

NASKAH PUBLIKASI

**IMPLEMENTASI ALGORITMA BACKPROPAGATION UNTUK
MENGENALI POLA TULISAN ANGKA**

PROYEK TUGAS AKHIR

Disusun oleh :

**HADYAN INDRA NAKSATRA
5130411014**

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN ELEKTRO
UNIVERSITAS TEKNOLOGI YOGYAKARTA
2019**

NASKAH PUBLIKASI

**IMPLEMENTASI ALGORITMA BACKPROPAGATION UNTUK
MENGENALI POLA TULISAN ANGKA**

PROYEK TUGAS AKHIR

Disusun oleh :
HADYAN INDRA NAKSATRA
5130411014

Telah disetujui oleh pembimbing

Pembimbing,

Dr. Enny Itje Sela, S.Si., M.Kom.

Tanggal :

IMPLEMENTASI ALGORITMA BACKPROPAGATION UNTUK MENGENALI POLA TULISAN ANGKA

Hadyan Indra Naksatra

Program Studi Teknik Informatika, Fakultas Teknologi Informasi dan Elektro

Universitas Teknologi Yogyakarta

Jl. Ringroad Utara Jombor Sleman Yogyakarta

E-mail : naksatra.hi@gmail.com

ABSTRAK

Seiring perkembangan zaman dan perkembangan teknologi informasi. Kini semuanya dapat di jangkau dengan menggunakan teknologi. Dengan kata lain Komputer memiliki peran penting. Komputer dapat mendukung pengenalan pola secara otomatis, seperti masalah yang menyita perhatian sekarang ini, baik pengenal pola wajah, sidik jari, tanda tangan maupun pola karakter lainnya. Salah satu kelemahan komputer dibandingkan dengan manusia adalah mengenali pola karakter jika tidak menggunakan metode pendukung. Jaringan Saraf Tiruan (JST) adalah suatu metode atau konsep yang mengambil sistem saraf manusia. Pada sistem JST terdapat beberapa metode yang digunakan untuk melakukan pelatihan terhadap komputer yang dibuat, pelatihan digunakan untuk meningkatkan ketelitian atau kemampuan komputer untuk mengenali pola. Salah satu algoritma JST yang digunakan untuk melatih data sistem yaitu backpropagation. Dengan adanya metode Jaringan Saraf Tiruan(JST) algoritma dapat menghasilkan sistem yang dapat mengenali pola karakter tulisan tangan angka dengan baik yang dapat mempermudah manusia untuk melakukan pengenalan pola. Hasil proses pengujian menggunakan algoritma Backpropagation mencapai 76,67% dengan jumlah data 30 data terlatih. Hasil pengujian dari data uji mencapai 10% dari 10 data uji.

Kata kunci: Pengenalan Pola, Angka, Jaringan Syaraf Tiruan, Backpropagation

1. PENDAHULUAN

Pada zaman sekarang peran komputer sangatlah penting. Komputer sangat sering digunakan oleh manusia untuk membantu melakukan komputasi yang sulit, dilakukan oleh manusia. Belakangan ini komputer sering digunakan untuk melakukan komputasi pengenalan pola karakter, seperti pengenalan pola karakter sidik jari, pola karakter tanda tangan, pola karakter plat nomor dan masih banyak lagi. Hal ini dilakukan karena komputer memiliki kemampuan untuk mengenali pola dengan baik dibandingkan dengan manusia.

Pengenalan pola yang dilakukan oleh komputer dapat dilakukan oleh sebuah komputer atau sistem komputer, akan tetapi komputer juga memiliki beberapa kekurangan, salah satunya yaitu dalam hal ketelitian jika tidak diberikan sebuah metode atau algoritma tambahan. Salah satu metode yang sering digunakan adalah Jaringan Saraf Tiruan (JST) yang mengambil konsep sistem saraf manusia.

Jaringan Saraf Tiruan (JST) adalah suatu metode atau konsep yang mengambil sistem saraf manusia. Pada sistem JST terdapat beberapa metode yang digunakan untuk melakukan pelatihan terhadap sistem yang dibuat, pelatihan digunakan untuk meningkatkan ketelitian atau kemampuan sistem untuk

mengenali pola, seperti pola karakter tulisan. Salah satu algoritma JST yang digunakan untuk melatih yaitu *backpropagation*.

Pengenalan pola karakter memiliki banyak karakter sebagai contoh adalah pola karakter tanda tangan, pola karakter plat nomor, pola karakter batik, dan masih banyak lagi. Di dalam penelitian ini penulis menggunakan pola karakter tulisan angka dari angka 0 hingga angka 9.

2. LANDASAN TEORI

2.1 Tinjauan Pustaka

Pengenalan Pola Karakter Plat Nomor Kendaraan Menggunakan Algoritma Momentum *Backpropagation Neural Network* yaitu menerapkan algoritma momentum *backpropagation* untuk mengenali pola karakter nomor kendaraan yang digunakan untuk sistem lalu lintas cerdas yang antara lain berfungsi untuk menemukan kendaraan yang dicuri, pembayaran tiket parkir otomatis, dan menindak para pelanggar lampu merah. Pengujian dilakukan terhadap angka dan huruf dari 276 plat nomor kendaraan yang ada di Indonesia. Hasil dari sistem tersebut adalah pengenalan terhadap angka dan huruf dari 276 plat nomor kendaraan tersebut dengan akurasi 97,10%. [1]

