

SISTEM PENGENALAN BERDASARKAN CIRI GARIS TELAPAK TANGAN MENGGUNAKAN METODE JARINGAN SARAF TIRUAN PERAMBATAN BALIK

Melly Arisandi^{*)}, R. Rizal Isnanto, and Ajub Ajulian Zahra

Jurusan Teknik Elektro, Universitas Diponegoro Semarang
Jl. Prof. Sudharto, SH, kampus UNDIP Tembalang, Semarang 50275, Indonesia

^{*)} Email: mellyarisandi@gmail.com

Abstrak

Pada era informasi saat ini, kebutuhan terhadap sistem pengenalan diri (personal recognition) secara otomatis yang handal dan dapat dipercaya semakin meningkat terutama untuk sistem keamanan. Salah satu cara yang aman adalah dengan menggunakan teknologi biometrik. Sistem Biometrik adalah teknologi pengenalan diri dengan menggunakan bagian tubuh atau perilaku manusia. Telapak tangan merupakan salah satu biometrika yang memiliki karakteristik unik berupa garis-garis utama pada telapak tangan dan bersifat stabil. Pada penelitian ini, ciri-ciri garis telapak tangan diperoleh dengan menggunakan metode deteksi garis dan operasi blok. Sedangkan proses pencocokan menggunakan metode jaringan saraf tiruan perambatan balik. Pengambilan citra 3 posisi sekaligus, yaitu posisi tegak, 90° ke kanan, dan 90° ke kiri. Pada sistem ini menggunakan 270 citra, di antaranya 180 citra latih dan 90 citra uji. Sistem ini diuji dengan menggunakan 90 citra uji telapak tangan milik 30 orang. Dari hasil pengujian, dapat disimpulkan bahwa sistem pengenalan garis-garis telapak tangan yang diujikan dengan citra uji 3 posisi menghasilkan tingkat keberhasilan sebesar 88,88%. Pengujian dengan citra uji posisi tegak menghasilkan tingkat keberhasilan sebesar 90%. Pengujian dengan citra uji posisi 90° ke kanan menghasilkan tingkat keberhasilan sebesar 93,33%. Pengujian dengan citra uji posisi 90° ke kiri menghasilkan tingkat keberhasilan sebesar 90%.

Kata kunci: biometrik, deteksi garis, operasi blok, jaringan saraf tiruan perambatan balik.

Abstract

In this information era, an automatic and reliable personal identification is coming necessary as a need. The reliability of conventional methods for identification using password and card is currently low as those could be used by anyone so that the security becoming unsafe. One of the technology for solving this issue is by using Biometrics. Biometrics is a personal authentication technology, which recognizes person by using body features or human behavior. Palm hands which is one of Biometrics has a unique characteristics on the palm prints. In this research, palmprint characteristics is provided by line detection method and block operation and for the matching process is by training neural network classifier for the verification. The palmprint take in 3 positions : straight vertical position, 90° angle to the right, and 90° angle to the left. This system uses 270 images, consists of 180 training images and 90 test images. The test results show that the performance of palmprint recognition system with 3 positions of test images provides success rate 88,88 %. The test with straight vertical position provides success rate 90 %. The test with 90° angle to the right position provides success rate 93,33 %. The test with 90° angle to the left position provides success rate 90 %.

Keywords: biometric, line detection, block operation, palm print, Backpropagation, neural network.

1. Pendahuluan

Pada saat ini, sistem pengenalan seseorang (*personal recognition*) secara otomatis menjadi sesuatu yang sangat penting. Pada dasarnya, metode-metode untuk sistem pengenalan dapat dikelompokkan menjadi 3, yaitu pertama, berdasarkan pada sesuatu yang dimiliki (*possessions based*), seperti kunci dan kartu. Kedua, berdasarkan pada sesuatu yang diketahui (*knowledge*

based), seperti identitas pengguna (*user ID*), PIN, dan *password*, dan ketiga, berdasarkan biometrika (*biometrics based*). Penggunaan kunci atau kartu memiliki beberapa kelemahan, seperti: dapat hilang atau dicuri, dapat digunakan secara bersama-sama, dan mudah diduplikasi. Demikian juga halnya dengan penggunaan *user ID*, PIN, dan *password* menimbulkan beberapa permasalahan, seperti: tidak diingat, dan dapat digunakan secara

bersama-sama, dan beberapa *password* mudah diperkirakan.

