

PENGENALAN TANDA TANGAN MENGUNAKAN METODE MOMENT INVARIANT DAN RESTRICTED BOLTZMANN MACHINE

Hamdan Abdul Majid¹, Kania Evita Dewi²

^{1,2}Teknik Informatika – Universitas Komputer Indonesia

Jl. Dipatiukur 112-114 Bandung

E-mail : hamdanabdoel@gmail.com¹, kania.evita.dewi@email.unikom.ac.id²

ABSTRAK

Penelitian pengenalan tanda tangan ini bertujuan untuk dapat dilakukan penerapan metode ekstraksi ciri *Moment Invariant* dan metode klasifikasi yaitu *Restricted Boltzmann Machine* serta mengetahui tingkat akurasi dalam pengenalan tanda tangan. Sebelum masuk pada tahap ekstraksi fitur terdapat pemrosesan *Preprocessing* yang terdiri dari *Cropping, Resize, Nilai RGB, Grayscale, Binerisasi Citra*, dan ekstraksi citra dengan moment invariant. Kemudian pada tahap selanjutnya dilakukan proses pelatihan dan pengujian dengan menggunakan metode *Restricted Boltzmann Machine*. Dalam penelitian ini digunakan sebanyak 150 tanda tangan dari 10 responden dimana setiap sesponden diambil 15 data tanda tangan. Untuk pelatihan menggunakan 120 tanda tangan dimana dari setiap respondennya diambil 12 tanda tangan, kemudian dalam pengujian data yang diuji sebanyak 30 tanda tangan yang didapatkan dari setiap respondennya sebanyak 3 tanda tangan. Berdasarkan hasil pengujian yang dilakukan terhadap data latih dan data uji menggunakan metode *Rule Based* dalam pengujian akurasi, maka data uji yang diperoleh sebesar 90%.

Kata kunci : tanda tangan, *machine learning*, metode *moment invariant*, *restricted boltzmann machine*.

1. PENDAHULUAN

Tanda Tangan atau dalam Bahasa Inggris *signature* yang juga berasal dari kata latin *signare* yang berarti “tanda” atau paraf merupakan tulisan tangan atau goresan tinta dari tangan, kadang-kadang diberi gaya tulisan tertentu dari nama seseorang atau tanda identifikasi lainnya yang ditulis pada dokumen sebagai suatu bukti dari identitas dan kemauan[1]. Contoh-contoh tanda tangan setiap orang umumnya identik namun tidak sama, artinya tanda tangan seseorang sering berubah-ubah setiap waktu. Perubahan ini menyangkut posisi, ukuran maupun faktor tekanan tanda tangan. Pada kenyataannya, perubahan-perubahan tersebut dipengaruhi oleh waktu, umur, kebiasaan dan keadaan mental tertentu [2].

Karenanya perlu dilakukan pengenalan tanda tangan secara otomatis. Pengenalan tanda tangan dapat dilakukan dengan menggunakan pemrosesan citra dan metode *machine learning*. Keberhasilan pengenalan tanda tangan dipengaruhi oleh proses ekstraksi ciri yang dilakukan dan pemilihan metode *machine learning* untuk mengenali citra gambar.

2. ISI PENELITIAN

2.1 Penelitian Terkait

Berdasarkan penelitian yang serupa sebelumnya pernah dilakukan oleh oleh Ainun Jariah, dkk. Pengenalan pola tanda tangan menggunakan metode moment invariant dan jaringan syaraf radial basis function (RBF) dapat mengenali pola tanda tangan dengan akurasi 80%. Metode moment invariant digunakan untuk mengekstraksi citra tanda tangan menjadi suatu vektor input yang merepresentasikan setiap citra tanda tangan [3].

Penelitian terkait yang pernah dilakukan juga oleh Susilawati dengan judul *Algoritma Restricted Boltzmann Machines (RBM)* untuk pengenalan tulisan tangan angka, pada penelitian tersebut hasil yang diperoleh dalam penentuan parameter *learning rate*, momentum dan fungsi aktivasi pada jaringan RBM sangat berpengaruh pada kinerja RBM. Dari hasil penelitian tersebut *learning rate* 0.05 dengan momentum 0.7 memiliki kinerja yang paling tinggi, mampu mengenali dataset pengujian sebesar 93,42% [4].

