

Verifikasi Citra Tanda Tangan Menggunakan Metode Prewitt dan Learning Vector Quantization

Asfanji Sefta¹, Syarif Hidayatulloh²

¹Universitas BSI
e-mail: Asfanjisefta@yahoo.co.id

²Fakultas Teknik Informatika
²Universitas BSI
e-mail: Syarif.sfq@bsi.ac.id

Abstrak

Tanda tangan adalah salah satu bukti persetujuan dari seseorang, Jadi tanda tangan ini memiliki arti yang sangat penting. Sering terjadi Kasus pemalsuan tanda tangan, antara lain disebabkan oleh sistem verifikasi yang tidak baik. Verifikasi tanda tangan ini kebanyakan dilakukan secara manual, Yaitu dengan membandingkan langsung dengan menggunakan mata Manusia yang memiliki banyak kelemahan. Jadi ketelitian dan keakuratan hasil yang diinginkan sering kurang memuaskan. Metode yang saya gunakan dalam membangun aplikasi verifikasi tanda tangan ini adalah dengan menggunakan metode *Edge Detection* dan metode *Vector Quantization Learning*. Program ini dibangun menggunakan Matlab. Berdasarkan hasil penelitian yang diperoleh dari pengujian sistem verifikasi tanda tangan dengan menerapkan metode *edge detection operator prewitt* dan metode *Learning vector Quantization* dengan penghitung vektor, maka diperoleh kesimpulan bahwa Metode *Edge Detection operator prewitt* dan *Learning Vector Quantization* dengan penghitung vektor dapat mengekstraksi fitur tanda tangan untuk memproses vektor yang digunakan dalam penghitungan vektor untuk mengenali tanda tangan yang asli dan yang palsu pada aplikasi verifikasi tanda tangan yang membantu memverifikasi tanda tangan sehingga meminimalisasi pemalsuan tanda tangan.

Kata Kunci : Citra, *Edge Detection*, *Vector Quantization Learning*, Tanda Tangan

Abstract

The signature is one of the proof of approval from a person, so this signature has a very important meaning. There are often cases of signature forgery, partly due to a poor verification system. This signature verification is mostly done manually, that is by comparing it directly using the eyes of a human who has many weaknesses. So the accuracy and accuracy of the desired results are often unsatisfactory. The method that I use in building this signature verification application is to use the Edge Detection method and the Vector Quantization Learning method. This program is built using Matlab. Based on the research results obtained from the signature verification system testing by applying edge detection operator prewitt method and Learning vector Quantization method with vector counters, the conclusion is that the Edge Detection Method operator prewitt and Learning Vector Quantization with vector counters can extract signature features to process vectors used in vector counting to identify original and fake signatures in the signature verification application that helps verify the signature so as to minimize signature falsification.

Keywords: Image, *Edge Detection*, *Vector Quantization Learning*, Signature

1. Pendahuluan

Tanda tangan merupakan salah satu bukti pengesahan dari seseorang sehingga tanda tangan ini mempunyai arti yang sangat penting. Seringnya terjadi kasus pemalsuan tanda tangan, antara lain disebabkan oleh sistem verifikasi yang kurang baik. Verifikasi tanda tangan

kebanyakan dilakukan dengan cara manual, yaitu dengan membandingkan secara langsung dengan menggunakan mata manusia yang mempunyai banyak kelemahan (Hayatunnufus, 2010). Sehingga ketelitian dan ketepatan hasil yang diinginkan seringkali kurang memuaskan. Karena tanda tangan merupakan mekanisme *primer* untuk

authentication dan *authorization* dalam transaksi legal, kebutuhan akan penelitian pada pengembangan aplikasi pengenalan dan verifikasi tanda tangan yang efisien meningkat pada tahun-tahun terakhir ini.

