

## PENGENALAN POLA TANDA TANGAN MENGGUNAKAN METODE JARINGAN SYARAF TIRUAN *BACKPROPAGATION*

Barry Ceasar Octariadi<sup>1)</sup>, Yulrio Brianorman<sup>2)</sup>

<sup>1,2)</sup>Teknik Informatika/Teknik, Universitas Muhammadiyah Pontianak

<sup>1,2)</sup>Jln. Jend. A. Yani No.111, Pontianak

Email: <sup>1</sup>barry.ceasaro@unmuhpnk.ac.id, <sup>2</sup>y.brianorman@unmuhpnk.ac.id

### Abstrak

Tanda-tangan merupakan fitur biometrik yang dapat digunakan untuk memverifikasi identitas seseorang. Penelitian ini menyajikan pengenalan pola tanda tangan menggunakan metode jaringan syaraf tiruan backpropagation (JST-BP) pada 100 contoh tanda-tangan. Dimana hasil pengenalan pola akan dibandingkan akurasi dengan metode Support Vector Machine (SVM). Menggunakan ekstraksi ciri vertical splitting, horizontal splitting untuk mendapatkan nilai sudut dan nilai jarak sebagai ciri dari tanda tangan yang selanjutnya akan di proses. Klasifikasi Jaringan saraf tiruan dengan metode pembelajaran jaringan saraf tiruan backpropagation (JST-BP). Jaringan saraf tiruan dengan metode pembelajaran Backpropagation yang digunakan terdiri dari 100 input, 3 hidden layer dan 2 output, fungsi pembelajarannya menggunakan resilient backpropagasi. JST-BP memiliki akurasi dalam memverifikasi sebesar 98.5% Pada tanda-tangan Traced, JST dengan metode backpropagation masih dapat memverifikasi dengan akurasi 82% sedangkan SVM memiliki akurasi dalam mengidentifikasi sebesar 94.5%. Pada tanda-tangan Traced, JST dengan metode backpropagation masih dapat mengidentifikasi dengan akurasi 74%, sedangkan untuk SVM tanda-tangan baru tidak bisa diidentifikasi dengan akurasi 51 %.

**Kata kunci** : Tanda-tangan, Jaringan saraf tiruan Backpropagation, vertical horizontal splitting, support vector machine

### 1. Pendahuluan

Tanda tangan (*signature*) adalah sebuah tanda (*sign*) atau symbol yang merupakan versi miniature dari pemiliknya. Tanda tangan bukanlah sembarang tanda atau symbol, tetapi merupakan symbol yang secara legal dan merupakan gambaran asli dari pemiliknya. Dalam kehidupan sehari-hari, tanda tangan digunakan sebagai identifikasi dari pemiliknya dan keberadaan tanda tangan dalam sebuah dokumen menyatakan bahwa pihak yang menandatangani, mengetahui dan menyetujui atau pengesahan seluruh isi dari dokumen. Pembubuhan tanda tangan sering dijumpai pada kegiatan administrasi instansi Pemerintahan maupun swasta, seperti pada perbankan, yaitu: transaksi penarikan uang secara tunai, penyeteroran, kliring giro dan transaksi lainnya. Pada saat mengenali

suatu tanda tangan secara visual memang cukup mudah untuk dilakukannya, dan jika banyak dokumen yang di validasi berdasarkan tanda tangan pihak yang berwenang maka hal ini akan membutuhkan waktu yang lebih banyak dan apabila dilakukan oleh satu orang, maka akan membingungkan karena harus mencocokkan setiap tanda tangan pada arsip yang satu dengan yang lainnya. Hal ini menarik perhatian peneliti untuk melakukan penelitian yang akan membuat suatu aplikasi identifikasi tanda tangan yang dapat dilakukan dengan menggunakan komputer, sehingga akan menghemat waktu jika dibandingkan secara manual. Pencocokan karakteristik tanda tangan dengan pemiliknya dapat dilakukan dengan menggunakan komputer, sehingga akan menghemat waktu jika dibandingkan dengan dilakukan secara manual. Dalam penelitian ini, pendekatan baru untuk identifikasi tanda tangan didasarkan pada Jaringan Saraf Tiruan Backpropagation (JST-BP). Backpropagation merupakan algoritma umum dari jaringan saraf tiruan. Pada proses pembelajaran algoritma Backpropagation termasuk kategori metode *supervised learning*. Metode pelatihan menggunakan algoritma ini dapat menghasilkan keseimbangan antara kemampuan jaringan untuk mengenali pola yang digunakan selama pelatihan serta kemampuan jaringan memberikan respon yang benar terhadap pola masukan serupa namun tidak sama dengan pola yang dipakai selama pelatihan.

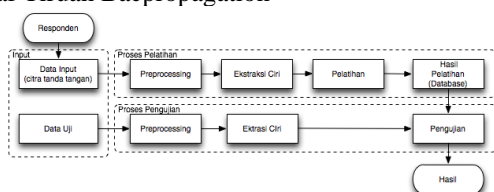
Dengan jaringan saraf tiruan Backpropagation ini, beberapa contoh tanda tangan akan diberikan untuk dilakukan proses pembelajaran atau pelatihan (*training*) agar nantinya JST ini dapat melakukan proses pengenalan ketika menerima input tanda tangan yang akan diuji (*testing*).