Salah satu cara yang aman adalah dengan menggunakan teknologi biometrik. Biometrika menggunakan karakteristik unik dari bagian tubuh atau tingkah laku manusia. Pada penelitian ini bertujuan untuk membuat penelitian system identifikasi dengan memakai salah satu dari biometrika yaitu telapak tangan. Telapak tangan adalah salah satu biometrika yang memiliki karakteristik unik berupa garis-garis utama pada telapak tangan dan bersifat stabil

2. Metode

2.1 Pra Pengolahan

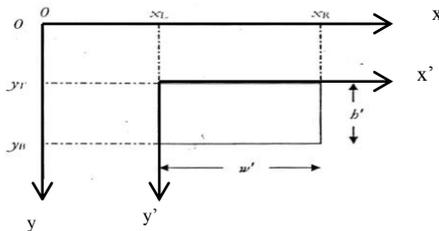
2.1.1 Pemotongan Citra

Pemotongan (*cropping*) adalah operasi untuk memotong satu bagian dari citra sehingga diperoleh citra yang berukuran lebih kecil. Rumus yang digunakan untuk operasi ini adalah

$$x' = x - x_L \quad \text{untuk } x = x_L \text{ sampai } x_R \quad (1)$$

$$y' = y - y_T \quad \text{untuk } y = y_T \text{ sampai } y_B \quad (2)$$

(x_L, y_T) dan (x_R, y_B) masing-masing adalah koordinat titik pojok kiri atas dan pojok kanan bawah bagian citra yang akan dipotong seperti ditunjukkan pada Gambar 1^[4]



Gambar 1. Koordinat titik pojok bagian bawah citra yang akan dipotong^[4]

Akibat pemotongan, ukuran citra berubah menjadi:

$$h' = y_B - y_T \quad (3)$$

$$w' = x_R - x_L \quad (4)$$

Keterangan :

x' = posisi kolom dari pojok kiri atas area yang akan dipotong.

y' = posisi baris dari pojok kiri atas area yang akan dipotong.

h' = tinggi area yang akan dipotong.

w' = lebar area yang akan dipotong.

2.1.2 Citra Aras Keabuan (*Grayscale*)

Proses awal yang banyak dilakukan dalam image processing adalah mengubah citra berwarna menjadi *gray-scale*. Hal ini digunakan untuk menyederhanakan model. Citra berwarna terdiri dari 3 layer matrik yaitu R-

layer, G-layer, dan B-layer. Dalam citra ini tidak ada lagi warna yang ada hanya derajat keabuan. Perhitungan yang digunakan untuk mengubah citra berwarna yang mempunyai nilai matriks masing-masing R, G, dan B menjadi citra *grayscale* dapat dilakukan dengan mengambil rata-rata dari nilai R, G, dan B Untuk mengubah citra berwarna menjadi Gray-scale digunakan rumus berikut

$$gray = 0.2899 * R + 0.5870 * G + 0.1140 * B \quad (5)$$

Keterangan:

Gray : Nilai *grayscale*

R : Nilai pada komponen R-layer

G : Nilai pada komponen G-layer

B : Nilai pada komponen B-layer

2.1.3 Penyesuaian Nilai Intensitas

Penyesuaian nilai intensitas digunakan untuk menyesuaikan nilai-nilai intensitas tiap piksel menjadi nilai-nilai dalam rentang yang dikehendaki. Pada penyesuaian nilai intensitas ini digunakan fungsi *imadjust*, dimana harus menentukan nilai-nilai [low_in high_in] dan [low_out high_out]. Dalam hal ini dapat menggunakan matriks kosong ([]) untuk [low_in high_in] atau [low_out high_out], dimana memiliki arti nilai default adalah [0 1].

Dalam menggunakan *imadjust* dapat bebas dalam menentukan rentang nilai input [low_in high_in] dan nilai output [low_out high_out] dengan menggunakan *imadjust*. Nilai pertama menetapkan nilai intensitas terendah dan nilai intensitas tertinggi dari gambar yang sebelum melalui tahap penyesuaian nilai intensitas. Nilai yang kedua menentukan rentang nilai output yang diinginkan pada gambar keluaran dari tahapan ini.