Penerapan Algoritma Jaringan Saraf Tiruan Backpropagation Untuk Pengenalan Pola Tanda Tangan dengan menerapkan algoritma *backpropagation* yang bertujuan untuk mengidentifikasi tanda tangan dari berbagai individu. Pengujian dilakukan dengan menggunakan tandatangan berupa citra digital atau dengan menggunakan tandatangan yang dibuat dengan *pointer mouse*. Hasil dari penelitian ini adalah pengenalan terhadap 30 data pelatihan dari 10 data uji, aplikasi memiliki akurasi pengenalan tandatangan sebesar 90%. [2]

Aplikasi Pengenalan Kata Pada Huruf Braille Dan Pelafalannya dengan menerapkan metode *Chain Approximation* untuk deteksi titik-titik *braille* dan metode *Horizontal and Vertical Projection* untuk pengelompokan posisi-posisi *braille* menjadi karakter sesuai dengan aturan pembacaan huruf *braille* yang berlaku. Tujuan dibuatnya sistem ini adalah untuk menyederhanakan proses pengenalan dari scanner menjadi kamera *handphone* atau *webcam*. Hasil dari penelitian menghasilkan sebuah aplikasi untuk mengenali huruf *braille* secara dinamis dengan akurasi 84%. [3]

Penerapan Jaringan Saraf Tiruan Metode Propagasi Balik dalam Pengenalan Tulisan Tangan Huruf Jepang Jenis Hiragana dan Katakana. Dalam penelitian ini membahas mengenai pengenalan pola tulisan tangan huruf Jepang. Penelitian menggunakan metode Jaringan Saraf Tiruan algoritma *backpropagation*. Input yang digunakan dalam sistem ini adalah citra tulisan tangan yang diubah kedalam matrik. [4]

Sistem Identifikasi dan Pengenalan Pola Citra Tanda-Tangan Menggunakan Sistem Jaringan Saraf Tiruan (Artificial Neural Networks) Dengan Metode Backpropagation. Dalam penelitiannya sistem menggunakan jumlah 1 pola pelatihan, sistem mampu mengenali pola luaran sebesar 50% sedangkan dengan 5 pola pelatihan sistem mampu mengenali pola luaran sebesar 70%. [5]

2.2 Dasar Teori

2.2.1 Jaringan Saraf Tiruan

Jaringan saraf tiruan (*artificial neural network*) atau disingkat JST adalah sistem komputasi dimana arsitektur dan operasi diilhami dari pengetahuan tentang sel saraf biologis di dalam otak, yang merupakan salah satu representasi buatan dari otak manusia yang selalu mencoba menstimulasi proses pembelajaran pada otak manusia tersebut. [6]

Setiap pengolahan elemen membuat perhitungan berdasarkan pada jumlah masukan (*input*). Sebuah kelompok pengolahan elemen disebut *layer* atau lapisan dalam jaringan. Lapisan pertama adalah *input* dan yang terakhir adalah *output*. Lapisan di antara

lapisan *input* dan *output* disebut lapisan tersembunyi (*hidden layer*). [6]

Jaringan saraf tiruan merupakan suatu bentuk arsitektur yang terdistribusi paralel dengan sejumlah besar *node* dan hubungan antar-*node* tersebut. Tiap titik hubungan dari satu *node* ke *node* yang lain mempunyai harga yang diasosiasikan dengan bobot. Setiap *node* memiliki suatu nilai yang diasosiasikan sebagai nilai aktivasi *node*. [6]

2.2.2 Arsitektur jaringan saraf tiruan

Jaringan saraf tiruan dirancang menggunakan suatu aturan yang bersifat menyeluruh (*general rule*) dimana seluruh model jaringan memiliki konsep dasar yang sama. Arsitektur sebuah jaringan akan menentukan keberhasilan target yang akan dicapai karena tidak semua permasalahan dapat diselesaikan dengan arsitektur yang sama [6]

2.2.3 Fungsi Aktifasi

Fungsi aktivasi merupakan bagian penting dalam tahapan perhitungan keluaran dari suatu algoritma. Beberapa fungsi aktivasi yang digunakan dalam jaringan saraf tiruan adalah sebagai berikut:

a. Fungsi identitas

$$f(x) = x \quad (2.1)$$

Fungsi identitas sering dipakai apabila kita menginginkan keluaran jaringan berupa sembarang bilangan riil (bukan hanya pada range $[0,1]$ atau $[-1,1]$).

b. Fungsi Threshold (batas ambang)

$$f(x) = \begin{cases} 1 & \text{jika } x \geq a \\ 0 & \text{jika } x < a \end{cases} \quad (2.2)$$

Untuk beberapa kasus, fungsi threshold yang dibuat tidak berharga 0 atau 1, tapi berharga -1 atau 1 (sering disebut threshold bipolar).

$$f(x) = \begin{cases} 1 & \text{jika } x \geq a \\ -1 & \text{jika } x < a \end{cases} \quad (2.3)$$

c. Fungsi sigmoid

Fungsi sigmoid merupakan salah satu aktivasi yang sering dipakai dalam algoritma *backpropagation* karena memenuhi syarat yaitu kontinu, terdiferensial dengan mudah dan merupakan fungsi tidak turun. Salah satu fungsi yang memenuhi ketiga syarat tersebut dan sering dipakai adalah fungsi sigmoid biner yang memiliki range $(0,1)$.