Pada penelitian sebelumnya yang dilakukan oleh [1] kinerja sistem identifikasi menggunakan Jaringan Saraf Tiruan Backpropagation pada pelatihan yang berbeda efisiensi tahapan ditemukan di atas lebih dari 95% keberhasilan. Dalam penelitian [2] untuk masalah identifikasi tanda tangan otomatis telah mendapat sedikit perhatian dibandingkan dengan masalah verifikasi tanda tangan manual. Penelitian ini menyajikan sebuah verifikasi tanda tangan seseorang dengan mencocokkan tanda tangan yang ada di database dengan menggunakan metode Backpropagation (BP) bahwa sistem aplikasi memiliki persentase 95% tingkat keberhasilan untuk mengidentifikasi tanda tangan dari pengujian data terlatih, sementara itu memiliki hanya 88% persentase

tingkat keberhasilan dari pengujian dari bagian luar database. Dan pada penelitian [3], *Backpropagation* (BP) digunakan untuk memverifikasi dan mengklasifikasikan tanda penelitian mengevaluasi sistem kinerja dan tingkat verifikasi 94,27%.

## 2. Metode Penelitian

Pengenalan tanda tangan dengan metode Jaringan Saraf Tiruan *Backpropagation*



Gambar 1. Rancangan Pengenalan Tanda Tangan

Pada rancangan sistem pengenalan tanda tangan Gambar 1. adalah gambaran umum dari proses yang akan dilakukan oleh Support Vector Machine dan Jaringan Saraf Tiruan *Backpropagation*. Yang dimulai dari

- Responden

Responden terdiri dari 100 orang responden yang memberikan 3 tanda tangan pada setiap orangnya. Setiap contoh tanda tangan dikoleksi pada kotak tepi dengan ukuran 7x4 cm, dengan menggunakan alat tulis dan menggunakan kertas putih. Setiap kertas berisikan tiga buah kotak contoh tanda tangan yang nantinya akan di scan dan disimpan agar contoh tanda tangan tersebut dapat digunakan untuk referensi selanjutnya.

- Data

Setelah di-scan, dengan menggunakan tools. Image yang terdapat pada kertas diolah terlebih dahulu dengan memotong gambar-gambar tersebut sehingga gambar dengan ukuran 300x100 pixel, kemudian gambar tersebut diperbaiki tingkat kontrasnya. Gambar-gambar tersebut nantinya merupakan input data dan data uji yang berupa citra tanda tangan.

- Proses Pelatihan

Setelah citra digital berupa citra tanda tangan telah siap di proses, citra akan dikenakan proses pra-pengolahan yang terdiri dari proses konversi rgb menjadi grayscale, lalu dilakukan penapisan menggunakan Median Filter untuk membuang derau *salt&pepper* yang timbul ketika proses *scan paper*. Masih dalam tahap pra-pengolahan, proses dilanjutkan dengan segmentasi menggunakan algoritma Otsu untuk mendapatkan bagian objek (*foreground*) yakni bagian tanda tangan, kegiatan ini akan menghasilkan citra biner. Kemudian proses dilanjutkan dengan penipisan (*thinning*) agar didapat tanda tangan yang lebarnya 1 piksel. Proses berikutnya yang masih merupakan bagian dari pra-pengolahan adalah *centering*, yakni proses yang menempatkan objek tepat di koordinat tengah dari citra. Proses ini didasarkan pada nilai *Center of Gravity* atau Pusat Titik Massa tanda tangan. Proses ini sangat penting untuk proses ekstraksi ciri karena merupakan koordinat referensi untuk mendapatkan ciri dari semua citra tanda tangan. Ukuran citra hasil pra-

pengolahan tidak berubah yakni 300 x 100 piksel.

Preprocessing akan menghasilkan citra yang telah bersih dari noise sehingga citra sesuai dengan yang diinginkan dan dilakukan proses ekstraksi ciri untuk mendapatkan fitur-fitur yang merupakan representasi dari setiap citra tanda tangan. Setiap citra tanda tangan akan memiliki vektor ciri berukuran 1 x 120. Setelah didapat fitur maka selanjutnya adalah proses pelatihan atau training dengan JST-BP.

- Proses Pengujian

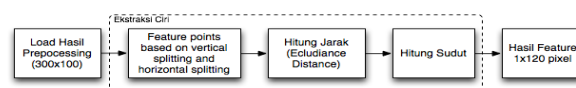
Setelah didapat hasil dari pelatihan atau training dengan Jaringan Saraf Tiruan *Backpropagation* yang berupa representasi dari setiap kelas, akan dilakukan pengujian dengan data uji. Sebelum dilakukan pengujian, data uji terlebih dahulu dipreprocessing dan selanjutnya dilakukan pengujian dengan Jaringan Saraf Tiruan *Backpropagation* dengan membandingkan hasil training dan data uji, diketahui data uji masuk di kelas yang mana dari hasil data training Jaringan Saraf Tiruan *Backpropagation* sebelumnya.