Pengenalan dan verifikasi tanda tangan meliputi dua bagian yang berbeda tetapi berkaitan erat satu sama lain. Yang pertama adalah identifikasi dari pemilik tanda tangan, sedangkan yang kedua adalah keputusan apakah tanda tangan tersebut asli atau dipalsukan (Yustisia, 2010). Selain itu, bergantung pada kebutuhannya, pengenalan tanda tangan dibagi menjadi dua kelas yang berbeda yaitu pengenalan dan verifikasi tanda tangan *online* dan *offline*. Pada pengenalan *online*, dibutuhkan beberapa alat bantu khusus yang digunakan untuk mengukur kecepatan dan tekanan tangan ketika membuat tanda tangan.

Dalam proses verifikasi tanda tangan, *proses* klasifikasi sama pentingnya dengan *proses* ekstraksi fitur. Setelah fitur-fitur penting data atau sinyal yang dihasilkan pada proses ekstraksi fitur, fitur-fitur tersebut nantinya akan digunakan untuk *proses* klasifikasi. Metode klasifikasi yang digunakan adalah pengklasifikasi LVQ (*Learning Vector Quantization*) adalah suatu metode klasifikasi pola yang masing-masing unit keluaran mewakili kategori atau kelas tertentu (SIRAIT, 2007). Suatu lapisan kompetitif akan secara otomatis belajar untuk mengklasifikasikan vektor-vektor *input*. Kelas-kelas yang didapatkan sebagai hasil dari lapisan kompetitif ini hanya tergantung pada jarak antara vektor-vektor *input*. Jika 2 vektor *input* mendekati sama, maka lapisan kompetitif akan meletakkan kedua vektor *input* tersebut ke dalam kelas yang sama, dengan membobotkan lagi hasil nilai kedalam vektor *Eigen* dan *Theta*. Nilai *Eigen* dan *theta* adalah nilai karakteristik dari suatu matriks berukuran $n \times n$, sementara vektor *Eigen* dan *theta* adalah vektor kolom bukan nol yang bila dikalikan dengan suatu matriks berukuran $n \times n$ akan menghasilkan vektor lain yang memiliki nilai kelipatan dari vektor *Eigen* dan *theta* itu sendiri.

2. Metode Penelitian

Metode yang digunakan adalah metode Research and Development (R&D). Metode penelitian dan pengembangan adalah metode yang digunakan untuk

menghasilkan produk tertentu, dan menguji keefektifan produk tersebut (Sugiyono, 2013).

Metode yang digunakan menggunakan model konseptual atau model teoritis, model hipotetik, model empiric dan model akhir.

Pengembangan model konseptual dihasilkan berdasarkan latar belakang masalah, kajian teoritis dan hasil penelitian-penelitian yang relevan. Model hipotetik dikembangkan dari hasil temuan di lapangan melalui observasi, wawancara, dan studi pustaka untuk mendapatkan hasil analisis kebutuhan (*need analysis*) sistem pendukung keputusan untuk menentukan pemeringkatan kualitas lembaga-lembaga litbang.

Model hipotetik didapatkan berdasarkan bagaimana keadaan di lapangan, dalam hal ini adalah bagaimana proses pengambilan keputusan untuk menentukan pemeringkatan lembaga litbang.

Dari model konseptual yang ada, dilakukan penelitian pendahuluan (*initial research*) dalam bentuk analisis kebutuhan terhadap kriteria yang ingin digunakan untuk penentuan pemeringkatan lembaga. Pengumpulan data dilakukan dengan observasi, wawancara. Analisis yang dilakukan yaitu dengan melakukan studi pustaka yang berkaitan dengan sistem pendukung keputusan, yang termasuk dalam penelitian, langkah-langkah perhitungannya, serta teori yang dibutuhkan. Observasi yang dilakukan secara langsung di sekelompok orang untuk mendapatkan informasi yang dibutuhkan.