$$f(x) = \frac{1}{1+e^{-x}} \text{ dengan turunan} \\ f'(x) = f'(x)(1 - f(x)) \quad (2.4)$$

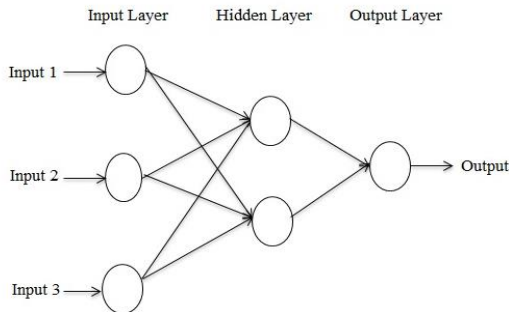
Fungsi lain yang sering dipakai adalah sigmoid bipolar yang bentuk fungsinya mirip dengan fungsi sigmoid biner, tetapi dengan range $(-1,1)$.

$$f(x) = \frac{2}{1+e^{-x}} - 1 \text{ dengan turunan} \\ f'(x) = \frac{(1+f(x))(1-f(x))}{2} \quad (2.5)$$

2.2.4 Algoritma Backpropagation

Jaringan perambatan galat mundur atau yang disebut juga dengan *backpropagation* merupakan salah satu algoritma yang sering digunakan dalam menyelesaikan masalah-masalah yang rumit. Hal ini dimungkinkan karena jaringan dengan algoritma ini dilatih dengan menggunakan metode belajar terbimbing. Pada jaringan diberikan sepasang pola yang terdiri atas pola masukan dan pola yang diinginkan. Ketika suatu pola diberikan kepada jaringan, bobot diubah untuk memperkecil perbedaan pola keluaran dan pola yang diinginkan. Latihan ini dilakukan berulang-ulang sehingga semua pola yang dikeluarkan jaringan dapat memenuhi pola yang diinginkan. Algoritma pelatihan jaringan saraf perambatan galat mundur terdiri atas dua langkah, yaitu perambatan maju (*forward propagation*) dan perambatan mundur (*backward*) [6].

Jaringan perambatan galat mundur terdiri atas tiga lapisan atau lebih unit pengolah. Gambar 2.1 menunjukkan jaringan perambatan galat mundur dengan tiga lapisan pengolah, bagian kiri sebagai masukan, bagian tengah disebut sebagai lapisan tersembunyi dan bagian kanan disebut lapisan keluaran. Ketiga lapisan terhubung secara penuh [6].



Gambar 1. Arsitektur Backpropagation

Perambatan maju dimulai dengan pola masukan ke lapisan masukan. Setelah perambatan maju selesai dikerjakan maka jaringan siap melakukan perambatan mundur. Langkah perambatan mundur adalah menghitung galat dan mengubah bobot-bobot pada semua interkoneksi. Perhitungan galat mundur dimulai dari lapisan keluaran dan mundur sampai lapisan masukan. Jaringan *backpropagation* dilatih dengan metode belajar terbimbing. Nilai 'benar' ditunjukkan dengan nilai RMS/SSE dengan galat yang biasanya mempunyai nilai dibawah 0,1. Dengan nilai RMS/SSE dibawah 0,1 maka jaringan sudah boleh dikatakan terlatih. Jika galat yang muncul lebih besar dari galat iterasi sebelumnya maka nilai bobot, prasikap, keluaran dan galat yang baru disimpan serta laju belajar harus ditingkatkan [6].

2.2.5 PCA (Principal component Analysis)

PCA atau principal component analysis adalah teknik yang digunakan untuk menyederhanakan suatu data, dengan cara mentransformasi data secara linier sehingga terbentuk sistem koordinat baru dengan varians maksimum. Analisis komponen utama dapat digunakan untuk mereduksi dimensi suatu data tanpa mengurangi karakteristik data tersebut secara signifikan.[7]

Sebuah citra 2D dengan dimensi b baris dan k kolom dapat direpresentasikan dalam bentuk citra 1D dengan dimensi $n = (b * k)$. Misalkan ada sampel berupa data latih sejumlah K sampel yang dinyatakan dengan $\{x_1, x_2, \dots, x_n\}$ yang diambil dari C buah kelas yang dinyatakan sebagai $\{x_1, x_2, \dots, x_n\}$. Matriks kovarian (S_T) dapat didefinisikan sebagai berikut [8]:

$$S_T = \sum_{k=1}^k (x_k - \mu)(x_k - \mu)^T$$

Dimana μ adalah rata-rata sampel *image* yang diperoleh dengan merata-rata *training image* $\{x_1, x_2, \dots, x_n\}$. Dengan dekomposisi *eigen, matrix covariance* ini dapat didekomposisi menjadi :

$$S_T = \theta \Lambda \theta^T \quad (2.6)$$

Dimana θ adalah *matrix eigenvector*, dan Λ adalah sebuah diagonal matrix dari nilai *eigen*. Kemudian dipilih sejumlah kolom *eigenvector* dari matrix θ yang berasosiasi dengan sejumlah nilai *eigen* terbesar. Pemilihan *eigenvector* ini menghasilkan matrix transformasi atau matrix proyeksi θ_m , yang mana terdiri dari kolom *eigenvector* terpilih yang biasa disebut juga dengan *eigenimage*. Berikutnya sebuah *image* x (berdimensi n) dapat diekstraksi kedalam fitur baru (berdimensi $m < n$) dengan memproyeksikan x searah dengan sebagai berikut :

$$y = \theta_m x \quad (2.7)$$

Dengan kata lain metode PCA memproyeksikan ruang asal \mathcal{R}^n kedalam ruang baru yang berdimensi lebih rendah \mathcal{R}^m , yang mana sebanyak mungkin kandungan informasi asal tetap dipertahankan untuk tidak terlalu banyak hilang setelah dibawa ke dimensi fitur yang lebih kecil. Disini terlihat reduksi fitur yang signifikan dari n buah menjadi m buah yang tentunya akan sangat meringankan komputasi dalam proses pengenalan berikutnya.[8]

