_{0j} = \alpha \delta_j$

### Perbaikan Bobot dan Bias

9. Setiap *unit output* ( $y_k$ ,  $k = 1, \dots, m$ ) memperbaiki bobot dan biasnya ( $j = 0, \dots, p$ ),

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta w_{jk}$$

Setiap *unit tersembunyi* ( $z_j$ ,  $j = 1, \dots, p$ ) memperbaiki bobot dan biasnya ( $i = 0, \dots, n$ ),

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij}$$

10. Tes kondisi berhenti.

Setelah proses *training*, sebuah jaringan syaraf *backpropagation* akan diaplikasikan dengan hanya menggunakan *fase feedforward* dari algoritma *training* saja. Prosedur aplikasi tersebut adalah sebagai berikut:

Inisialisasi matriks bobot (dari algoritma *training*).

Untuk setiap vektor *input*, lakukan langkah 3-5.

Untuk  $i = 1, \dots, n$ : set nilai aktivasi untuk setiap *input unit*  $x$ .

Untuk  $j = 1, \dots, p$ :

$$z_{in_j} = v_{0j} + \sum_{i=1}^n x_i v_{ij}$$

$$z_j = f(z_{in_j})$$

Untuk  $k = 1, \dots, m$ :

$$y_{in_k} = w_{0k} + \sum_{j=1}^p z_j w_{jk}$$

$$y_k = f(y_{in_k})$$

Jadi, pada landasan teori ini, telah dibahas teori-teori yang berhubungan dengan perancangan program aplikasi, seperti metode perancangan program, *image processing*, histogram, *Artificial Neural Networks*, dan algoritma *Back Propagation*.

### Analisis Sistem Pengenalan Tangan

Untuk dapat memasuki proses *training*, gambar tangan yang menunjukkan angka-angka tersebut harus diproses terlebih dahulu menggunakan *image processing*. Adapun langkah-langkah dalam tahap analisis pemrosesan tangan terbagi lagi menjadi:

- Mengambil gambar tangan dari video yang dihasilkan oleh Webcam  
 Dari video yang dihasilkan oleh webcam, diambil gambar setiap satu detik untuk diproses. Webcam yang digunakan dalam makalah ini menghasilkan gambar 24 bit per-pixel berukuran 320x200 pixel. Dalam makalah ini tangan yang mau dikenali haruslah tangan kanan, dan harus diletakkan di suatu posisi sedemikian rupa sehingga terletak di dalam kotak yang telah ditandakan dari video yang dihasilkan webcam tersebut. Seperti yang dapat dilihat pada gambar.



Gambar 1 Tampilan dari Video yang Dihasilkan oleh Webcam

Jadi gambar yang akan diproses kemudian adalah gambar yang berada di dalam kotak merah tersebut. Hal tersebut dilakukan supaya gambar yang dihasilkan jelas dan tidak terlalu banyak noise.

– Image Processing

Kemudian setelah gambar tersebut diambil dan dipotong sesuai kotak (100x100 pixel), gambar tangan tersebut diubah ke dalam bentuk *grayscale*, menghilangkan warna-warna, kemudian dengan menggunakan *threshold* mengubah gambar tersebut menjadi hitam dan putih.



Gambar 2 Tangan setelah Diubah Menjadi Grayscale

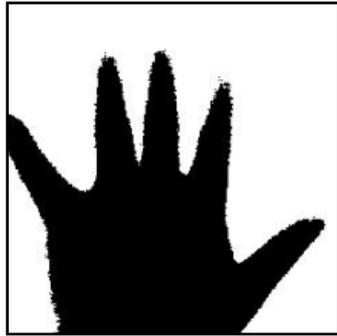


Gambar 3 Tangan setelah Noise Dihilangkan dengan Threshold

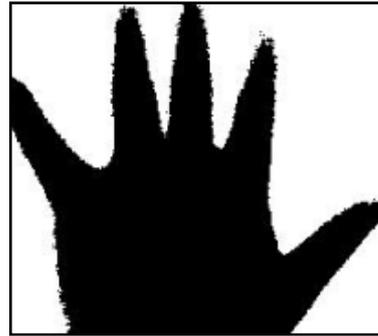
Nilai *threshold* yang digunakan adalah 128, biasanya digunakan nilai 128 karena berada pas di tengah, sehingga *pixel* yang bernilai kurang dari 128 diubah menjadi warna putih, dan bernilai sama atau lebih menjadi warna hitam, teknik ini digunakan terutama untuk menghilangkan noise yang ada di gambar tersebut.

– Pengolahan gambar menjadi input menggunakan histogram

Setelah gambar menjadi hitam-putih, maka gambar tersebut dipotong sebelah kiri, kanan, dan atas untuk menghilangkan bagian yang hanya berwarna putih, membentuk sebuah gambar kotak yang penuh dengan tangan user, dengan ukuran yang lebih kecil. Hal ini berguna agar nanti ketika dibuat perbandingan warna putih dengan hitam, besar dan jarak tangan dari kamera tidak akan mempengaruhi dalam pemrosesan menjadi input.