2.1.4 Penajaman Citra (*image sharpening*)

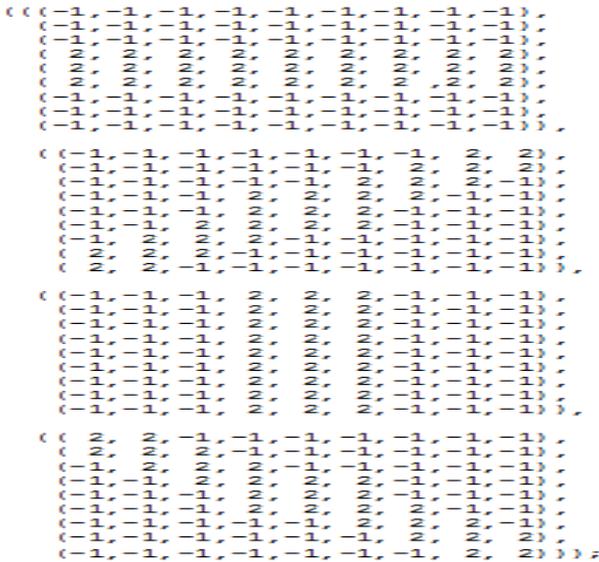
Penajaman citra yaitu mengubah nilai piksel secara sistematis sehingga menghasilkan efek kenampakan citra yang lebih ekspresif sesuai dengan kebutuhan pengguna. Penajaman Kontras ini bertujuan untuk memperoleh kesan kontras yang lebih tinggi. Dengan mentransformasi seluruh nilai kecerahan. Hasilnya adalah berupa citra baru dengan nilai maksimum baru lebih tinggi dari nilai maksimum awal, dan nilai minimum baru lebih rendah dari nilai minimum awal. Jika dilihat secara visual hasilnya berupa citra baru yang variasi hitam putihnya lebih menonjol sehingga tampak lebih tajam dan memudahkan proses interpretasi^[12].

2.2 Ekstraksi Ciri

2.2.1 Deteksi garis (*line detection*)

Deteksi garis (*line detection*) digunakan untuk mendeteksi garis-garis dalam sebuah citra, dalam hal ini

garis telapak tangan. Untuk mendeteksi garis digunakan 4 mask pendeteksi garis pada arah 0°, 45°, 90° dan 135° seperti ditunjukkan pada Gambar 2. [8]. Untuk menghasilkan rajah yang lebih jelas, maka mask tersebut diperluas menjadi 9x9.



Gambar 2. Masking dengan arah 0°, 45°, 90°, 135°

Jika mask pertama digerakkan mengelilingi citra maka akan menghasilkan lebih dekat garis-garis (ketebalan satu piksel) dengan arah horisontal. Dengan latar belakang yang konstan, respon maksimum akan dihasilkan ketika garis dilewati baris tengah dari mask. Dengan cara yang sama, mask yang kedua pada gambar akan memberikan respon terbaik untuk garis yang diarahkan 45°, mask yang ketiga untuk garis-garis vertikal dan mask yang keempat untuk garis-garis dengan arah 135°.

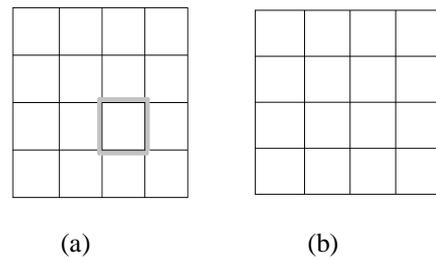
Dari proses masking di atas diperoleh 4 citra dengan arah berbeda-beda yaitu $I_1(x,y), I_2(x,y), I_3(x,y), I_4(x,y)$. Citra akhir hasil deteksi garis adalah $I'(x,y)$ diperoleh dengan mencari nilai maksimum dari keempat citra tersebut^[14].

$$I'(x,y) = \max \{I_1(x,y), I_2(x,y), I_3(x,y), I_4(x,y)\} \quad (6)$$

2.2.2 Operasi blok (block processing)

Disebut juga operasi blok-blok pembeda (*Distinct Blocks*). *Distinct Blocks* adalah partisi, berbentuk kotak yang membagi sebuah daerah matriks berukuran m x n. *Distinct Blocks* membagi matriks citra mulai dari kiri atas tanpa ada *overlapping* / penumpukan. Jika blok tidak dapat terbagi secara pas, maka *toolbox* akan menambahkan *zero padding*^[10]. Terdapat 2 jenis pembagian blok yaitu pembagian blok yang saling tumpang tindih (*overlapping*) dan pembagian blok yang tidak saling tumpang tindih (*non-overlapping*).