2.2.6 Konvergensi

Konvergensi merupakan nilai data yang akan semakin mengecil, dimana jaringan akan pada kondisi stabil di nilai minimum. Dapat dibuktikan pada tiap unit pengolahan nilai akan selalu berkurang atau tetap, sebagai hasilnya unit pengolahan nilai berada pada nilai minimum untuk konvergen. [6]

2.2.7 Normalisasi

Normalisasi adalah proses transformasi dimana sebuah atribut numerik diskalakan dalam range yang lebih kecil seperti -1 sampai 1 atau 0 sampai 1. Ada beberapa metode atau teknik yang diterapkan untuk normalisasi data, diantaranya [9]:

a. Min-Max Normalization

Min-max normalization memetakan sebuah value v dari atribut A menjadi v' kedalam range $[new_min_A, new_max_A]$ berdasarkan rumus:

$$v' = \frac{v - min_A}{max_A - min_A} (new_max_A - new_min_A) + new_min_A \quad (2.8)$$

Dimana:

- v' = nilai hasil normalisasi
- v = nilai yang akan dinormalisasi
- max_A = nilai maksimum dari data per kolom
- min_A = nilai minimum dari data per kolom
- new_max_A = nilai maksimum baru
- new_min_A = nilai minimum baru

b. Z-Score Normalization

Disebut juga zero-mean normalization, dimana value dari sebuah atribut A dinormalisasikan berdasarkan nilai rata-rata dan standar deviasi dari atribut A . Sebuah value v dari atribut A dinormalisasikan menjadi v' dengan rumus:

$$v' = \frac{v - \bar{A}}{\sigma_A} \quad (2.9)$$

\bar{A} dan σ_A adalah nilai rata-rata dan standar deviasi dari atribut A .

c. Normalization by Decimal Scalling

Normalisasi yang diperoleh dengan melakukan penggeseran titik desimal dari value sebuah atribut A . Jumlah titik desimal yang digeser tergantung dari nilai absolut maksimum dari atribut A .

2.2.8 Ukuran Ketepatan

Ukuran Ketepatan yang sering digunakan untuk mengetahui ketepatan suatu metode yaitu sebagai berikut [10]:

a. Nilai Tengah Galat (*Mean Squared Error*)

$$MSE = \frac{1}{n} \sum_{i=1}^n e_i^2 \text{ dimana, } e_i = X_i - F_i \quad (2.10)$$

b. Jumlah Kuadrat Galat (*Sum of Squared Error*)

$$SSE = \sum_{i=1}^n e_i^2 \text{ dimana, } e_i = X_i - F_i \quad (2.11)$$

c. Nilai Tengah Galat Absolut (*Mean Absolute Error*)

$$MAE = \frac{1}{n} \sum_{i=1}^n |e_i| \text{ dimana, } e_i = X_i - F_i \quad (2.12)$$

d. Mean Absolute Percentage Error

$$MAPE = \frac{1}{n} \sum_{i=1}^n |PEt| \text{ dimana, } |PEt| = \left(\frac{X_t - F_t}{X_t} \right) \times 100\% \quad (2.13)$$

Dengan:

- X_i : nilai yang diinginkan
- F_i : nilai yang dihasilkan
- X_t : hasil JST
- F_t : target
- n : jumlah

3. METODOLOGI PENELITIAN

3.1 Objek Penelitian

Pada penelitian Tugas Akhir ini yang menjadi objek penelitian adalah pola tulisan tangan angka yang ditulis oleh penulis. Pola tulisan tangan angka terdiri dari angka 0 sampai angka 9. Data tulisan angka didapat dari hasil *scan* menggunakan scanner dengan format .jpg. Ukuran dari pola tulisan tangan angka tersebut adalah 100 x 100 pixel.



Gambar 2. Pola Tuisan Tangan Angka

3.2 Metode Penelitian

3.2.1 Pengumpulan Data

Metode penelitian yang dilakukan penulis dalam melakukan penelitian untuk mendapatkan data yang lengkap dan akurat, dilakukan metode pengumpulan data sebagai berikut :

a. Observasi

Observasi adalah suatu kegiatan dengan melakukan pengamatan pada suatu objek atau bidang yang sedang di teliti. Pengumpulan data pola tulisan angka dilakukan dengan menuliskan angka dari 0 sampai 9 dengan berbagai bentuk pola tulisan angka.

b. Wawancara

Wawancara adalah kegiatan yang dilakukan dalam rangka mengumpulkan data yang diperlukan untuk membangun sebuah sistem dengan cara tatap muka secara langsung dengan orang yang ahli dibidang yang akan diteliti. Wawancara dilakukan dengan dosen pembimbing dengan melakukan tanya jawab sehingga bisa diketahui suatu permasalahan dan penyelesaiannya.

c. Studi Pustaka

Studi pustaka merupakan kegiatan mencari materi-materi yang dibutuhkan sebagai acuan untuk keperluan penelitian. Studi pustaka dapat diperoleh dari buku, *e-book*, jurnal serta *website* yang mempunyai keterkaitan dengan Pengenalan pola dan Jaringan Saraf Tiruan.