- Hasil

Setelah dilakukan proses preprocessing, ekstraksi ciri, pelatihan, pengujian akan didapat sebuah citra tanda tangan yang sesuai dengan hasil proses pelatihan dan proses pengujian pada Jaringan Saraf Tiruan *Backpropagation*

### 2.1 Rancangan Preprocessing

Proses Preprocessing adalah pemrosesan awal bertujuan untuk mengolah citra agar dapat diambil karakteristiknya. Tahap dalam pemrosesan awal dapat dilihat pada Gambar 2.



Gambar 2. Proses Preprocessing

Pertama yang dilakukan adalah membaca tanda tangan yang diambil dari bank Data Sampel Tanda Tangan. Yang seterusnya akan dikonversi menjadi gray scale.

Citra warna diubah menjadi citra gray scale format BMP berukuran 300x100 piksel. Tujuan pembuatan citra gray scale adalah menyederhanakan perhitungan dengan mengubah representasi nilai-nilai komponen RGB pada koordinat piksel, sebuah citra yang hanya mempunyai nilai derajat keabuan yang biasanya disebut citra *grayscale*. Proses ini dilakukan dengan persamaan (1) dengan matrik berdimensi  $M \times N \times 3$  menjadi  $M \times N \times 1$  [4].

$$\text{Grayscale} = 0,299R + 0,587G + 0,114B \quad (1)$$

Filter dengan Median Filter bertujuan untuk menghilangkan *noise* atau derau pada citra misalnya jenis *salt and pepper noise* atau bisa disebut dengan derau impulse. Dimana Median Filter ini dikenakan kepada piksel citra tanda tangan yang sudah di deteksi sebagai noise.

Segmentasi dengan menggunakan Metode Otsu Metode

Otsu ini didasarkan pada histogramnya. Histogram menunjukkan sebarang nilai intensitas dari tiap piksel pada citra dalam 1 dimensi. Jadi sumbu x biasanya menyatakan level intensitas yang berbeda sedangkan sumbu y menyatakan jumlah piksel yang memiliki nilai intensitas tersebut. Dengan menggunakan histogram kita bisa melakukan pengelompokan terhadap piksel-piksel dalam citra. Pengelompokan ini didasarkan pada nilai ambang batas atau *threshold*. Nilai *threshold* ini menjadi objektif atau tujuan dari metode Otsu. Dasar dari metode Otsu adalah perbedaan intensitas dari piksel-piksel yang dipisahkan dalam kelas-kelas tertentu. Sebuah *threshold* yang mampu memisahkan kelas-kelas sehingga piksel-piksel antar kelas memiliki nilai intensitas yang berbeda maka *threshold* tersebut dikatakan sudah optimal. Jadi tampak bahwa metode Otsu ini segmentasinya didasarkan pada nilai intensitas dari piksel-piksel pada citra.

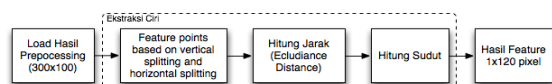
Penipisan citra (*Thinning*) yang dimaksudkan untuk mengubah citra menjadi hanya berukuran satu piksel saja. Pada proses ini data citra tetap dipertahankan keutuhan kurva citra dan sekaligus tidak mengubah ukuran citra dengan menghapuskan piksel-piksel yang terdapat pada ujung-ujung citra.

Setelah dilakukan *Thinning* dilakukan centering ketika kita menulis sebuah karakter yang akan digunakan sebagai inputan data, kadang kita lihat tulisan cenderung berada pada satu sisi tertentu. Untuk itu inputan data tersebut harus ditempatkan ditengah pada suatu form tertentu. Hal ini bertujuan untuk memfokuskan objek inputan data.

Setelah dilakukan proses *grayscale*, *median filter*, segmentasi dengan metode *otsu*, penipisan (*thinning*) dan terakhir dilakukan *centering* dihasilkan sebuah citra tanda tangan 2 dimensi dengan ukuran 300x100 pixel dengan format BMP dan disimpan ke database untuk selanjutnya dilakukan proses ekstraksi ciri.

Setelah didapat hasil *preprocessing* pada setiap tanda tangan dari semua proses dari *preprocessing* disimpan di data base dengan folder 'preproc\_input' dan 'preproc\_train'.

## 2. 2 Rancangan Proses Ekstraksi Ciri Tanda Tangan



Gambar 3. Proses Ekstraksi Ciri [5]

Pada Gambar 3 hasil dari *Preprocessing* sebelumnya yaitu proses *grayscale*, *median filter*, segmentasi dengan metode *otsu*, *Thinning* dan terakhir dilakukan centering dihasilkan sebuah citra tanda tangan 2 dimensi dengan ukuran 300x100 pixel dengan format BMP dan disimpan ke database untuk selanjutnya dilakukan proses ekstraksi ciri. Selanjutnya dilakukan proses ekstraksi ciri dengan berdasarkan 2 feature *Feature points based on vertical splitting dan horizontal splitting*, *Feature points based on vertical grid and horizontal grid*. Pertama dilakukan dengan me-load atau memanggil folder 'preproc\_input' dan 'preproc\_train', diproses ekstraksi ciri. Setekah didapat citra yang akan di ekstraksi , selanjutnya

ditemukan center of gravity yang mana citra ditempatkan di tengah atau center pada window yang telah ditentukan, agar mempermudah proses dalam ekstraksi ciri.