Pada pembagian blok yang saling tumpang tindih, suatu blok dengan blok lain yang saling berdampingan terdapat sejumlah piksel yang saling tumpang tindih. Sedangkan pada pembagian blok yang tidak saling tumpang tindih (*non-overlapping*), piksel suatu blok dengan blok yang lain tidak saling tumpang tindih. Gambar 3. menunjukkan pembagian blok secara tumpang tindih dan tidak saling tumpang tindih. Daerah abu-abu pada gambar menunjukkan bagian yang saling tumpang tindih.



Gambar 3. (a) Pembagian blok yang saling tumpang tindih. (b) pembagian blok tidak saling tumpang tindih.

Vektor ciri dari blok yang dibentuk dengan menghitung nilai rata-rata setiap blok. Dengan rumus di bawah ini.

$$\mu = M^{-1} \sum_{i=1}^M x_i \quad (7)$$

Dengan M menyatakan jumlah seluruh piksel pada setiap blok dan x adalah nilai dari piksel.

2.3 Metode Pengenalan

2.3.1 Jaringan Saraf Tiruan Perambatan Balik

Jaringan Saraf Tiruan Perambatan Balik (*Backpropagation Neural Network*) merupakan salah satu model dari jaringan saraf tiruan umpan maju dengan menggunakan pelatihan terbimbing yang disusun berdasar pada algoritma kesalahan perambatan balik. Pola masukan dan target diberikan sebagai sepasang data. Bobot-bobot awal dilatih dengan melalui tahap maju untuk mendapatkan galat keluaran yang selanjutnya galat ini digunakan dengan tahap mundur untuk memperoleh nilai bobot yang sesuai agar dapat memperkecil nilai galat sehingga target keluaran yang dikehendaki tercapai. Tujuan dari model ini adalah untuk mendapatkan keseimbangan antara kemampuan jaringan untuk mengenali pola yang digunakan selama proses pelatihan berlangsung serta kemampuan jaringan memberikan tanggapan yang benar terhadap pola masukan yang berbeda dengan pola masukan selama pelatihan^[2].

2.4 Perancangan Sistem

Hasil perancangan perangkat keras sistem identifikasi garis-garis telapak tangan dapat dilihat pada Gambar 4.

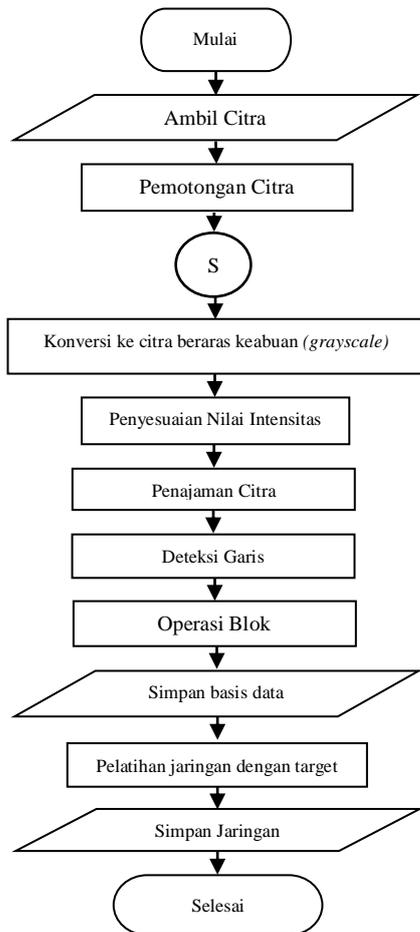


Gambar 4. Alat pengambilan citra telapak tangan.

Gambar di atas menunjukkan alat yang digunakan untuk mengambil citra telapak tangan. Terdiri dari sebuah tiang untuk menyangga kamera setinggi 40 cm (A), dan papan berukuran 40x40 cm dengan gambar telapak tangan dengan 3 posisi yang berbeda (B). Gambar telapak tangan yang terletak di papan berfungsi agar telapak tangan tidak berubah saat pengambilan gambar.