3.2.2 Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan pada penelitian ini adalah model *waterfall*. Model *waterfall* sering disebut juga model sekuensi linear (*sequential linear*) atau alur hidup klasik (*classic life style*). Model ini menyediakan pendekatan alur perangkat lunak secara sekuensial dimulai dari analisis, desain, pengkodean, pengujian dan tahap pendukung.

a. Analisis Sistem

Analisis sistem dilakukan untuk mengetahui informasi yang dibutuhkan untuk membangun sebuah program aplikasi. Tahapan ini dilakukan untuk mengetahui alur proses dari program aplikasi yang akan dibuat.

b. Desain

Desain merupakan tahap menspesifikasikan bagaimana sistem dapat memenuhi kebutuhan, untuk dapat memenuhi kebutuhan pengguna diperlukan beberapa tahap perancangan. Berikut ini akan diberikan perincian tentang perancangan yang akan dibuat sebagai berikut:

1. Desain Input

Desain input berfungsi untuk memasukkan data dan memprosesnya ke dalam format yang sesuai. Input data yang akan digunakan dalam sistem ini diperoleh dari pola tulisan tangan angka yang discan. Hasil scan tersebut kemudian dikonversi menjadi data oleh sistem. Data inputan tersebut berupa nilai-nilai eigen.

2. Desain Proses

Desain proses merupakan tahap untuk membuat rancangan yang terdiri dari modul-modul yang akan dimiliki sistem. Desain yang akan digunakan adalah *Data Flow Diagram* (DFD) yang akan menjelaskan alur atau proses dari rancangan program yang akan dibangun.

3. Desain Output

Desain *output* merupakan hasil akhir dari aplikasi yang dibangun. Output berupa hasil pengenalan angka menggunakan algoritma *backpropagation*.

4. Desain Interface

Desain *interface* perancangan antarmuka dilakukan sesederhana mungkin tetapi tidak menghilangkan unsur-unsur penting dalam menyampaikan informasi. Desain *interface* pada program aplikasi ini dibuat menggunakan Netbeans IDE 8.0.2.

c. Pengkodean

Pengkodean merupakan tahap menerjemahkan desain ke program perangkat lunak. Hasil dari pengkodean sesuai dengan desain yang telah dirancang sebelumnya. Bahasa pemrograman yang digunakan adalah Java.

d. Pengujian

Pengujian dilakukan untuk mengecek apakah program aplikasi baik dari sisi logika maupun fungsional yang dibuat telah sama dan sesuai dengan yang diinginkan.

3.3. Kebutuhan Perangkat Keras dan Perangkat Lunak

Perangkat pendukung pada penelitian terdiri atas perangkat keras dan perangkat lunak. Perangkat keras yang digunakan dalam penelitian dapat dilihat pada Tabel 1.

Tabel 1 Perangkat Keras Penelitian

Processor	Intel Core i3 5005U, 2.0GHz
RAM	8 GB
Hardisk	500 GB
Scanner	Cannon Pixma MP258

Sedangkan perangkat lunak yang digunakan dalam penelitian dapat dilihat pada Tabel 2.

Tabel 2 Perangkat Lunak Penelitian

Sistem Operasi	Windows 10
Program Aplikasi	Netbeans IDE 8.0.2

4. ANALISIS DAN PERANCANGAN SISTEM

4.1. Analisis Sistem yang Berjalan

Sistem yang akan dibangun merupakan sebuah sistem yang berbasis dekstop yang digunakan untuk mengenali pola tulisan angka. Sistem yang dibangun menggunakan metode Jaringan Syaraf Tiruan (JST) algoritma *backpropagation*. Dari aplikasi yang dibangun, memiliki beberapa menu yang dapat memproses data yaitu input gambar yang digunakan untuk tahap *preprocessing* dari hasil scanner dan penyimpanan data kedalam *database*, pelatihan digunakan untuk melatih sistem agar pintar dalam mengenali pola tulisan tangan angka dan pengujian yang digunakan untuk menguji sistem apakah sudah baik atau belum, pengujian dilakukan dengan menggunakan data latihan maupun data uji.

4.2. Rancangan Sistem

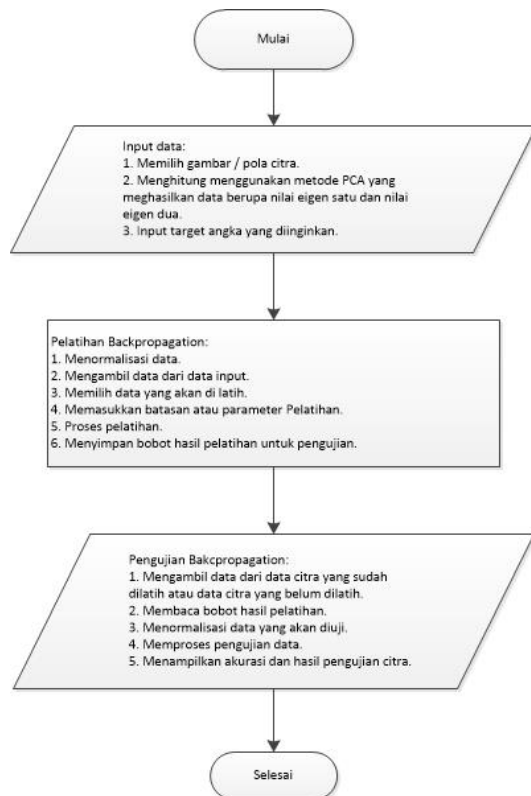
Tahap rancangan sistem adalah tahap dimana akan dilakukan perancangan keseluruhan sistem yang telah diuraikan pada tahap analisis sistem. Pada tahap ini akan dijelaskan perancangan yang digunakan untuk membangun sistem.