Titik tengah dari citra merupakan koordinat referensi yang akan digunakan untuk mendapatkan CoG (*Center of Gravity*) lanjutan yang akan digunakan sebagai bagian dari proses pengambilan ciri citra. Pada proses menemukan CoG berdasarkan *vertical splitting*, langkah awal adalah dengan membagi citra menjadi 2 bagian kiri dan kanan menggunakan garis vertikal (tegak) terhadap koordinat dari titik tengah citra. Proses menemukan CoG selanjutnya dilakukan dengan langkah – langkah dengan persamaan (2) dan (3)[5].

$$XCG = \frac{\sum_x W}{\sum W} \quad (2)$$

$$YCG = \frac{\sum_y W}{\sum W} \quad (3)$$

### 1. *Feature points based on vertical splitting.*

Pada proses *vertical splitting*, langkah awal menemukan CoG berdasarkan *vertical splitting* dengan membagi citra menjadi 2 bagian kiri dan kanan menggunakan garis vertikal (tegak) terhadap koordinat x dari titik tengah citra. Ini adalah prosedur untuk menghasilkan fitur poin berdasarkan pembelahan secara vertical, dengan menggunakan persamaan (2) dan (3). Pada ( $V_0$ ) ditandai sebagai titik tengah atau CoG dari citra tanda tangan.

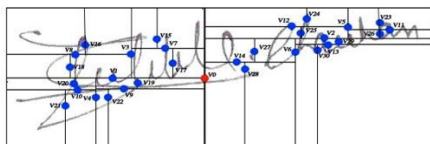
Ini adalah prosedur untuk menghasilkan fitur poin berdasarkan pemisahan vertical.

Input : Citra tanda tangan

Output : Fitur Vertikal poin:  $V_1, V_2, V_3, V_4, \dots, V_{29}, V_{30}$ .

Langkah-langkahnya adalah:

- Membagi gambar dengan garis vertikal melewati koordinat x, CoG ( $V_0$ ) dengan membagi gambar menjadi dua bagian yaitu bagian kiri dan bagian kanan.
- Cari CoG  $V_1$  dan  $V_2$  untuk kiri dan kanan secara bersamaan.
- Membagi bagian kiri dan kanan dengan garis horizontal melewati koordinat y melalui  $V_1$  dan  $V_2$  untuk membagi dua bagian menjadi empat bagian: Atas-kiri, bawah-kiri serta atas-kanan dan bawah-kanan yang mana memperoleh fitur poin  $V_3, V_4$  dan  $V_5, V_6$ .
- Dilakukan kembali membagi setiap bagian dari citra melalui CoG secara vertical melewati koordinat x untuk memperoleh fitur poin  $V_7, V_8, V_9, V_{10}, V_{11}, V_{12}, V_{14}$ .
- Lalu dilakukan kembali membagi setiap bagian sekali lagi secara horizontal melewati koordinat y untuk mendapatkan semua fitur poin  $V_{15}, V_{16}, V_{17}, V_{18}, V_{19}, V_{20}, V_{21}, V_{21}, V_{23}, V_{24}, V_{25}, V_{26}, V_{27}, V_{28}, V_{29}, V_{30}$ . Ditunjukkan pada Gambar 4.



Gambar 4. Feature points based on vertical splitting[6]

### 2. Feature points based on horizontal splitting

Mirip dengan *vertical splitting*, langkah awal dari *horizontal splitting* adalah dengan membagi citra menjadi 2 bagian atas dan bawah dengan menggunakan garis horizontal (mendatar) terhadap koordinat tengah. Ini adalah prosedur untuk menghasilkan fitur poin berdasarkan membelah secara horizontal. Dengan menggunakan persamaan (2) dan (3). H0 ditandai sebagai titik tengah atau CoG dari citra tanda tangan.

Proses menemukan CoG selanjutnya dilakukan dengan langkah – langkah dengan persamaan (2) dan (3). Dijelaskan x dan y adalah jumlah koordinat dan w adalah jumlah yang hitam pada sebuah citra tandatangan. Proses CoG akan digunakan setiap menentukan titik pusat(CoG) apakah itu secara vertical(yCG) maupun horizontal (xCG).

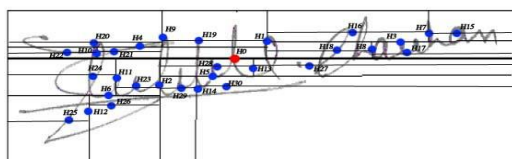
Ini adalah prosedur untuk menghasilkan fitur poin berdasarkan pemisahan horizontal.

Input : Gambar tanda tangan

Output : Fitur Horizontal poin: H<sub>1</sub>, H<sub>2</sub>, H<sub>3</sub>, H<sub>4</sub>, ..... H<sub>29</sub>, H<sub>30</sub>.