Untuk membuat perangkat lunak, penulis melakukan tahapan-tahapan dalam penggunaan metode *Waterfall*. Dan berikut ini tahapan yang penulis gunakan dalam penelitian ini :

Analisa Kebutuhan Software

Penentuan pemeringkatan yang digunakan sebelumnya masih kurang efisien dan efektif, maka dibutuhkan sebuah sistem yang mampu mempermudah dan lebih efektif dan efisien. dan juga penulis memerlukan suatu *interface* sistem yang dapat memudahkan bagi pengguna untuk mendapatkan kemudahan dalam pengisian dan juga membantu penilaian terhadap tim penilai. Dari uraian diatas penulis menentukan *software* yang sesuai dengan kriteria yang dibutuhkan dan yang akan

digunakan sebagai proses verifikasi citra tanda tangan adalah Matlab R2013B atau R2015B

Perancangan Sistem dan Perangkat Lunak

Pada perancangan sistem ini, mengimplementasikan Algoritma *Edge Detection Operator Prewitt dan Learning vector quantization* dengan penghitung vector untuk autentikasi tanda tangan manual dan digital kedalam aplikasi. Aplikasi ini dibangun dengan menggunakan Matlab R2013B atau R2015B.

Implementasi dan Pengujian Unit

Pengimplementasian system yang akan penulis buat ini mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak, agar sesuai dengan perkembangan Teknologi Informasi saat ini. Dan untuk proses pengujian yang penulis lakukan terhadap program ini menggunakan sistem *Blackbox Testing* yang cenderung dapat menemukan hal-hal seperti fungsi yang tidak benar atau tidak

menggunakan alat *scanner* yang kemudian disimpan kedalam *database*. Aplikasi ini akan mendeteksi keaslian tandatangan seseorang dengan cara memverifikasi data tanda tangan yang *input* dengan data tanda tangan yang berada di dalam *database*.

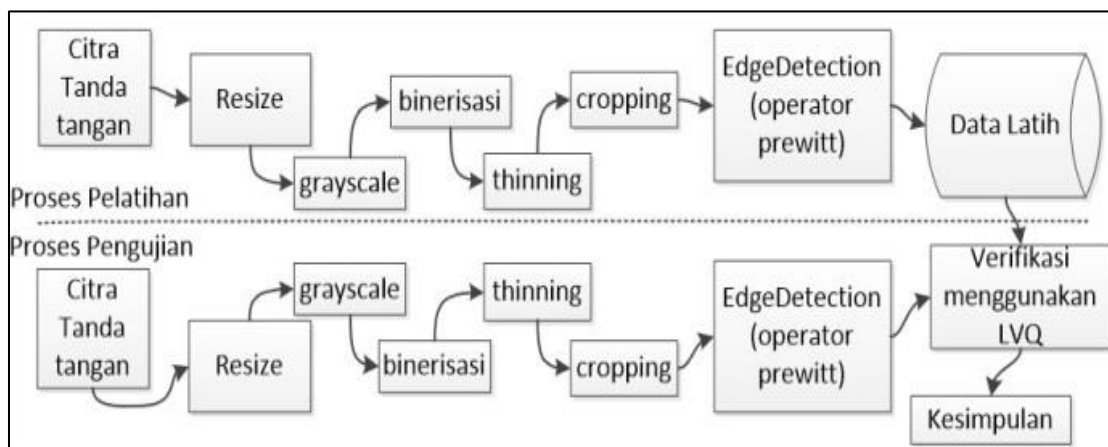
3.1. Analisa Kebutuhan

Dalam pembuatan aplikasi verifikasi tanda tangan untuk tugas akhir ini, perlu diidentifikasi setidaknya tiga kebutuhan perangkat yaitu perangkat lunak (*software*), perangkat keras (*hardware*) dan pengguna yang digunakan dan menunjang proses pembuatan sistem aplikasi ini. Kebutuhan tersebut sebagai berikut :

- Analisis kebutuhan software
- Analisis kebutuhan hardware
- Analisis kebutuhan pengguna(user)

3.2. Perancangan

Untuk menggambarkan langkah-langkah dan urutan prosedur dari aplikasi Verifikasi tanda tangan secara umum maka dibuat sebuah model kerja sistem aplikasi



Gambar 1. - Tahapan cara kerja sistem

ada, kesalahan antarmuka (*interface errors*), kesalahan pada struktur data dan akses basis data, kesalahan performansi (*performance errors*), kesalahan inisialisasi dan terminasi.