4.4.1 Gambaran Umum Sistem

Sistem ini digunakan untuk mengenali pola tulisan angka. Untuk melakukan proses pengenalan pola tulisan angka dibutuhkan input yaitu sebuah gambar tulisan angka yang yang digunakan untuk inputan. Input tersebut akan diolah dengan menggunakan metode Jaringan Saraf Tiruan algoritma *Backpropagation*.

Sistem dimulai dengan memasukkan gambar pola angka tulisan tangan yang sudah discan. Dari

gambar tersebut kemudian diolah oleh sistem yang berupa nilai eigen. Hasil masukan nilai eigen tersebut digunakan untuk proses selanjutnya yaitu pelatihan dan pengujian. Pada proses selanjutnya yaitu proses pelatihan. Data masukan awal diolah untuk dilatih dengan menggunakan parameter yang sudah ditentukan. Hasil dari pelatihan tersebut adalah berupa bobot yang digunakan untuk menguji data uji.



Gambar 3. Alur Program

4.4.2 Algoritma Preprocessing

Preprocessing merupakan proses dilakukannya pengolahan data dari data mentah sampai data siap diolah, berikut beberapa langkah yang harus dilakukan pada sistem ini:

a. Citra pola tulisan angka

Input dari sistem berasal dari citra tulisan angka yang ditulis oleh penulis. Kemudian hasil tulisan tangan tersebut di *scan* dengan menggunakan *scanner*. Hasil scan dipotong dan diresize menjadi ukuran 100x100 pixel dengan menggunakan aplikasi *paint*.



Gambar 4. Keseluruhan Data Hasil Scanner

Sedangkan untuk hasil dari pemotongan dan *resize* dengan aplikasi *paint* dengan ukuran 100x100 *pixel* dapat dilihat pada gambar 5.

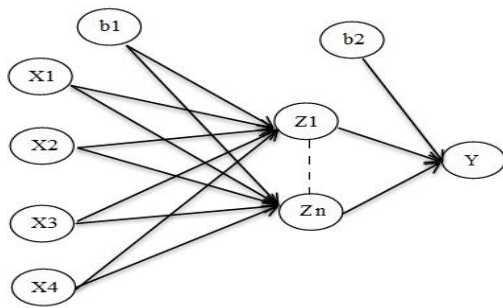


Gambar 5. Potongan Citra Hasil

Gambar 5 adalah gambar hasil pemotongan dan *resize*. Gambar tersebut akan digunakan untuk input sistem. Yang akan dikonversi dengan menggunakan metode *Principal Component Analysis (PCA)* menjadi data nilai eigen

4.4.3 Pelatihan Backpropagation

Melakukan pelatihan data citra pada pelatihan, fungsi aktivasi yang digunakan adalah fungsi aktivasi sigmoid. Pelatihan dimulai dengan menetapkan parameter yang akan digunakan, memanggil data hasil dari citra yang berupa nilai-nilai eigen pada table citra, kemudian proses pelatihan dimulai yang akan menyimpan bobot hasil pelatihan kedalam tabel yang nantinya akan digunakan untuk pengujian. Untuk arsitektur jaringan saraf tiruan *backpropagation* yang digunakan dapat dilihat pada gambar 6. Algoritma *backpropagation* terdiri dari 3 *layer* yaitu *input layer*, *hidden layer* dan *output layer*.



Gambar 6. Arsitektur Jaringan Backpropagation

4.4.4 Validasi Backpropagation

Setelah dilakukan pelatihan data *backpropagation*, tahap selanjutnya adalah validasi data *backpropagation*. Validasi data *backpropagation* merupakan pengujian data dengan menggunakan data latih untuk melihat akurasi data hasil pelatihan *backpropagation*. Proses validasi yang akan berjalan pada sistem akan dijelaskan pada algoritma validasi *backpropagation*. Langkah pertama yaitu mulai, kemudian panggil data hasil input citra, ambil bobot dari tabel bobot dan panggil fungsi aktivasi untuk dilakukan perhitungan JST dan akurasi. Data validasi yang digunakan berasal dari tabel citra dengan pilih 1 yang berarti data pelatihan dan bobot diambil dari tabel bobot di SQLyog yang telah disimpan pada pelatihan sebelumnya.

4.4.5 Pengujian Backpropagation

Tahap selanjutnya adalah pengujian data *backpropagation*. Pengujian data merupakan pengujian data dengan menggunakan data uji untuk melihat akurasi data hasil pelatihan. Proses pengujian yang akan berjalan pada sistem dijelaskan pada algoritma pengujian *backpropagation*. Langkah pertama yaitu mulai, kemudian panggil data hasil input citra dan bobot pada tabel bobot untuk dilakukan perhitungan JST dan akurasi. Data validasi yang digunakan berasal dari tabel citra dengan pilih 0 yang berarti data uji, setelah itu bobot dimuat dari tabel bobot pada SQLyog yang telah disimpan pada pelatihan sebelumnya.