Langkah-langkahnya adalah:

- Membagi citra secara horizontal melewati koordinat y melalui CoG (H<sub>0</sub>) yang membagi citra menjadi dua bagian yaitu bagian atas dan bagian bawah.
- Cari CoG dari H<sub>1</sub> dan H<sub>2</sub> untuk bagian atas dan secara bersamaan.
- Membagi bagian atas dan bawah dengan garis vertikal melewati koordinat x dengan melalui H<sub>1</sub> dan H<sub>2</sub> untuk membagi dua bagian menjadi empat bagian: kiri-atas, kanan-atas serta kiri-bawah dan bagian kanan-bawah yang mana memperoleh fitur poin H<sub>3</sub>, H<sub>4</sub> dan H<sub>5</sub>, H<sub>6</sub>.
- Dilakukan kembali membagi setiap bagian dari citra melalui CoG secara horizontal melewati koordinat y untuk memperoleh Fitur poin H<sub>7</sub>, H<sub>8</sub>, H<sub>9</sub>, H<sub>10</sub>, H<sub>11</sub>, H<sub>12</sub>, H<sub>13</sub>, H<sub>14</sub>.
- Lalu dilakukan kembali membagi setiap bagian sekali lagi secara vertical melewati koordinat x untuk mendapatkan fitur poin H<sub>15</sub>, H<sub>16</sub>, H<sub>17</sub>, H<sub>18</sub>, H<sub>19</sub>, H<sub>20</sub>, H<sub>21</sub>, H<sub>22</sub>, H<sub>23</sub>, H<sub>24</sub>, H<sub>25</sub>, H<sub>26</sub>, H<sub>27</sub>, H<sub>28</sub>, H<sub>29</sub>, H<sub>30</sub>. Ditunjukkan pada Gambar 5.



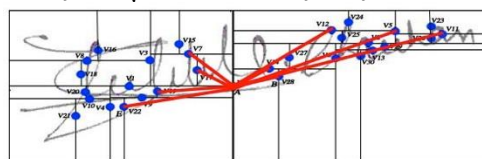
Gambar 5. Feature points based on horizontal splitting([6])

### 3. Hitung Jarak (Euclidean distance)

Setelah didapat hasil dari *Feature points based on vertical*

*splitting dan horizontal splitting* sebanyak 60 vektor lalu dihitung jarak antar titik – titik tersebut terhadap titik tengah citra menggunakan perhitungan jarak *Euclidean distance* yang ditunjukkan oleh persamaan (4). Nilai perhitungan jarak tersebut akan dijadikan ciri pertama dari tanda tangan[6]. Dapat dilihat pada Gambar 6.

$$d(x, y) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (4)$$



Gambar 6. Feature points based on vertical grid and horizontal grid[6]

### 4. Hitung Sudut

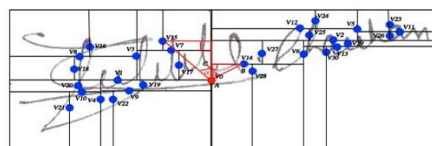
Ciri selanjutnya didapat dengan menghitung sudut yang terbentuk antara titik tengah dengan titik – titik CoG yang diperoleh melalui proses *vertical splitting dan horizontal splitting*. Ciri ini disebut dengan *vertical grid dan horizontal grid*. Rumus menghitung sudut ditunjukkan oleh persamaan (5),(6),(7)[6]. Ditunjukkan pada Gambar 7.

$$\theta_i = \cos^{-1} \left( \frac{AB}{AC} \right) \quad (5)$$

$$AB = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (6)$$

$$AC = y_2 - y_1 \quad (7)$$

$$AC = x_2 - x_1 \quad (8)$$



Gambar 7. Feature points based on vertical grid and horizontal grid[6]

Langkah dalam mendapatkan nilai sudut, hanya mengkalkulasi jarak AB dan AC. Panjang AB dihitung dengan jarak Ecludcean. Panjang AC dihitung dengan mencari titik koordinat C terlebih dahulu. Pencarian titik koordinat C berbasis pada titik koordinat B. Titik koordinat x pada titik B diganti dengan 0 sehingga menjadi (0,y) pada *vertical splitting*. Kemudian, pencarian titik koordinat C juga berbasis pada titik koordinat B untuk *horizontal splitting*. Koordinat y pada titik C diganti dengan 0 sehingga menjadi (x,0). Untuk mendapatkan jarak AC dirumuskan seperti persamaan (7) untuk *vertical splitting dan Jarak AC untuk horizontal dirumuskan seperti persamaan (8)*.