Berdasarkan permasalahan yang sudah dipaparkan sebelumnya maka solusi yang dapat digunakan untuk mengatasi masalah ini adalah dapat mengenali tanda tangan dengan metode moment invariant sebagai ekstraksi dan *Restricted Boltzmann Machines (RBM)* sebagai klasifikasinya. Kemudian dapat melihat hasil akurasi yang diperoleh.

2.2 Citra Digital

Citra Digital merupakan gambar dua dimensi yang bisa ditampilkan pada layar computer sebagai himpunan atau diskrit nilai digital yang disebut *pixel/picture elements*. Dalam tinjauan matematis, citra merupakan fungsi kontinu dari intensitas cahaya pada bidang dua dimensi [5]. Foto adalah contoh gambar

berdimensi dua yang dapat diolah dengan mudah. Setiap foto dalam bentuk citra digital dapat diolah menggunakan perangkat lunak tertentu. Dalam penelitian ini pengolahan citra yang dilakukan diantaranya cropping, ekstraksi nilai RGB, *grayscale*, dan ekstraksi fitur.

2.3 Data Masukkan

Data masukan pada penelitian ini yang akan diolah RBM adalah pengambilan gambar tanda tangan pada media kertas, kemudian dilakukan proses *scanning*, hasil scan gambar tanda tangan yang tersimpan dalam bentuk file .jpg. Dalam analisis ini jumlah yang di ambil dari 10 koresponden berjumlah 150 tanda tangan, setiap responden diambil 15 tanda tangan. Berikut adalah contoh citra tanda tangan yang telah dikumpulkan dari koresponden memiliki ukuran sebesar 1000x750 piksel seperti gambar 2.1 di bawah ini.

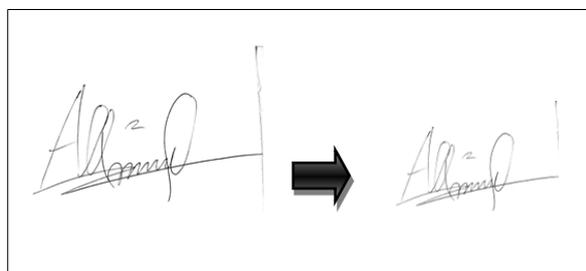


Gambar 1. Data Masukkan

Data tersebut akan dibagi menjadi data latih dan data uji dengan perbandingan 80:20 sehingga terdapat 12 data sebagai data latih dan 3 data sebagai data uji.

2.4 Cropping

Pada tahapan cropping citra tanda tangan dengan ukuran 1000x750 piksel kemudian diatur lebar dan tinggi citra tanda tangan terus citra akan di resize dengan ukuran 200x200 piksel



Gambar 2 Citra Cropping

Gambar di atas telah dilakukan resize dengan ukuran 200x200pixel.

2.5 Ekstraksi Nilai RGB

Seperti diketahui bahwa tiap citra yang berwarna memiliki nilai indeks warna RGB yang bermacam-macam. Perbedaan presentase indeks RGB membuat

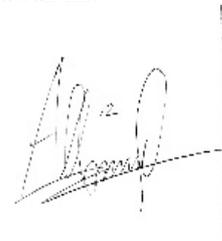
suatu citra menjadi berwarna merah, hijau, biru. Semakin tinggi indeks warnanya, maka citra tersebut akan semakin terang. Begitu pula sebaliknya, semakin kecil nilai indeks warnanya, maka citra akan semakin gelap. Dalam tahap RGB ini, selanjutnya citra tanda tangan ditulis ke dalam bentuk matriks warna RGB(Red Green Blue) dengan tujuan untuk dapat memperlihatkan nilai-nilai yang dihasilkan dari setiap proses. Citra masukan tersebut berukuran 200x200 piksel yang telah *resize*, matriks tersebut dapat dilihat dan disimbolkan dengan *rgb*.

2.6 Grayscale

Grayscale digunakan untuk mempermudah proses lanjutan dengan cara menyederhanakan nilai-nilai pada matriks citra ke rentang nilai warna putih dan abu-abu. Proses grayscale dilakukan untuk mengubah intensitas pixel. Proses ini menghasilkan citra dengan warna biner yaitu hitam dan putih[1]. Pada grayscale nilai intensitas tersebut dapat diseragamkan dengan suatu fungsi. Berikut ini adalah rumus konversi citra berwarna (RGB) menjadi nilai intensitas grayscale [7].