3. Hasil dan Pembahasan

Tahapan ini dilakukan untuk mengetahui apa saja yang pengguna (*user*) butuhkan terhadap aplikasi yang akan dibuat. Hal ini dilakukan agar aplikasi yang akan dibuat sesuai dengan yang dibutuhkan oleh pengguna. Sasaran dari aplikasi ini adalah pengguna dapat melakukan penginputan tandatangan dengan

secara umum seperti pada Gambar 1.

Penjelasan dari gambar 1, Sistem ini dibagi menjadi dua tahap yaitu *proses pelatihan* dan *proses klasifikasi / pencocokan*.

Pada *proses pendaftaran* dan *klasifikasi* memiliki tahapan yang sama dari tahap akuisisi citra sampai ekstraksi ciri. Algoritma *proses pendaftaran* akan dijelaskan pada *flowchart*.



Gambar 2. - flowchart pendaftaran / data latih

Seperti yang ditunjukkan pada gambar 2, dijelaskan *flowchart* proses pendaftaran aplikasi verifikasi tanda tangan.

3.2.1. Tahap Preprocessing

Tahap preprocessing adalah tahap awal yang terdiri dari proses *Load Image*, *Resize*, *Grayscale*, *binerization*, *thinning* dan *cropping*.

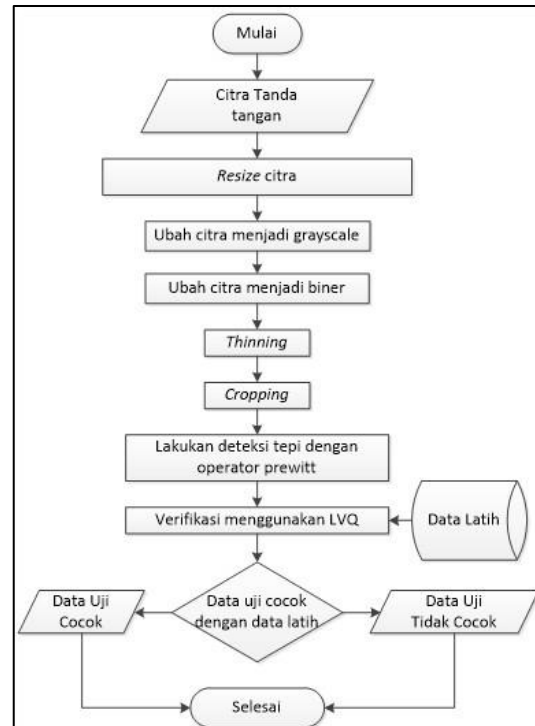
3.2.2. Tahap Ekstraksi ciri

Edge detection merupakan langkah pertama untuk melingkupi informasi didalam citra. Tepi mencirikan batas-batas objek dan karena itu tepi berguna untuk segmentasi dan identifikasi objek di dalam citra.

Pada aplikasi verifikasi tanda tangan digunakan deteksi tepi dengan operator gradien pertama yaitu operator *prewitt*.

3.2.3. Tahap Pencocokan

Setelah tahap ekstraksi ciri maka dilakukan pencocokan dengan metode jaringan syaraf tiruan *Learning Vector Quantization*. Dengan membandingkan bobot yang ada di data latih. Algoritma pada proses pencocokan di jelaskan pada *flowchart* yang digambarkan pada gambar III.5.



Gambar 3. - flowchart tahap pencocokan

3.2.4. Tahap LVQ

Algoritma LVQ bertujuan akhir mencari nilai bobot yang sesuai untuk mengelompokkan vektor – vektor kedalam kelas tujuan yang telah di inialisasi pada saat pembentukan jaringan LVQ. berikut algoritma dari pelatihan LVQ:

- A. Tetapkan: Bobot (W), Maksimum Epoch ($MaxEpoch$), error minimum yang diharapkan (Eps), learning rate (α)
- B. Masukkan :
 1. Data input: $x(m,n)$
 2. Target berupa kelas : $T(1,n)$
- C. Tetapkan kondisi awal :
 1. Epoch = 0;
 2. Err = 1.
- D. Kerjakan selama : (epoch < $MaxEpoch$) dan ($\alpha > Eps$)
 1. Epoch = epoch+1;
 2. Kerjakan untuk $i = 1$ sampai n

- i. Memilih (J) jarak sedemikian hingga $\|X - W_j\|$ minimum (sebut sebagai C_j)
- ii. Perbaiki W_j dengan ketentuan:

- Jika $T = C_j$ maka :

$$W_j(\text{baru}) = W_j(\text{lama}) + \alpha (X - W_j(\text{lama}))$$

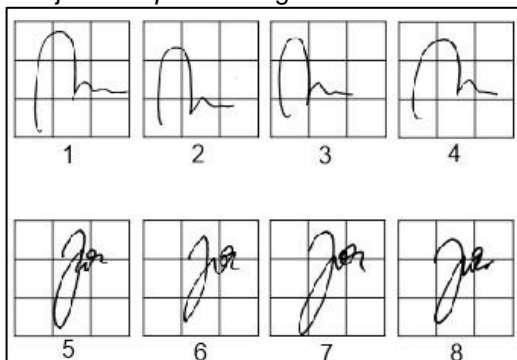
- Jika $C_j \neq T$ maka:

$$W_j(\text{baru}) = W_j(\text{lama}) - \alpha (X - W_j(\text{lama}))$$

- E. Kurangi nilai Pengurangan α . Maka di hasilkan bobot akhir yang akan dihitung dengan data uji untuk bobot akhir dari proses data latih.

3.2.5. Contoh Perhitungan Dengan LVQ

Pada subbab ini disimulasikan perhitungan LVQ menggunakan sampel citra berukuran 3×3 *pixel*, citra sampel memakai nilai yang sederhana sehingga mudah untuk dipahami maksud dari sistem kerja perangkat lunak ini. Misalkan diketahui 8 input vector dalam 2 kelas di contohkan dengan gambar tanda tangan dan dibagi menjadi 3×3 *pixel* sebagai berikut :



Gambar 4. - Tanda tangan dengan 3×3 pixel

Tanda tangan yang dibuat 3×3 *pixel* bertujuan untuk mengetahui *pixel* mana saja yang mengenai garis tanda tangan. *Pixel* yang mengenai garis tanda tangan diberi nilai 1 dan yang tidak mengenai garis tanda tangan diberi nilai 0 contoh sebagai berikut:

1	1	0	1	0	0	1	0	0	1	1	0
1	1	1	1	1	0	1	1	1	1	1	1
1	0	0	1	1	1	1	0	0	1	0	0
1			2			3			4		
0	1	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	1	0	1	1	0	1	1
0	1	0	0	1	0	1	1	0	0	1	0
5			6			7			8		

Gambar 5. - matrik dari tanda tangan

Matrik 3×3 *pixel* ini merupakan hasil perubahan dari gambar tanda tangan pada gambar III.7. Angka 1 merupakan *pixel* yang mengenai garis dan 0 yang tidak mengenai garis. Dari hasil matrik tersebut, ubah bentuk matrik kedalam bentuk vektor. Pada setiap vektor, insialisasi kelas untuk mengklasifikasi setiap inputan.

3.2.6. Perancangan Tampilan

Pada tahap ini dilakukan desain perancangan aplikasi untuk implementasi metode tersebut kedalam sebuah aplikasi, berikut adalah skenario singkat dari aplikasi yang akan dibangun:

- a. *User* dari aplikasi ini adalah manusia.
- b. *User* tidak memerlukan *login* untuk mengakses aplikasi ini.
- c. *User* dapat mengambil citra dengan memilih *button* ambil gambar.
- d. Setelah memilih citra, *User* dapat melakukan proses ekstraksi ciri dengan memilih *button* ubah ke *prewitt* maka akan muncul gambar yang sudah di ekstraksi ciri.
- e. Setelah melakukan proses langkah d, Citra akan ter *save* pada data yang berupa vektor yang di *save* dalam bentuk *.mat*. *user* dapat mengulang langkah a sampai dengan e untuk mengisi data latih.
- f. Setelah melakukan proses langkah e, *User* dapat melakukan proses verifikasi dengan kembali mengambil citra, citra yang di uji
- g. Setelah melakukan proses langkah f, *User* memilih *button* verifikasi untuk melihat hasil kecocokan pada hasil pencocokan dengan *database*.
- h. Setelah keluar hasil *user* dapat melakukan penghapusan *axes* dan kolom agar kembali kosong seperti semula dengan menekan *button* *Reset*.