2.4.1 Algoritma Proses Pelatihan

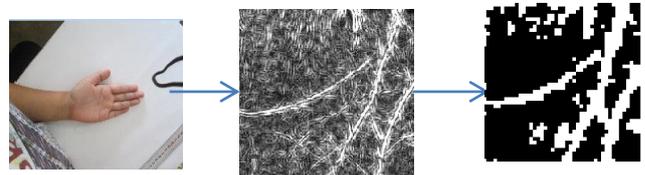
Proses pelatihan merupakan proses pendaftaran ciri telapak tangan pengguna ke sistem untuk membangun basis data acuan. Ciri yang diperoleh merupakan hasil proses ekstraksi ciri citra telapak tangan. Diagram alir proses ekstraksi ciri dapat dilihat pada Gambar 5.



Gambar 5. Diagram alir untuk proses pelatihan

Tahapan awal proses prapengolahan adalah pemotongan citra untuk mengambil daerah garis-garis utama telapak tangan. Selanjutnya citra akan diubah menjadi citra keabuan (*grayscale*). Proses penyesuaian nilai intensitas bertujuan menyesuaikan nilai-nilai intensitas tiap piksel menjadi nilai-nilai dalam rentang yang dikehendaki. Dalam program ini menggunakan range [0.3 0.7] pada nilai input dan [0 1] pada nilai output, bertujuan untuk memperjelas daerah terdekat yang terdapat garis telapak tangan.

Proses penajaman citra bertujuan untuk menajamkan citra terutama untuk bagian tepi citra. Kemudian citra akan melalui proses deteksi garis. Proses deteksi garis bertujuan untuk mengetahui garis-garis pada telapak tangan. Proses selanjutnya adalah operasi blok. Hasil dari proses operasi blok berupa matriks yang disimpan dalam basis data sebagai matrik fitur telapak tangan acuan. Hasil akhir dari proses operasi blok adalah ciri yang digunakan untuk pengenalan. Kemudian langkah selanjutnya adalah pembuatan jaringan dan pembuatan target. Target digunakan untuk acuan pada saat proses pelatihan tersebut. Setelah pembuatan jaringan selesai kemudian proses pelatihan jaringan dengan target tersebut. Kemudian jaringan yang telah dilatih disimpan dalam *.mat*. Gambar 6. dibawah ini adalah contoh gambar sebelum dan sesudah melalui proses prapengolahan dan ekstraksi ciri.



Gambar 6. contoh gambar sebelum dan sesudah melalui proses prapengolahan dan ekstraksi ciri.

2.4.2 Algoritma Proses Pengujian

Proses identifikasi merupakan proses utama dari sistem identifikasi telapak tangan. Algoritma proses identifikasi akan menunjukkan bagaimana sistem melakukan proses identifikasi, sehingga dapat menentukan identitas dari telapak tangan yang dijadikan sebagai citra masukan. Diagram alir dari proses identifikasi dapat dilihat pada Gambar 7.

3. Hasil dan Analisa

3.1 Pengujian dengan Citra Uji

Pada proses pengambilan data citra digunakan 30 responden dengan masing-masing responden 9 citra maka total data 270. Data dibagi 2 yaitu data latih 180 citra dan data uji 90 citra. Dalam pengambilan data terdapat 3 macam posisi yaitu posisi normal, 90° ke kanan, dan 90° ke kiri, sehingga pada masing-masing orang diambil setiap posisi 3 citra, dua di antaranya dijadikan citra latih,

sisanya adalah citra uji. Parameter yang dihitung dari pengujian ini adalah tingkat keberhasilan sistem dalam mengidentifikasi suatu telapak tangan. Persamaan untuk tingkat keberhasilan adalah.

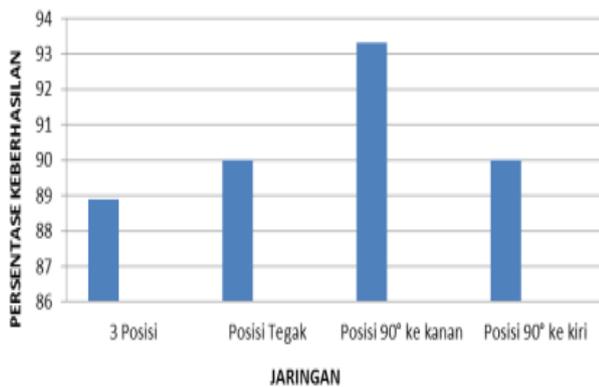
$$\% \text{Keberhasilan} = \frac{\sum \text{Benar}}{\sum \text{Data}} \times 100 \% \quad (8)$$