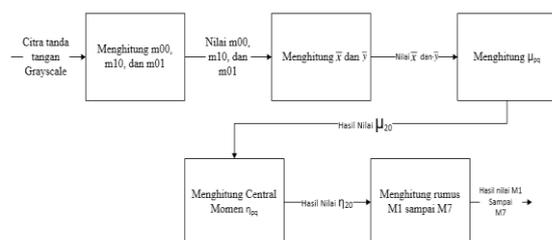
$$G = wR.Red + wG.Green + wB.Blue \quad (2.6.1)$$

Nilai wR , wG , dan wB masing-masing adalah bobot untuk elemen warna Red, Green dan Blue sesuai pendefinisian bobot untuk konversi citra warna ke skala keabuan dimana $wR = 0.299$, $wG = 0.587$ dan $wB = 0.114$. Nilai grayscale yang didapat akan menggantikan nilai RGB pada setiap piksel citra. Hasil dari matriks yang dihitung dari persamaan grayscale merubah citra seperti yang dapat dilihat pada gambar 3 di bawah ini.



Gambar 3 Citra Grayscale

2.7 Ekstraksi Citra Moment Invariant



Gambar 4 Alur Moment Invariant

Ekstraksi Ciri merupakan tahapan mengekstrak ciri/informasi dari objek di dalam citra

yang ingin dikenali/dibedakan dengan objek lainnya. Ciri yang telah diekstrak kemudian digunakan sebagai parameter/nilai masukan untuk membedakan antara objek satu dengan lainnya pada tahapan identifikasi/klasifikasi. Moment invariant ini digunakan sebagai fitur ekstraksi citra. Moment dapat memberikan karakteristik suatu objek yang secara unik merepresentasikan bentuknya. Pengenalan bentuk invariant dilakukan dengan klasifikasi dalam ruang fitur moment invariant multi dimensi. Beberapa teknik telah dikembangkan tentang penurunan fitur invariant dari moment objek untuk representasi dan pengenalan objek. Teknik ini dibedakan oleh definisi momentnya. Seperti jenis data yang dieksplotasi dan metode untuk menurunkan nilai invariant dari moment citra [8].

Moment Invariant terdapat 7 nilai descriptor bentuk, yang dihitung dari moment pusat melalui tiga derajat yang bebas terhadap translasi, skala dan arah objek [8]. Invariant translasi dicapai dengan menghitung moment yang dinormalisasi dengan pusat gravitasi sehingga pusat dari masa distribusi berada pada moment pusat. Moment Invariant ukuran diturunkan dari invariant aljabar tapi moment ini dapat diperlihatkan dari hasil penyederhanaan momen ukuran. Dari nilai order dua dan tiga moment pusat yang ternormalisasi, 7 moment invariant dapat dihitung yang juga bebas rotasi. Moment digunakan untuk membentuk moment invariant yang didefinisikan secara kontinu namun untuk implementasi praktis, moment dihitung secara diskrit.

Langkah pertama adalah menghitung central momen sesuai dengan parameter p dan q tertentu. Jika melihat rumus untuk fitur pertama:

$$M_1 = (\eta_{20} + \eta_{02}) \quad (2.4.1)$$

Untuk dapat menghitung rumus di atas terlebih dahulu untuk menghitung m_{00} , m_{01} , dan m_{10} dengan menggunakan rumus dibawah ini:

$$m_{pq} = \sum_{x=0}^{x=M-1} \sum_{y=0}^{y=M-1} (x)^p \cdot (y)^q f(x,y)$$

Pada rumus diatas masukkan nilai parameter m_{00} , m_{01} dan m_{10} kemudian dihitung dengan memasukkan nilai $p=0$ dan $q=0$. Setelah nilai m_{00} , m_{10} , m_{01} diketahui langkah selanjutnya menghitung nilai \bar{x} dan \bar{y} dengan rumus dibawah ini:

$$\bar{x} = \frac{m_{10}}{m_{00}} \text{ and } \bar{y} = \frac{m_{01}}{m_{00}}$$

Kemudian setelah \bar{x} dan \bar{y} di dapatkan maka selanjutnya menghitung μ_{pq} pada rumus dibawah ini:

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p \cdot (y - \bar{y})^q f(x,y)$$

Nilai p dan q yang digunakan adalah 2 dan 0, setelah μ_{pq} telah didapatkan maka γ dan η_{pq} dapat dihitung. Dengan rumus dibawah ini:

$$\gamma = \left\lfloor \frac{p+q}{2} \right\rfloor + 1 \quad \eta_{pq} = \frac{\mu_{pq}}{\mu^{\gamma_{00}}}$$

Kemudian lakukan perhitungan dengan cara yang sama tetapi dengan menggunakan $p=0$ dan $q=2$. Setelah mengetahui cara perhitungan central moment

maka perhitungan selanjutnya adalah menghitung moment fitur M1 sampai dengan M7 dengan menggunakan rumus berikut ini:

$$\phi_1 = \eta_{20} + \eta_{02}$$

$$\phi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2$$

$$\phi_3 = (\eta_{30} - 3\eta_{12})^2 + (\eta_{03} - 3\eta_{21})^2$$

$$\phi_4 = (\eta_{30} - \eta_{12})^2 - (\eta_{03} + \eta_{21})^2$$

$$\begin{aligned} \phi_5 = & (3\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 \\ & - 3(\eta_{21} + \eta_{03})^2] + (\eta_{21} \\ & - \eta_{03})(\eta_{30} + \eta_{12})x[3(\eta_{30} \\ & + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \end{aligned}$$

$$\begin{aligned} \phi_6 = & (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2 \\ & + 4\eta_{11}(\eta_{30} - \eta_{12})(\eta_{21} + \eta_{03}) \end{aligned}$$

$$\begin{aligned} \phi_7 = & (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 \\ & - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{12} \\ & - \eta_{30})(\eta_{21} + \eta_{03})x[3(\eta_{30} \\ & + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \end{aligned}$$

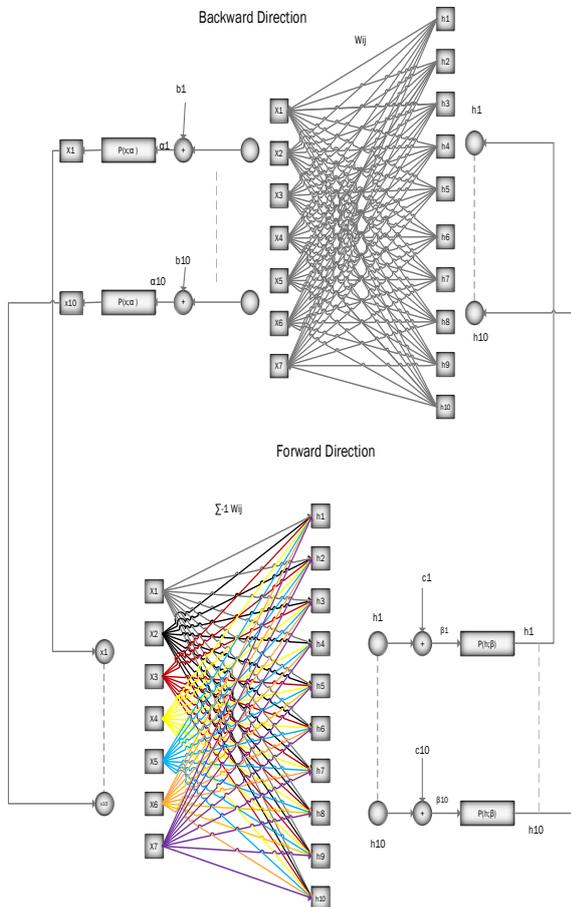
Dengan melakukan hal yang sama pada keseluruhan citra tanda tangan. Hasil ekstraksi fitur akan digunakan sebagai data input dalam algoritma berikutnya, yaitu algoritma Restricted Boltzmann Machine.

2.8 Restricted Boltzmann Machine

Restricted Boltzmann Machine merupakan jaringan saraf yang bersifat *stochastic* (jaringan saraf yang berarti memiliki unit neuron berupa aktivasi biner yang bergantung pada neuron-neuron yang saling terhubung, sedangkan *stochastic* berarti aktivasi yang memiliki unsur probabilistik) yang terdiri dari dua binary unit yaitu visible layer merupakan state yang akan diobservasi dan hidden layer merupakan feature detectors serta unit bias[9]. Selanjutnya masing-masing visible unit terhubung ke semua hidden unit yang diwakili oleh array bobot, sehingga setiap hidden unit juga terhubung ke semua visible unit dan unit bias terhubung ke semua visible unit dan semua hidden unit. Untuk memudahkan proses pembelajaran, jaringan dibatasi sehingga tidak ada visible unit terhubung ke visible unit lain dan hidden unit terhubung dengan hidden unit lainnya.