5. IMPLEMENTASI

5.1. Perangkat Pendukung Penelitian.

Implementasi sistem adalah tahap yang dilakukan untuk membangun sistem yang digunakan untuk mengenali pola tulisan angka. Pada sistem ini menggunakan metode jaringan syaraf tiruan (JST) dengan algoritma *backpropagation*. Pada tahap ini akan dijelaskan mengenai perangkat lunak dan perangkat keras yang digunakan untuk membangun sistem ini.

Tabel 3. Perangkat Keras Pembuatan Sistem

Perangkat Keras	
Laptop	Asus A455L
Processor	Prosesor Intel Core i3-5005U, 2.0GHz
RAM	RAM 8 GB
Harddisk	500 GB
VGA	NVIDIA GEFORCE 930M
Scanner	Cannon Pixma MP258

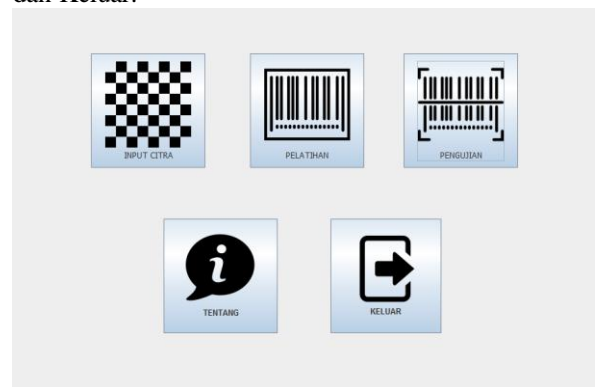
Tabel 4. Perangkat Lunak Pembuatan Sistem

Perangkat Lunak	
Sistem Operasi	Windows 10
Program Aplikasi	Netbeans IDE 8.0.2.
Database	SQLyog
Koneksi Database	Xampp
Desain	Balsamiq Mockup

5.3. Implementasi Sistem

5.3.1 Implementasi Form Menu Utama

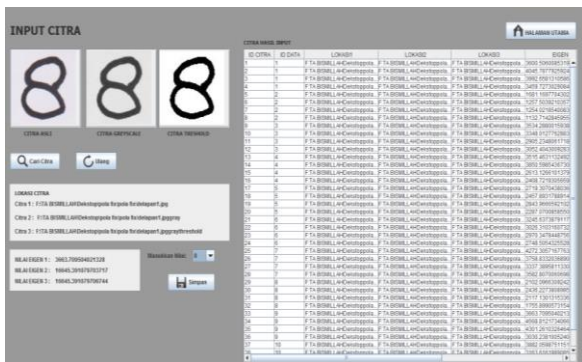
Halaman menu utama adalah halaman ketika program pertama kali dijalankan. Terdapat beberapa menu yaitu Input Citra, Pelatihan, Pengujian, Tentang dan Keluar.



Gambar 7. Form Halaman Utama

5.3.2 Implementasi Form Input Citra

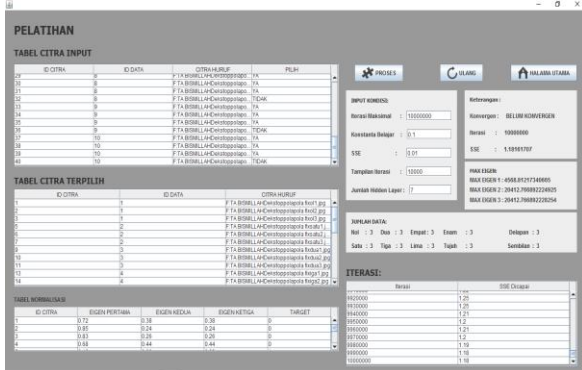
Form input citra digunakan untuk memasukkan data citra ke dalam database. Pada form ini pengguna mencari citra yang akan di input, setelah itu pengguna diminta untuk memilih nilai angka yang sesuai dengan citra yang dipilih. Form ini memiliki data berupa id data yang diperoleh dari memasukkan nilai angka, lokasi citra dan nilai eigen dari citra tersebut.



Gambar 8. Form Input Citra

5.3.3 Implementasi Form Pelatihan

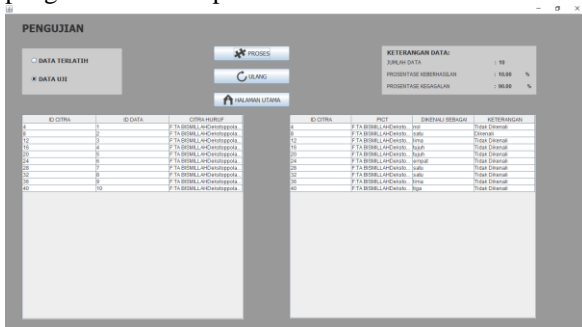
Form ini digunakan untuk melatih sistem agar dapat mengenali citra pada proses pengujian. Pada proses pelatihan pengguna diminta untuk memasukkan parameter yang akan digunakan untuk pelatihan, parameter tersebut berupa maksimal iterasi, konstanta belajar, tampilan iterasi, SSE dan hidden layer.



Gambar 9. Form Pelatihan

5.3.4 Implementasi Form Pengujian

Form Pengujian digunakan untuk menguji data baik yang sudah terlatih atau disebut data terlatih maupun data yang belum terlatih atau disebut data uji. Pada form ini terdapat hasil dari pengujian baik keberhasilan maupun kegagalan sistem melakukan pengenalan terhadap data citra.