3.2.7. Perancangan software

Perancangan sistem yang dilakukan pada penelitian ini menggunakan UML (*Undefined Model Language*). Pada sistem aplikasi ini digunakan *use case diagram*, skenario *use case*, *class diagram*, *diagram activity* dan *sequencediagram*.

3.3. Tampilan Hasil

Tahap implementasi merupakan tahap penerapan sistem yang telah didesain atau dirancang, sehingga sistem yang telah dibuat dapat dioperasikan dan digunakan secara optimal sesuai dengan kebutuhan :

3.3.1. Tampilan Form Opening

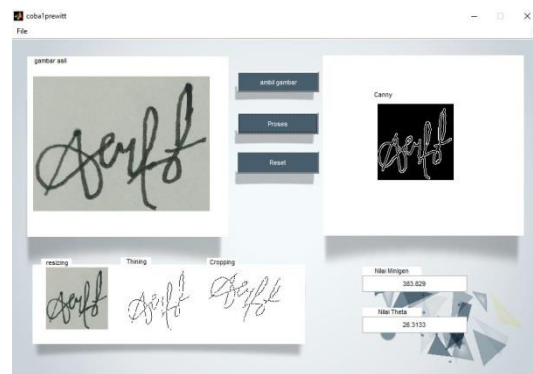
Tampilan awal dari program klasifikasi tanda tangan menampilkan tampilan awal untuk memilih tombol *proses* verifikasi dan tombol melakukan verifikasi tanda tangan. Tombol *proses* verifikasi memperlihatkan *proses* yang program lakukan untuk mendapatkan klasifikasi.



Gambar 6. Tampilan Form Opening

3.3.2. Tampilan Form Process Verification

Pada *form process verification* memberikan informasi tampilan *proses* dalam klasifikasi. Diantaranya ada tombol ambil gambar untuk mengambil citra yang akan diproses, tombol *proses* untuk memberikan informasi ketika gambar diolah untuk memberikan informasi dan tombol *reset* dipergunakan untuk menghapus atau mengembalikan kepada posisi semula.



Gambar 7. Tampilan Form Process

3.3.3. Tampilan Form Start Verification

Pada *form* ini menampilkan hasil klasifikasi yang terdiri dari gambar asli dan gambar tanda tangan yang akan diujikan. Pada *form* ini menampilkan hasil dari klasifikasi tanda tangan dengan 2 kemungkinan yaitu cocok atau tidak cocok.



Gambar 8. Form Start Verification

3.4. Stuktur Coding

Berikut adalah penjelasan *listing* program untuk *proses* verifikasi tanda tangan dengan metode *edge detection* operator *prewitt* dan metode *learning vector quantization*.

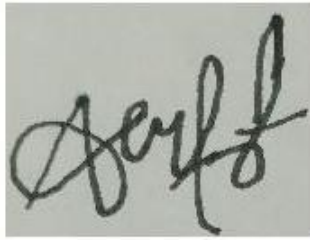
3.5. Cara Kerja Perangkat Lunak

Berikut ini adalah penjelasan beberapa *syntax* program yang digunakan dalam aplikasi :

3.5.1. Masukan Tanda Tangan Asli

Pada *form* klasifikasi dimulai dengan mengambil gambar tanda tangan asli yang akan diujikan dengan tanda tangan uji.

Tanda tangan Asli



Gambar 9. Tanda Tangan Asli

3.5.2. Masukan Tanda Tangan Uji

Pada *form* klasifikasi dimulai dengan mengambil gambar tanda tangan uji yang akan diujikan dengan tanda tangan asli.