Dalam melatih RBM, sampel data yang digunakan sebagai masukan untuk RBM melalui neuron visible, dan kemudian jaringan sampel bolak-balik antara neuron visible dan hidden. Tujuan dari pelatihan adalah untuk pembelajaran koneksi bobot pada visible atau hidden dan bias aktivasi neuron sehingga RBM belajar untuk merekonstruksi data input selama fase dimana sampel neuron visible dari neuron hidden. Setiap proses sampling pada dasarnya berupa perkalian matriks-matriks antara sekumpulan sampel pelatihan dan matriks bobot,

diikuti dengan fungsi aktivasi neuron, yaitu fungsi *truncated exponential distribution* (TED). Sampling antara lapisan hidden dan visible diikuti oleh modifikasi parameter (dikontrol oleh learning rate) dan diulang untuk setiap kelompok data dalam training set, dan untuk state sebanyak yang diperlukan untuk mencapai konvergensi.



Gambar 5 Gambar Alur Restricted Boltzmann Machine

Konfigurasi unit visible (x) dan unit hidden (h) memiliki energi [9]. Ditunjukkan pada persamaan berikut:

$$E(x, h; \Lambda) = -x'b - c'h - x'Wh \quad (2.8.1)$$

Dimana Λ merupakan set parameter RBM $\{b, c, W\}$, b adalah bias untuk visible, kemudian c merupakan bias untuk hidden, dan W adalah bobot antara unit visible dan unit hidden.

Unit visible diinisialisasi dan diperbarui menggunakan persamaan berikut, dimana untuk mencari nilai x dengan nilai h (hidden) diatur dengan probabilitas:

$$p(x|h; \Lambda) = \prod_{i=1}^N C(\alpha_i) e^{\alpha_i x_i} = C(\alpha) e^{\alpha'x} \quad (2.8.2)$$

Dimana α diperoleh dari rumus dibawah ini:

$$\alpha = b + Wh \quad (2.8.3)$$

Kemudian $C(\alpha)$ adalah fungsi TED-TED

$$C(\alpha) = \left(\frac{\alpha}{e^{\alpha}-1} \right) \quad (2.8.4)$$

Unit hidden diinisialisasi dan derparui menggunakan persamaan berikut, dimana untuk mencari h dengan nilai x , diatur dengan probabilitas sebagai berikut:

$$p(h|x; \Lambda) = \prod_{j=1}^M C(\beta_j) e^{\beta_j h_j} = C(\beta) e^{\beta'h} \quad (2.8.5)$$

Dimana β diperoleh dari rumus dibawah ini:

$$\beta = c + W'x \quad (2.8.6)$$

2.8.1 Langkah-langkah RBM

Pada tahap klasifikasi, terdapat beberapa parameter yang perlu diperhatikan yaitu:

- Tentukan Jumlah node tampak yang digunakan dalam sistem RBM, dalam kasus ini jumlah node tampak adalah 7 node karena mengikuti jumlah data fitur yang dihasilkan dari algoritma moment invariant.
- Tentukan Jumlah node tersembunyi yang digunakan dalam sistem RBM, dalam kasus ini jumlah node tersembunyi adalah 10 node karena diperlukan 10 nilai untuk mendapatkan nilai terbaik dari berbagai percobaan yang saya lakukan.
- Tentukan rasio pembelajaran yang digunakan pada saat tahap pelatihan, Dalam kasus ini nilai rasio pembelajaran adalah 0.01. Semakin rendah nilainya, maka proses pelatihan akan membutuhkan waktu yang semakin lama, Tetapi semakin tinggi nilainya, maka proses pelatihan tidak akan dapat menemukan nilai yang diperkirakan bisa lebih baik
- Tentukan jumlah iterasi yang digunakan pada saat tahap pelatihan, Diasumsikan dalam kasus ini jumlah iterasi adalah 70.
- Langkah-langkah penggunaan algoritma Restricted Boltzmann Machine:
 - Lakukan proses pencarian nilai bobot dan bias terbaik dengan nilai random yang kecil
 - Tetapkan learning rate (ϵ) dan maksimum epoch
 - Lakukan langkah dibawah ini selama (Epoch < Maksimum Epoch)
 - Lakukan langkah dibawah ini selama (datasampel < maksimumdatasampel)
 - Positive phase (mengambil data dan sampel dari hidden unit)
 - Hitunglah energi aktivasi, probabilitas dan state dari unit hidden dengan menggunakan persamaan 2.8.2
 - Hitung positif asosiatif
 $Pos_Asso = (data)^T * P(h)$
 Positif asosiatif diperoleh dari perkalian matriks data sampel yang