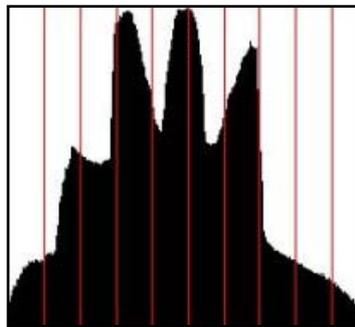


Gambar 4 Tangan yang Masih Memiliki Bagian Putih di Atas dan di Kanan



Gambar 5 Tangan yang sudah Dipotong Bagian Putihnya Membentuk Kotak

Kemudian gambar tangan yang sudah dipotong tersebut akan dihitung perbandingan hitam dengan bagian putih dengan menggunakan teknik histogram untuk mendapatkan nilai yang dapat dimasukkan ke dalam program sebagai input. Seperti yang ditunjukkan oleh gambar berikut:



Gambar 6 Tampilan Perbandingan Hitam dan Putih dalam 10 Bagian.

Cara pemrosesan gambar dari tangan hitam putih menjadi gambar di atas adalah dengan menghitung nilai hitam untuk setiap baris vertikal gambar tersebut, sehingga seakan-akan gambar tangan tersebut ditarik ke bawah (perbedaan yang mencolok terdapat pada bagian jempol dan kelingking). Setelah itu gambar tersebut dibagi menjadi 10, dan persentase hitam terhadap gambar di satu bagian itulah yang menjadi sebuah input yang akan dimasukkan ke dalam *Neural Network*. Sehingga total input yang dapat dimasukkan ke dalam proses training adalah 10 input. Pada contoh di atas, persentase hitam yang ada adalah 17%, 39%, 54%, 92%, 79%, 79%, 76%, 39%, 19%, dan 12%.



Gambar 7 Tampilan Bagian ke-10 dari Percobaan di Atas dengan 12% Hitam.

Demikianlah analisis sistem pengenalan tangan yang dilakukan dari sebuah gambar tangan, dan diproses sedemikian rupa sehingga menjadi input-input yang akan menjadi bagian dari proses training.

## **Sistem Training**

Pada tahap training ini *Neural Network* akan dilatih menggunakan *Backpropagation* sebagai alat training *Neural Network* dengan menggunakan *input layer* yang berjumlah 10 node, *hidden layer* 7 node dan *output layer* 3 node. Jumlah node tersebut ditentukan karena setelah dilakukan beberapa kali percobaan dan didapat bahwa nilai tersebut mampu menghasilkan *Neural Network* dengan akurasi yang lebih tepat dengan waktu training yang tidak terlalu lama.

Sebelum di-training, data gambar diproses seperti yang telah dilakukan pada analisis sistem pemrosesan gambar untuk mendapatkan gambar yang menjadi data untuk di-training. Tahap training ini memerlukan waktu yang cukup lama bila data yang di-training cukup besar dan banyak.

## **Sistem Pengenalan**

Tahap pengenalan inilah yang merupakan inti dari program aplikasi ini. Apabila proses training sudah selesai dilakukan, maka *neural network* yang merupakan hasil dari proses training dapat digunakan untuk mengenali gambar yang diberikan oleh pengguna program aplikasi. Sebelum dikenali, data gambar diproses seperti yang telah dilakukan pada analisis sistem pemrosesan untuk mendapatkan gambar yang akan dikenali pola kemiripannya oleh algoritma *Backpropagation* pada saat gambar yang baru ingin dikenali.

Dalam tahap pengenalan ini dilakukan proses *feedforward* pada *Backpropagation* untuk mendapatkan gambar-gambar yang telah direkam dalam bentuk angka. Hasil dari *feedforward* merupakan tiga digit bilangan biner yang akan dikonversi menjadi satu bilangan desimal. Hasil pengenalan yang berupa serangkaian kata-kata berurutan akan ditampilkan sebagai output program. Contoh dalam training ini akan menghasilkan 3 nilai output yang berupa konversi nilai 1-5 ke dalam bilangan biner. 001, 010, 011, 100, 101. Output yang dihasilkan oleh network ini dibuat demikian karena telah dicoba menghasilkan training yang lebih cepat dan lebih akurat daripada hanya mengeluarkan satu nilai output saja yang langsung bernilai bilangan bulat.