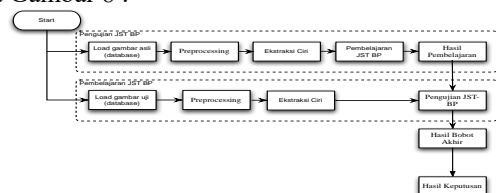
Langkah pencarian nilai sudut itu akan menghasilkan ciri kedua dengan dimensi 1 x 60. Dengan demikian, vektor ciri yang dihasilkan berukuran 1 x 120. Selanjutnya vektor tersebut akan digunakan untuk klasifikasi dari ciri masing-masing setiap citra tanda tangan. Ciri yang didapat akan disusun ulang sebelum dimasukkan ke dalam database. Bentuk penyusunannya adalah dengan meletakkan secara selang-seling 60 nilai pertama, yaitu hasil perhitungan



jarak ecludiean berdasarkan *vertical splitting* ditempatkan pada urutan ganjil sedangkan hasil dari *horizontal splitting* pada urutan genap, ( $Ve, He$ ). Untuk 60 nilai kedua, ciri *vertical grid* (sudut yang diperoleh antara titik tengah dan titik-titik *vertical splitting*) diletakkan pada urutan ganjil seperti 61, 63, ..., 119. Sedangkan cirri *horizontal grid* (sudut yang diperoleh antara titik tengah dan titik – titik *horizontal splitting*) diletakkan pada urutan genap seperti 62, 64, ..., 120. Sehingga urutan untuk ciri kedua ini berbentuk ( $Vg, Hg$ ). Bentuk akhir dari semua ciri yang telah disusun ulang adalah  $[Ve_1, He_1, Ve_2, He_2, Ve_3, He_3 \dots \dots Ve/He_n]$  dan  $[Vg_1, Hg_1, Vg_2, Hg_2, Vg_3, Hg_3 \dots \dots Vg/hg_n]$  dengan total 120 ciri. Yang selanjutnya akan digunakan untuk klasifikasi dari ciri masing-masing setiap citra tanda tangan. Setelah diproses ekstrasi ciri dari setiap tanda tangan akan disimpan dalam data base dengan nama 'ciri\_input' dan 'ciri\_train' dengan format .math.

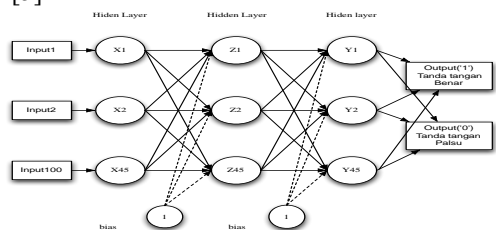
### 2. 3. Rancangan Proses Jaringan Saraf Tiruan Backpropagation

Sistem Identifikasi tanda tangan yang dibuat memiliki beberapa proses yang akan dijalankan ketika program berjalan, adapun yang akan berjalan pada sistem adalah proses preprocessing, proses pembelajaran, proses pengujian, proses postprocessing. Proses identifikasi tanda tangan dengan JST Backpropagation ini ditunjukkan pada Gambar 8 .



Gambar 8. Rancangan Proses Jaringan Saraf Tiruan bacpropagation

Pelatihan propagasi balik meliputi 3 tahap. Tahap pertama adalah tahap maju, dimana pola masukan dihitung maju mulai dari lapis masukan hingga lapis keluaran menggunakan fungsi aktivasi yang ditentukan. Tahap kedua adalah tahap mundur, yaitu selisih antara keluaran jaring dengan target yang diinginkan merupakan kesalahan yang terjadi. Kesalahan tersebut dipropagasikan mundur, dimulai dari garis yang berhubungan langsung dengan unit-unit di lapis keluaran. Tahap ketiga adalah modifikasi bobot untuk menurunkan kesalahan yang terjadi[5].



Gambar 9. Arsitektur Propagasi Balik[7]

Pada Gambar 9 Arsitektur Jaringan Syarat Tiruan (JST)

yang digunakan pada aplikasi ini adalah JST *Backpropagation*. BP yang didesain merupakan BP standard tanpa momentum. Pada Gambar 9 merupakan model Jaringan Saraf Tiruan *Backpropagation* ini *neuron* diorganisasikan ke dalam struktur data yang kita sebut *layer* (lapisan), *layer* yang ada pada Jaringan Saraf Tiruan *Backpropagation* yang digunakan menggunakan lapisan banyak (*multilayer*), terdiri atas 3 yaitu *input* , 3 lunit tersembunyi (*hidden layer*), dan 1 *output* juga terdapat bias yang diberikan pada unit-uniti tersembunyi dan output Input layer (lapisan masukan) terdiri atas 100 variabel input sel. Jumlah sel hidden layer (lapisan tersembunyi) 3 dengan jumlah *node* pada hidden layer sebanyak 45 *node*. Jumlah *node* pada hiden layer ditentukan dengan cara *trial and error*; dimana jumlah *node* yang terlalu sedikit menyebabkan proses pelatihan tikan akan menghasilkan bobot yang stabil, namun jumlah *node* yang terlalu banyak akan menyebabkan proses pelatihan menjadi lebih lambat, dalam arti hasil pembelajaran yang tercepat dan terbaik itu lah akan yang menentukan jumlah *node* pada *hiden layer* tersebut. , sedangkan output layer(lapisan keluaran) terdiri dari 1 unit sel.