ditranpose dari visible neuron dengan probabilitas yang dihasilkan dari langkah 5a

6. Negative Phase (rekonstruksi visible unit dan data sampel dari hidden unit)
 - a. Hitung energi aktivasi dan probabilitas dari unit visible dengan menggunakan persamaan 2.8.5
 - b. Lakukan langkah 5.a untuk update hidden unit
 - c. Hitunglah negatif assosiatif

$$Neg_Asso = (data)^T * P(h) \quad (2.8.8)$$

Negatif assosiatif diperoleh dari matriks data (probabilitas dari unit visible yang diperoleh dari langkah 6.a) yang ditranpose dengan probabilitas dari unit hidden yang dihasilkan dari langkah 6.b

7. Update parameter
 - a. Update bobot

$$W_{ij} = W_{ij} + \Delta W_{jk} \quad (2.8.9)$$

$$\Delta W_{ij} = \varepsilon (Pos_{Asso} - Neg_{Asso})$$

Dimana ε adalah learning rate.

8. Hitung error

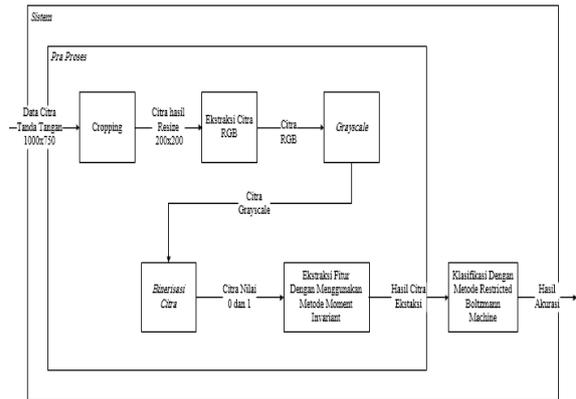
$$Error = \frac{1}{2} \sum_{i=1}^p (O_i - t_i)^2 \quad (2.8.10)$$

Error dihitung dengan pengurangan O_i merupakan data sampel dan t_i merupakan visibel probabilitas yang dihasilkan dari phase negatif pada langkah 6.c

2.9 Rule Based

Metode *rule based* yaitu metode dari salah satu sistem pendukung keputusan yang di dalamnya memiliki basis pengetahuan. Penerapan dari *rule based* dilakukan dengan cara menguji aturan-aturan satu persatu dalam urutan tertentu. Untuk menguji *rule based* tiap aturan akan mengevaluasi apakah kondisinya benar atau salah. Jika kondisi data yang diuji benar maka kemudian dilanjutkan ke aturan berikutnya yang diuji. Sebaliknya apabila kondisi datanya salah maka aturan itu tidak disimpan dan aturan berikutnya diuji. Proses ini akan berulang sampai seluruh basis aturan teruji dengan berbagai kondisi.

2.9 Gambaran Umum Sistem



Gambar 6 Alur Umum Sistem

Pada gambar diagram di atas, proses awal yang dilakukan yaitu memasukkan data citra tanda tangan ke dalam sistem. Langkah selanjutnya melalui tahap pengolahan citra meliputi *cropping*, *nilai rgb*, *grayscale*, dan *ekstraksi ciri* dengan *moment invariant*. Hasil dari pengolahan citra akan didapatkan 7 nilai dari setiap citranya yang didapatkan dari ekstraksi cirinya, kemudian diolah kembali pada tahap klasifikasi dengan metode RBM baik pelatihan maupun pengujian. Demikian ula dengan *rule based* akan tetapi tidak melalui tahap pelatihan. Hasil dari tahap pengujian yaitu berupa akurasi klasifikasi pengenalan tanda tangan.