# **HASIL DAN PEMBAHASAN**

## **Pengujian Sistem**

Pengujian sistem dilakukan pada 10 orang yang terdiri dari pria dan wanita. Pengujian dilakukan dalam 3 tahap yaitu:

### **Perekaman Gambar**

Pengujian dilakukan kepada 10 orang pria dan wanita dengan melakukan perekaman terhadap gerakan tangan: “satu”, “dua”, “tiga”, “empat”, dan “lima” yang kemudian disimpan ke dalam format file gambar dengan menggunakan form Simulasi.

### **Training Gambar**

Hasil rekaman yang berupa gambar tangan dari 10 pria dan wanita tersebut akan digunakan sebagai data training *Artificial Neural Network* dengan menggunakan form Training.

## Pengenalan Gambar

Pengenalan tangan pada 10 orang pria dan wanita tersebut dilakukan dengan menggerakkan tangan membentuk simbol satu sampai lima. Setelah dilakukan proses *recognize* tangan dilakukan pencatatan data gambar yang dilakukan dan hasil pengenalannya. Dari data tersebut dimasukkan ke dalam tabel uji dan dihitung tingkat akurasi dari pengenalan tangan yang telah dilakukan sebelumnya.

Pada sistem ini dilakukan pengujian tingkat akurasi terhadap gerakan tangan berupa gambar yang disimpan dan dikenali secara langsung dengan menggunakan *form Training*, dan *form Simulasi*.

## Hasil Pengujian

Tabel 1 berikut ini adalah hasil evaluasi uji coba pengenalan tangan yang telah dilakukan terhadap gerakan tangan 10 orang pria dan wanita tersebut.

Tabel 1 Hasil Uji Pengenalan Gerakan Tangan 'satu' sampai 'lima' Pada 10 rang Pria & Wanita

No.	Pengenalan untuk gerakan tangan	Akurasi
1	Satu	100%
2	Dua	100%
3	Tiga	100%
4	Empat	80%
5	Lima	100%

Sedangkan Tabel 2 berikut ini adalah hasil dari evaluasi pengenalan tangan yang dilakukan di lingkungan (ruangan dan cahaya) yang berbeda terhadap *network* yang telah dilatih dengan data 10 orang sebelumnya, yang ditunjukkan oleh tabel berikut.

Tabel 2 Hasil Uji Pengenalan Gerakan Tangan 'satu' sampai 'lima' Pada Lingkungan yang Berbeda

No.	Dicoba	Dikenali	Akurasi
1	satu	satu	100%
2	dua	dua	100%
3	tiga	satu	0%
4	empat	empat	100%
5	lima	lima	100%

Hasil pengujian menunjukkan bahwa program aplikasi *Hands Recognizer* ini mampu mengenali gerakan tangan dengan tingkat akurasi hingga 96% untuk gambar yang sebelumnya telah di-*training*, dan tingkat akurasi 80% untuk gambar yang diambil di lingkungan baru, dan belum pernah di-*training* sebelumnya.

## PENUTUP

Berdasarkan hasil perancangan program ini didapat bahwa, program aplikasi *Hands Recognizer* ini telah dapat diimplementasikan dengan cukup baik, dalam arti dapat mengenali gerakan-gerakan tangan manusia berupa simbol angka satu sampai dengan lima dengan tingkat akurasi yang cukup tinggi. Program aplikasi *Hands Recognizer* ini mampu melakukan pengenalan dimulai dengan menerima input dari tangan user, mengenalinya, dan kemudian menunjukkan hasil yang ditangkap oleh webcam. Metode *Back Propagation* yang digunakan dalam program aplikasi pengenalan tangan (*Hands Recognizer*) ini menggunakan input layer dengan 10 node, satu hidden layer dengan 7 node, dan output layer dengan 3 node. Jumlah node pada setiap layer tersebut didapat setelah melakukan beberapa percobaan terlebih dahulu, sehingga proses pengenalan tangan dapat berjalan sebaik mungkin.

### Ucapan Terima Kasih

Dalam kesempatan ini pemakalah mengucapkan banyak terima kasih kepada **Andra Basuki** yang telah membantu dalam penyelesaian makalah ini, khususnya dalam pembuatan program komputernya, sehingga makalah ini dapat terselesaikan dengan baik.

## DAFTAR PUSTAKA

- Fausett, L. (1994). *Fundamentals of Neural Networks, Architecture, Algorithms, and Applications*. Eaglewood Cliffs: Prentice Hall.
- Halici, U. (2000). *Introduction to Neural Networks "Biological to Artificial Neuron"*. <http://vision1.eee.metu.edu.tr/~halici/courses/543LectureNotes/lecturenotes-pdf/ch1.pdf>
- Siang, J. J. (2007). *Jaringan Syaraf Tiruan dan Pemrogramannya Menggunakan Matlab*. Yogyakarta: Andi.
- Yoruk, E. Konukoglu, E. Sankur, B. Darbon, J., (2006). Shape-based hand recognition, *IEEE transactions on image processing*, Vol. 15, Issue 7, Page 1803-1815.