Pada proses *backpropagation* ini mempunyai dua prose penting yaitu *feed forward* dan *backward*. *Feed Forward* digunakan dalam proses *training* yang mana dilakukan perhitungan aktivasi yang didapat dari perkalian nilai input dan *weight*(bobot). Hasil dari nilai aktivasi dijadikan nilai input oleh layer yang berada didepanya. Sedangkan *backward* pada proses ini dilakukan perubahan nilai *weight*(bobot) supaya nilai *output* yang dihasilkan sesuai keinginan. Selain perubahan bobot, pada proses ini juga dilakukan perhitungan nilai *error*. Nilai *error* ini merepresentasikan tingkat kesalahan yang dari *backpropagation*.

### 3. Hasil dan Pembahasan

Dalam pembuatan aplikasi identifikasi tanda tangan dibagi menjadi beberapa tahap, antara lain:

#### 1. Tahap preprocessing

Pada tahap ini citra tanda tangan yang telah disesuaikan ukurannya dalam hal ini 100x300 piksel kemudian citra di proses konversi rgb to gray, Median Filter, Metode Otsu, Thinning dan centering .Proses preprocessing ini bertujuan untuk mendapatkan hasil citra yang lebih baik dari tanda tangan yang digunakan dalam proses ekstrasi ciri, pelatihan dan pengujian, ditunjukkan proses prepeocsing pada Table 1.

Tabel 1 Hasil Preprocessing Citra Tanda Tangan


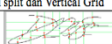

RGB to Gray scale (100x300)	Median Filter	Segmentasi Metode Otsu	Thinning	Centering

#### 2. Tahap ekstrasi ciri

Pada tahap ini dilakukan setelah proses *Preprocessing* , ekstrasi ciri bertujuan untuk mendapatkan informasi-informasi penting dari pola suatu citra. Proses ekstrasi ciri dengan berdasarkan 2 feature *Feature points based on*

vertical splitting dan horizontal splitting, Feature points based on vertical grid and horizontal grid, ditunjukkan pada Tabel 2.

Tabel 2 Hasil Ekstraksi Ciri

Center of gravity	Vertical split dan Vertical Grid	Horizontal split dan Horizontal grid
		

### 3. Tahap pelatihan dan pengujian

Pada pelatihan citra tanda tangan ini dilakukan dengan data pelatihan menggunakan 100 data untuk input dan 100 data untuk pelatihan yang mewakili setiap sampel tanda tangan. Dan hasil pelatihan membentuk jaringan Backpropagation.

Pada pelatihan jaringan saraf tiruan backpropagation, penelitian ini penulis melakukan *trial and error* untuk mendapatkan nilai performance yang baik dengan mendekati nilai goal 0.0001, ditunjukkan pada Tabel 3.

Tabel 3 Hasil uji coba

No	Input	Node Hidden Layer 1	Node Hidden Layer 2	Node Hidden Layer 2	Epoch	Performance
1	100	15	20	15	100	0.000818
2	100	20	20	20	100	0.000559
3	100	30	35	35	100	0.000530
4	100	30	40	30	100	0.000781
5	100	30	30	30	100	0.000274
6	100	35	35	35	100	0.000215
7	100	35	40	35	100	0.000241
8	100	40	40	40	100	0.000230
9	100	40	45	40	100	0.000195
10	100	45	45	45	100	0.000955

Pada Tabel 3, jumlah *node* pada hidden layer ditentukan dengan cara *trial and error*; dimana jumlah *node* yang terlalu sedikit menyebabkan proses pelatihan tidak akan menghasilkan bobot yang stabil, namun jumlah *node* yang terlalu banyak akan menyebabkan proses pelatihan menjadi lambat, dalam arti hasil pembelajaran yang tercepat dan terbaik itu lah akan yang menentukan jumlah *node* pada *hidden layer* tersebut. Disini penulis memilih dengan jumlah *node* pada *hidden layer 1* dengan jumlah 40 *node*, *hidden layer 2* dengan jumlah 45 *node* dan *hidden layer 3* dengan jumlah 40 *node*. Demikian juga jumlah *epoch* 100 hingga mendapat nilai *performance* 0.000195 dengan tempuh lama waktu pembelajaran 0:07:49.

Pada pengujian ini dilakukan dengan menggunakan 100 tanda tangan pada setiap citra tanda tangan yang dilatih. Proses pengujian ini dilakukan untuk melihat identifikasi dari citra tanda tangan tersebut dengan bermacam-macam sample. Hasil dari pengujian adalah apakah citra tanda tangan dikenali sesuai dengan kelasnya sebagai kelas lain yang terdapat dalam jaringan syaraf tiruan Backpropagation.