2.10 Pengolahan Citra Tanda Tangan

Dalam melakukan pengolahan citra dilakukan beberapa proses agar citra input dapat digunakan pada tahap klasifikasi menggunakan metode *Restricted Boltzmann Machine* (RBM). Berikut penjabaran dari proses pengolahan citra yang dilakukan sesuai dengan alur sistem yang dibuat.

1. Cropping

Data citra tanda tangan 1000x750 piksel yang telah di *scan* kemudian perlu diseragamkan menjadi suatu ukuran tertentu agar sistem dapat konsisten membaca citra. Dalam penelitian ini citra di crop sesuai ukuran dan di-*resize* menjadi ukuran yang sudah ditentukan.

2. Nilai RGB

Dalam proses nilai *rgb* (*red, green, blue*), tiap citra yang berwarna memiliki nilai indeks warna RGB yang bermacam-macam. Dalam tahap RGB ini, selanjutnya citra tanda tangan ditulis ke dalam bentuk matriks warna RGB (Red Green Blue) dengan tujuan untuk dapat memperlihatkan nilai-nilai yang dihasilkan dari setiap proses.

3. Grayscale

Proses *grayscale* citra, dalam proses ini hasil dari nilai *rgb* sebelumnya akan diproses ke tahap *grayscale* yang nantinya menghasilkan citra dengan range warna dari 0 sampai dengan 255 yang merupakan skala keabuan.

4. Binerisasi Citra

Langkah selanjutnya dalam pengolahan citra adalah *thresholding*. Proses ini dilakukan terhadap citra *grayscale*. Metode *threshold* yang digunakan dalam penelitian ini adalah *simple threshold*. Hasil akhir dari proses ini adalah citra hitam putih dengan nilai range 0 dan 1..

5. Ekstraksi ciri (*moment invariant*)

hasil dari proses ini yaitu berupa 7 nilai angka biner pada tiap citranya yang nantinya digunakan sebagai data input pada tahap klasifikasi menggunakan metode *Restricted Boltzmann Machine* (RBM).

2.11 Pengujian Akurasi

Pengujian akurasi pada penelitian ini akan dihitung nilai ketepatan atau kecocokan dari hasil sistem pengenalan tanda tangan dalam mendeteksi tanda tangan responden. Dalam pengujian ini parameter pelatihan yang digunakan sama dan akurasi yang dihasilkan pengujian rbm serta rule base yaitu sebagai berikut.

Tabel 1 Akurasi Data Uji

Nama	Output Tertinggi	Terdeteksi sebagai	Benar / Salah
'ACEP',	0.6774	ACEP	Benar
'ACEP',	0.6774	ACEP	Benar
'ACEP',	0.7245	ACEP	Benar
'ARI',	0.68	ARI	Benar
'ARI',	0.68	ARI	Benar
'ARI',	0.7245	ARI	Benar
'ERVAN',	0.6826	ERVAN	Benar
'ERVAN',	0.6826	ERVAN	Benar
'ERVAN',	0.6826	ERVAN	Benar
'FADLI',	0.7929	FADLI	Benar
'FADLI',	0.7929	FADLI	Benar
'FADLI',	0.7082	FADLI	Benar
'FAJAR',	0.68	FAJAR	Benar
'FAJAR',	0.68	FAJAR	Benar
'FAJAR',	0.7245	FADLI	Salah
'IQRAM',	0.68	IQRAM	Benar
'IQRAM',	0.68	IQRAM	Benar
'IQRAM',	0.68	IQRAM	Benar
'MAARIF',	0.6696	MAARIF	Benar
'MAARIF',	0.6696	MAARIF	Benar
'MAARIF',	0.6696	MAARIF	Benar
'OKI',	0.7881	OKI	Benar
'OKI',	0.7881	OKI	Benar
'OKI',	0.7881	OKI	Benar
'RIZKI',	0.7463	RIZKI	Benar
'RIZKI',	0.7463	RIZKI	Benar
'RIZKI',	0.7463	RIZKI	Benar
'SAEFUL',	0.6455	OKI	Salah