Gambar 10. Form Pengujian Data Uji

5.3.5 Implementasi Form Tentang

Form adalah form yang hanya berisi informasi tentang identitas penulis dan informasi mengenai sistem pengenalan pola tulisan angka yang dibuat. Tampilan form pengujian dapat dilihat pada gambar 11.



Gambar 11. Form Tentang

6. PENUTUP

6.1. Kesimpulan

Dari hasil implementasi algoritma *backpropagation* pada sistem pengenalan pola angka tulisan tangan, penulis dapat mengambil kesimpulan sebagai berikut :

- Sistem pengenalan angka tulisan tangan dengan jaringan saraf tiruan dapat mengenali citra tulisan angka dengan cukup baik. Apabila menggunakan data latih yang sedikit akan menghasilkan proses pengenalan yang cepat, sedangkan dengan menggunakan banyak citra maka prosesnya akan lama.
- Proses pengujian (eksperimen 2 nilai eigen) dengan rentang nilai bobot -0,7 sampai 0,7, citra terlatih menghasilkan prosentase 20 persen dari 30 data, sedangkan untuk citra uji menghasilkan prosentase 10 persen dari 10 data uji.
- Proses pengujian (eksperimen 2 nilai eigen) dengan rentang nilai bobot -100 sampai 100, citra terlatih menghasilkan prosentase 70 persen dari 30 data, sedangkan untuk citra uji menghasilkan prosentase 30 persen dari 10 data uji.
- Proses pengujian (eksperimen 3 nilai eigen) dengan rentang nilai bobot -0,7 sampai 0,7, citra terlatih menghasilkan prosentase 13,33 persen dari 30 data, sedangkan untuk citra uji menghasilkan prosentase 20 persen dari 10 data uji.
- Proses pengujian (eksperimen 3 nilai eigen) dengan rentang nilai bobot -100 sampai 100, citra terlatih menghasilkan prosentase 76,67 persen dari 30 data, sedangkan untuk citra uji menghasilkan prosentase 10 persen dari 10 data uji.
- Eksperimen terbaik didapat dari 3 nilai eigen dan rentang nilai bobot antara -100 sampai 100.
- Prosentase ketepatan pada pengujian tergantung pada citra yang digunakan pada proses pelatihan.

6.2. Saran

Dari hasil implementasi sistem pengenalan pola angka tulisan tangan, penulis dapat memberikan saran untuk pembangunan atau pengembangan sistem selanjutnya. Saran tersebut sebagai berikut :

- a. Untuk meningkatkan akurasi pada sistem berikutnya perlu adanya penambahan pada data pelatihan atau data latih.
- b. Untuk pengembangan aplikasi pengenalan pola angka tulisan tangan angka dapat menggunakan pola angka dobel atau menggunakan pola huruf.

DAFTAR PUSTAKA

- [1] Avianto, D. (2016), *Pengenalan Pola Karakter Plat Nomor Kendaraan Menggunakan Algoritma Momentum Backpropagation Neural Network*, *Jurnal Informatika*, 10(1), 1199–1209.
- [2] David dan Kosasi, S. (2013), *Penerapan Algoritma Jaringan Saraf Tiruan Backpropagation Untuk Pengenalan Pola Tanda Tangan*, *Jurnal Teknologi*, 6(2).
- [3] Iswara, A.M. Purnomo, M.H. dan Sumpeno, S. (2013), *Aplikasi Pengenalan Kata pada Huruf Braille dan Pelafalannya*, *Jurnal Teknik ITS*, 2(2), A335–A340.
- [4] Handoyo, E.D. dan Susanto, L.W. (2011), *Penerapan Jaringan Syaraf Tiruan metode Propagasi Balik Dalam Pengenalan Tulisan Tangan Huruf Jepang Jenis Hiragana dan Katakana*, *Informatika*, 7(1), 1–112..
- [5] Pauzi, G.A. Warsito dan Zaitun (2015), *Sistem Identifikasi dan Pengenalan Pola Citra Tanda-Tangan Menggunakan Sistem Jaringan Saraf Tiruan (Artificial Neural Networks) Dengan Metode Backpropagation*, *JURNAL Teori dan Aplikasi Fisika FMIPA Universitas Lampung*, 03(02), 93–101.
- [6] Hermawan, A. (2006), *Jaringan Saraf Tiruan Teori dan Aplikasi*, Sigit Suyantoro, Ed. Yogyakarta: ANDI OFFSET.
- [7] Miranda, A.A. dan Borgne, L. (2007), *New Routes from Minimal Approximation Error to Principal Components*, Universit'e Libre de Bruxelles.
- [8] Faridh, M.M. (2013), *Pengenalan Karakter Huruf Tulisan Tangan Menggunakan Metode Principal Components Analysis*, Universitas Dian Nuswantoro Semarang.
- [9] Junaedi, H. Budianto, H. Maryati, I. dan Melani, Y. (2011), *Data transformation pada data mining*, *Prosiding Konferensi Nasional Inovasi dalam Desain dan Teknologi-IDEaTech*, 7, 93–99.
- [10] Sungkawa, I. dan Megasari, R.T. (2011), *Penerapan Ukuran Ketepatan Nilai Ramalan Data Deret Waktu Dalam Seleksi Model Peramalan Volume Penjualan PT Satriamandiri Citramulia*, *ComTech*, 2(2), 636–645.