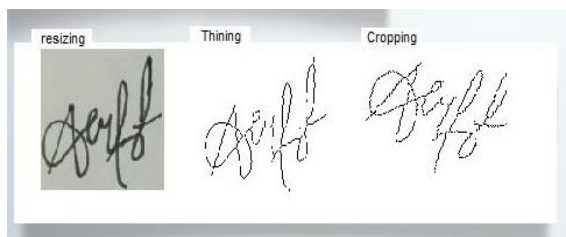
Tanda tangan Uji



Gambar 10. Tanda Tangan Uji

3.5.3. Perbaikan Kualitas Citra

Perbaikan kualitas citra dilakukan untuk mendapatkan hasil yang maksimal dari perhitungan gambar tanda tangan asli dan tanda tangan uji, agar nilai yang diperoleh lebih akurat.



Gambar 11. Perbaikan Kualitas Citra

3.5.4. Edge Detection Operator Prewitt

Deteksi tepi dilakukan dengan tujuan untuk memperoleh nilai yang akurat untuk perhitungan tanda tangan, dikarenakan pola yang dibentuk dari tanda tangan harus dideteksi tepi dengan akurat.

Canny



Gambar 12. Deteksi tepi dengan Operator Prewitt

3.6. Klasifikasi Tanda Tangan

Klasifikasi tanda tangan dilakukan dengan cara membandingkan nilai minIgen dan nilai theta. Dari gambar 4.8 dan gambar 4.9 diperlihatkan hasil klasifikasi antara tanda tangan yang cocok dengan tanda tangan yang tidak cocok.



Gambar 13. Tanda Tangan Cocok



Gambar 14 . Tanda Tangan Tidak Cocok

4. Kesimpulan Dan Saran

Berdasarkan hasil penelitian yang diperoleh dari pengujian sistem verifikasi tanda tangan dengan menerapkan metode *edge detection* operator *prewitt* dan metode *Learning vector Quantization* dengan penghitung vektor, maka diperoleh kesimpulan bahwa Metode *Edge Detection* operator *prewitt* dan *Learning Vector Quantization* dengan penghitung vektor dapat mengekstraksi fitur tanda tangan

untuk memproses vektor yang digunakan dalam penghitungan vektor untuk mengenali tanda tangan yang asli dan yang palsu.

Aplikasi verifikasi tanda tangan menggunakan metode *edge detection* operator *prewitt* dan *learning vector quantization* dengan penghitung vektor dapat mempermudah pendeteksian keaslian tanda tangan.