'SAEFUL',	0.5269	OKI	Salah
'SAEFUL',	0.4928	SAEFUL	Benar

3.1 Kesimpulan

Berdasarkan hasil penelitian terdapat 150 citra tanda tangan dari 10 responden, dari setiap responden mempunyai 15 tanda tangan. Untuk data latih sebanyak 120 tanda tangan, dari setiap respondennya diambil 12 tanda tangan, kemudian sisanya dijadikan data uji sebanyak 30 citra tanda tangan yang di dapatkan dari masing-masing responden yaitu 3 citra tanda tangan. Dari pengujian yang telah dilakukan dengan metode RBM citra tanda tangan menghasilkan akurasi dengan rata-rata akurasi untuk seluruh fitur yang diuji RBM cukup baik yaitu sebesar 90%. Berikut ini adalah penjelasan dari penyebab nilai akurasi yang dihasilkan sistem sangat baik akurasinya yaitu data latih dan data uji yang diambil polanya diseleksi keseragamannya sama dan ditentukan ukurannya, kemudian dalam[5] proses perhitungan rasio pembelajaran atau learning rate, kemudian jumlah iterasi dan node tersembunyi nilainya yang tepat. Maka dari itu akurasi yang di dapatkan sangat baik.

Dapat diambil kesimpulan bahwa nilai akurasi klasifikasi metode RBM sangat baik, dan masih bisa ditingkatkan lagi untuk digunakan dalam sistem pengenalan tanda tangan, akan tetapi dengan mengembangkan atau menambahkan proses fitur dalam image processing dan diperbanyak untuk data latihnya agar dapat mengklasifikasikan dengan lebih baik.

3.2 Saran

Berdasarkan hasil pengujian yang telah dilakukan, maka saran yang diharapkan adalah sebagai berikut.

1. Menambahkan data citra tanda tangan dalam data latih dan data uji, untuk memperoleh hasil yang lebih baik dalam mengenali tanda tangan dari setiap respondennya.
2. Menambahkan fitur yang belum ada pada penelitian ini untuk melengkapi sistem pengenalan tanda tangan untuk mendapatkan hasil yang baik dalam mengenali tanda tangan.

DAFTAR PUSTAKA

- [1] “arti tanda tangan dan fungsinya,” 2016. [Daring]. Tersedia pada: <https://www.masterpendidikan.com>.
- [2] A. R, *A Prototype System for off-line Signature Verification using Multilayered Feedforward Neural Networks*. Tesis Departmen of Computer Science, RMIT. 1994
- [3] A. Jariah, “PENGENALAN POLA TANDA TANGAN MENGGUNAKAN METODE MOMENT INVARIANT DAN JARINGAN SYARAF RADIAL BASIS FUNCTION (RBF) Pengertian Tanda Tangan,” hal. 85–92, 2011.
- [4] M. I. D. Putra, A. Romadhony, M. D. Suliiyo, F. T. Informatika, dan U. Telkom, “NEURAL NETWORK Latar Belakang Tanda tangan merupakan sebuah karakter unik yang dimiliki seseorang,” hal. 0–6, 2013.
- [5] R. Sofian Bahri and I. Maliki, “Perbandingan Algoritma Template Matching dan Feature Extraction pada Optical Character Recognition,” *J. Komput. dan Inform.*, vol. 1, pp. 29–35, 2012.
- [6] T. Callahan, J. Hauser, and J. Wawrzynek, “The Garp Architecture and C Compiler,” *Computer*, vol. 33, no. 4, pp. 62-69, 2000.
- [7] Jauhari, nurdin, *Edge Linking Detection dan Perbandingan dari 3 Metodenya*, <http://ahtovicblogs.blog.ugm.ac.id/?p=80>, 17 Maret 2011 20.25
- [8] H. Ming-Kuei, “Visual pattern recognition by moment invariants,” *IRE Trans. Inf. Theory*, vol. 8, no. 2, pp. 179–187, 1962.
- [9] Jose, Stephane, *Why Should I Care About SQL Server*, <http://blog.iweb.com/en/2010/06/why-should-i-care-about-sql-server/4772.html>, 17 Maret 2011 19.30
- [10] S. Nasional, T. Informatika, U. M. Area, I. L. Belakang, and S. Deshmukh, “Algoritma Restricted Boltzmann Machines (RBM) untuk Pengenalan Tulisan Tangan Angka,” pp. 140–148, 2017.