- **Pengujian dengan Jaringan Kombinasi 3 Posisi**
Pengujian ini menggunakan jaringan yang dilatihkan untuk 3 posisi yaitu posisi tegak, 90° ke kanan, dan 90° ke kiri.
- **Pengujian dengan Masing-Masing Jaringan Posisi Tegak 90° ke Kanan, dan 90° ke Kiri.**
Pada Pengujian ini menggunakan masing-masing jaringan yang telah dilatihkan tiap-tiap posisi, sehingga pada pengujian ini dilatihkan 3 jaringan untuk pengujian masing-masing posisi di antaranya pengujian posisi tegak, posisi 90° ke kanan, dan posisi 90° ke kiri.

Tabel 1. Hasil Pengujian citra uji

Hasil Pengujian				
Kombinasi posisi	3	Posisi Tegak	Posisi 90° ke kanan	Posisi 90° ke kiri
	88,88%	90 %	93,33%	90%

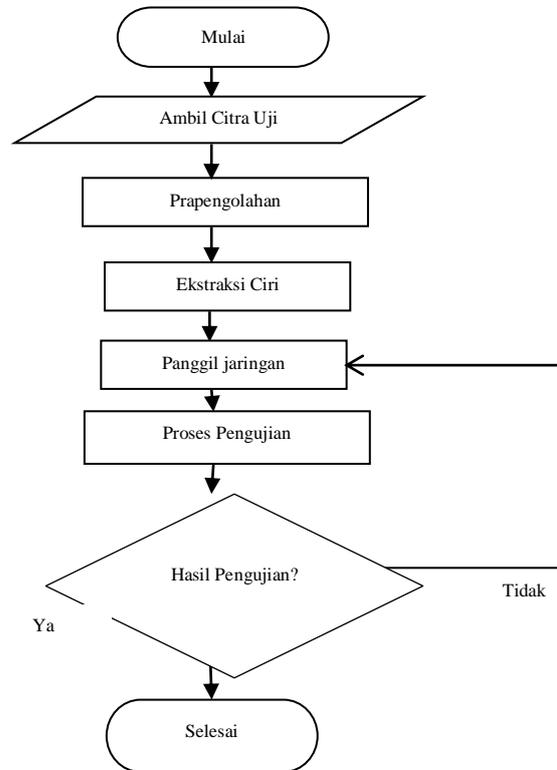
Berdasarkan tabel di atas dapat terlihat persentase masing-masing hasil pengujian. Tabel 1. dapat ditampilkan dalam bentuk grafik batang seperti pada Gambar 8.



Gambar 8. Perbandingan persentase keberhasilan dari 4 pengujian data citra uji

Berdasarkan Gambar 8. merupakan perbandingan persentase keberhasilan pada keempat pengujian. Dapat disimpulkan juga bahwa pengujian dengan menggunakan jaringan yang dilatihkan pada 1 posisi citra telapak tangan hasil persentase keberhasilannya lebih tinggi dibandingkan pengujian dengan menggunakan jaringan yang dilatihkan pada 3 posisi sekaligus. Hal ini dikarenakan adanya korelasi di antara ketiga posisi yang digunakan sebagai variasi data tersebut berbeda jauh yaitu

sebesar 90°, sehingga pengujian dengan jaringan 3 posisi sekaligus hasilnya paling rendah.