Tabel 4 Hasil Pengujian Jst-Backpropagation (JST-BP) dan Support Vector Machine (SVM)

Banyak User	Banyak TTD	Support Vector Machine		JST Bacpropagation	
		Berhasil	Tidak	Berhasil	Tidak
100	200	189	11	197	3

Untuk mendapatkan tingkat akurasi pengenalan dalam setiap pengujian Jaringan Saraf Tiruan *Backpropagation*

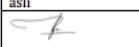

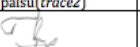
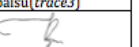
jaringan saraf tiruan *backpropagation* digunakan rumus pada persamaan(8):

$$\frac{\Sigma \text{Benar}}{\Sigma \text{Data Uji}} \times 100\% \quad (9)$$

Dimana jumlah data citra yang berhasil dikenali dengan tepat di bagi dengan jumlah data citra tanda tangan yang diuji dikalikan 100% akan mendapatkan hasil akurasi. Berdasarkan Tabel 4. pengujian JST dan SVM diatas, di uji setiap user 2 sampel tanda tangan dimana jumlah user nya 100 user yang memberikan tanda tangan, penerapan JST dengan metode backpropagation memiliki rata-rata dalam mengidentifikasi 197 data yang berhasil dikenali dari 200 sampel tanda tangan uji sehingga akurasinya sebesar 98.5% sedangkan SVM memiliki rata-rata dalam mengidentifikasi mengidentifikasi 189 data yang berhasil dikenali dari 200 sampel tanda tangan uji sehingga akurasinya sebesar 94.5%. Berdasarkan nilai rata-rata error, JST dengan metode backpropagation memiliki kinerja yang lebih baik dibandingkan dengan SVM untuk mengidentifikasi tanda-tangan yang diujikan, akan tetapi waktu yang diperlukan JST dengan metode backpropagation untuk proses pembelajaran lebih lama bila dibandingkan dengan SVM.

Tanda tangan yang akan dikenali diperoleh dari beberapa jenis tanda tangan dari orang yang berbeda, dan kemudian menguji jaringan maka tanda di buat beberapa variasi diantaranya dengan cara di *trace1* (meniru tanda tangan asli dengan tanda tangan tiruan atau tidak asli), *trace2*(meniru tanda tangan asli dan menambahkan pola garis pada tanda tangan asli), *trace3*(meniru tanda tangan asli dan menghilangkan pola garis pada tanda tangan asli).Seperti terlihat pada Tabel 5.

Tabel 5 Sampel Tanda Tangan Pengujian

Tanda tangan uji asli	Tanda tangan uji palsu( <i>trace</i> )	Tanda tangan uji palsu( <i>trace2</i> )	Tanda tangan uji palsu( <i>trace3</i> )
			

Hasil pengujian tanda-tangan untuk sampel yang tidak diketahui atau tidak dilatihkan dalam proses pembelajaran baik menggunakan metode JST dan SVM pada sistem identifikasi tanda-tangan dapat dilihat pada Tabel 6.

Tabel 6 Hasil Uji Coba Tanda Tangan JST-BP dan SVM

Banyak User	Banyak TTD	Support Vector Machine		JST Bacpropagation	
		Berhasil	Tidak	Berhasil	Tidak
13	39	32	7	19	10

Berdasarkan Tabel 6 hasil pengujian tanda-tangan untuk sampel yang tidak diketahui atau tidak dilatihkan, terlihat bahwa dengan menggunakan metode Jaringan Saraf Tiruan *Backpropagation* tanda tangan baru masih bisa di identifikasi memiliki rata-rata dalam mengidentifikasi 32 data yang berhasil dikenali dari 39 sampel tanda tangan uji sehingga akurasinya sebesar 82% sedangkan untuk SVM, tanda tangan baru tidak bisa

diidentifikasi dengan baik dengan rata-rata dalam mengidentifikasi 19 data yang berhasil dikenali dari 39 sampel tanda tangan uji sehingga akurasi sebesar 48%. penulis dan editor akan dilakukan melalui Open Journal System dan email.

#### Daftar Pustaka

- [1]. Arora, A.Choubey, A.S., Offline Signature Verification and Recognition using Neural Network, International Journal of Science and Research (IJSR),India , Agustus 2013
- [2]. Hidayanto, A., Isnanto, R.R., Buana, D.K.W., 2008, *Identifikasi Tanda-Tangan Menggunakan Jaringan Saraf Tiruan Perambatan - Balik(Backpropagation)* , Jurnal Teknologi , Vol. 1, No. 2, Desember 2008
- [3]. Sisodia, K. Anand, S.M, Offline Handwritten Signature Verification Using Artificial Neural Network Classifier, International Journal of Recent Trends in Engineering, Vol2,No.2,November 2009
- [4]. Thiang, 2005, "*Pengenalan Huruf Braille dengan Menggunakan Hamming Network*", Prosiding Seminar Nasional Soft Computing, Intelligent System and Information Technology , Surabaya, Universitas Kristen Petra.
- [5]. Chauhan, M.S., Nath, D.,2007, *Offline Signature Verification Scheme using Feature Extraction Method*, Computer Science and Engineering,2007.
- [6]. Raja, K.B.,Prashanth, C.R., *Offline Signature Verification Based on Angular Features*, International Journal of Modeling Optimization,Vol2,No.4, August 2012.
- [7]. Kusumadewi, S, *Artificial Intelligence (Teknik dan Aplikasinya)*, Graha Ilmu,Yogyakarta, 2003.
- [8]. Hsu, Chih-Wei., Chih-Jen Lin., 2002. "*A Comparison Methods for Multi-class Support Vector Machines*". IEEE Transactions on Neural Networks, Vol.13, No.2, 415-425.