Referensi

- Faroqi, A. & Wahana, A . (2012). Sistem pendukung keputusan untuk menyeleksi calon siswa menggunakan metode Analytical Hierarchy Process (AHP). *Jurnal ISTEK*, 60-70.
- Afif, R. (2010). *Autentifikasi Pengenalan Pola Tanda Tangan Manual Menggunakan jaringan syaraf tiruan LVQ dan tanda tangan digital menggunakan algoritma RSA*. Jakarta: Andi Publisher.
- Arief M Rudianto. (2011). *Pemrograman Web Dinamis Menggunakan Php dan Mysql*. Yogyakarta: ANDI.
- Dewi, K. (2013). *Artificial Intelegence teknik dan aplikasinya* . Yogyakarta: Graha ilmu.
- Dikti, K. R. (2016, Mei Senin). *Visi, Misi & Strategi*. Diambil kembali dari Dikti.go.id: <http://dikti.go.id/visi-misi-strategi/>
- Dwi, P. (2010). *Pengolahan Citra Digital*. Yogyakarta: EDI Publisher.
- Eva Yulianti dan Riska Damayanti. (2015). Sistem pendukung keputusan untuk menentukan penerimaan beasiswa bagi siswa SMAN 9 Padang dengan menggunakan metode AHP (Analytical Hierarchy Process). *Jurnal TEKNOIF*, 21-28.
- Faroqi Adam, Agung Wahana, Muktar Isep. (2015). Sistem pendukung keputusan untuk menyeleksi calon siswa menggunakan metode Analytic hierarchy process. *Jurnal ISTEK* 6, 60-69.
- Hayatunnufus, A. (2010). Pendeteksi dan Verifikasi Tanda Tangan Menggunakan Metode Image Domain Spasial. *Jurnal Sistem Komputer*, 1-8.
- IMade dan Tineke. (2015). SISTEM PENDUKUNG KEPUTUSAN DENGAN MENGGUNAKAN METODE ANALYTICAL HIERARCHY PROCESS (AHP) UNTUK MENENTUKAN PERINGKAT SISWA BERDASARKAN NILAI HARD SKILL DAN SOFT SKILL. *Edu Komputika Journal*, 1-6.
- Iwan Rijayana dan Lirien Okirindho. (2012). Sistem pendukung keputusan pemilihan karyawan berprestasi berdasarkan kinerja menggunakan metode Analytical Hierarchy Process . *SemnasIF*, 48-53.
- Kusrini, M. (2007). *Konsep dan aplikasi sistem pendukung keputusan*. Yogyakarta: Andi.
- Munawar. (2005). *Pemodelan Visual dengan UML*. Yogyakarta: Graha Ilmu.
- muntasa, a. (2010). *konsep pengolahan citra digital dan ekstraksi fitur* . yogyakarta: Graha Ilmu.
- Nalwan, A. (2012). *Pengolahan Gambar Secara Digital*. Jakarta: PT Elex Media Komputindo.
- PUI, R. (2016, APRIL 5). *Profil Pusat Unggulan Iptek*. Dipetik APRIL 6, 2016, dari PUI Ristekdikti: <http://pui.ristek.go.id>
- Putra, D. (2009). *Sistem biometrika konsep dasar teknik analisis citra dan tahapan membangun aplikasi sistem biometrika*. jakarta: dharma publisher.
- Ristekdikti PUI. (2015). Pusat Unggulan Iptek. Dalam P. Ristekdikti, *Pusat Unggulan Iptek Unggul, Inovatif, Berdaya saing* (hal. 18). Jakarta: Direktorat Lembaga Litbang.
- Romeo. (2003). *Black Box Testing*. kansas.

-
- Setiawan, E. (2016, April 28). *Kamus Besar Bahasa Indonesia (KBBI)*. Dipetik April 28, 2016, dari kbbi.web.id: <http://kbbi.web.id/capai>
- Setiawan, E. (2016, April 28). *KBBI*. Dipetik April 28, 2016, dari kbbi.web.id: <http://kbbi.web.id/manfaat>
- SIRAIT, H. (2007). *KLASIFIKASI SIDIKJARI DENGAN PEMROSESAN AWAL . Pascasarjana Universitas Hasanuddin Makassar.*
- Sparague, W. (1993). *Decision Support System : Putting Theory Into Practice*. New Jersey: Englewood Clifts Prentice Hall.
- Sugiyono. (2013). *Metode Penelitian Kuantitatif, Kualitatif dan R&D*. Jakarta: Alfabeta.
- Suhendra, A. (2011). *Catatan Kuliah Pengantar pengolahan Citra*. Yogyakarta: Andi Publisher.
- Sutoyo. (2010). *Teori Pengolahan Citra Digital*. Yogyakarta: Andi Publisher.
- Tineke karunika dan I made Sudana. (2015). Sistem Pendukung Keputusan dengan menggunakan metode Analytical Hierarchy Process (AHP) untuk menentukan peringkat siswa berdasarkan nilai Hard skill dan Soft skill. *Edu Komputika Journal*, 1-6.
- Turban, E. A. (2001). *Decision Support Systems and Intelligent Systems.6th edition*. New Jersey: Upper Saddle River.
- Yustisia, D. (2010). Jaringan Syaraf Tiruan Learning Vector Quantization Untuk Aplikasi Pengenalan Tanda Tangan . *Seminar Nasional Aplikasi Teknologi Informasi*, 6-10.