Gambar 7. Diagram alir proses pengujian

4. Kesimpulan

Dari hasil pengujian dan analisis yang telah dilakukan pada bab sebelumnya maka dapat diambil kesimpulan bahwa persentase keberhasilan pada pengujian citra uji dengan menggunakan jaringan yang dilatihkan pada ketiga posisi (tegak, 90° ke kanan, dan 90° ke kiri) yaitu sebesar 88,88%. Persentase keberhasilan pada pengujian citra uji dengan menggunakan jaringan yang dilatihkan dengan citra telapak tangan posisi tegak yaitu sebesar 90%. Persentase keberhasilan pada pengujian citra uji dengan menggunakan jaringan yang dilatihkan dengan citra telapak tangan posisi 90° ke kanan yaitu sebesar 93,33%. Persentase keberhasilan pada pengujian citra uji dengan menggunakan jaringan yang dilatihkan dengan citra telapak tangan posisi 90° ke kiri yaitu sebesar 90%. Berdasarkan keempat pengujian dengan menggunakan citra uji maka dapat disimpulkan bahwa persentase tertinggi adalah pada saat menggunakan jaringan yang telah dilatihkan pada posisi 90° ke kanan dan juga dapat dilihat pengujian dengan persentase terendah adalah pada saat menggunakan jaringan yang telah dilatihkan pada ketiga posisi sekaligus. Hasil persentase keberhasilan pengujian dengan jaringan yang telah dilatihkan dengan ketiga posisi sekaligus dapat dilihat bahwa hasilnya terendah di antara keempat pengujian yang lain. Hal ini dikarenakan adanya korelasi di antara ketiga posisi yang

digunakan sebagai variasi data tersebut berbeda sebesar 90°, ini sangat berpengaruh terhadap keberhasilan pengujian.

Untuk Penelitian lebih lanjut diharapkan dapat memperbaiki kekurangan yang ada pada penelitian ini dan diharapkan dapat mengembangkan apa yang telah dilakukan pada penelitian ini. Adapun saran yang dapat diberikan sehubungan penelitian ini adalah perlu penelitian lebih lanjut untuk pengembangan sistem identifikasi telapak tangan dapat dikembangkan dengan metode ekstraksi ciri yang berbeda seperti metode *Principal Component Analysis* (PCA). Selain itu juga dapat dikembangkan dengan menggunakan variasi tipe jaringan saraf tiruan yang lain seperti perceptron, ADALINE, dan LVQ.

Referensi

- [1]. Fatta, H.A., *Konversi Format Citra RGB ke Format Grayscale Menggunakan Visual Basic*, Skripsi S-1, STMIK AMIKOM, Yogyakarta, 2007.
- [2]. Jong, J.S., *Jaringan Syaraf Tiruan dan Pemrogramannya Menggunakan Matlab*, Andi Offset, Yogyakarta, 2005.
- [3]. McAndrew, A., *An Introduction to Digital Image Processing with Matlab*, School of Computer Science and Mathematics Victoria University of Technology, 2004.
- [4]. Mugni, I., *Sistem Identifikasi Berdasarkan Garis Garis Utama Telapak Tangan Menggunakan Metode Overlapping Block*, Universitas Diponegoro, Semarang, 2012.
- [5]. Munir, R., *Pengolahan Citra Digital dengan Pendekatan Algoritmik*, Penerbit Informatika, Bandung, 2004.
- [6]. Putra, D., *Sistem Verifikasi Menggunakan Garis-Garis Telapak Tangan*, Universitas Udayana, 2007.
- [7]. Putra, D., *Sistem Verifikasi Biometrika Telapak Tangan dengan Metode Dimensi Fraktal dan Lacunarity*, Universitas Udayana, 2009.
- [8]. Putra, D., *Sistem Biometrika*, Penerbit Andi, Yogyakarta, 2009
- [9]. Utami, S.E., *Pembacaan Plat Nomor Kendaraan Menggunakan Metode Jaringan Syaraf Tiruan (JST) Backpropagation Berbasis Image Processing*, Jurusan Teknik Telekomunikasi ITS, Surabaya, 2009.
- [10]. Wijaya, CH. M., dan A. Prijono, *Pengolahan Citra Digital Menggunakan MATLAB*, Penerbit Informatika, Bandung, 2007.
- [11]. ---, *Pengolahan Citra :Konsep Dasar*, Grafika Komputer dan Pengolahan Citra, Universitas Gunadarma, 2006 . (23 Mei 2013)
- [12]. Somantri, L., *Penajaman citra (Enhancement)*, http://file.upi.edu/browse.php?dir=Direktori/FPIPS/JUR._PEND._GEOGRAFI/132314541-LILI_SOMANTRI/ ,Mei 2013.
- [13]. ---, *Penyesuaian nilai intensitas*, <http://www.mathworks.com/help/images/ref/imadjust.html> , Oktober 2013.
- [14]. ---, *Image Enhancement Techniques*, <http://www.slideshare.net/matlabcontent/matlab-image-enhancement-techniques-2875192> , Oktober 2013.