

APLIKASI PENGENALAN AKSARA CARAKAN MADURA DENGAN MENGUNAKAN METODE *BACKPROPAGATION*

Erwien Tjipta W, Iswan Wahyu Al Farqi
STMIK ASIA Malang

ABSTRAK

Pengolahan citra (*Image Processing*) adalah suatu bidang yang berkembang sangat pesat sejalan dengan kemajuan teknologi pada saat ini.

Salah satu kasus yang bisa menerapkan bidang tersebut adalah pengenalan aksara carakan madura, proses pengenalan tersebut dengan mengambil gambar menggunakan webcam atau load gambar yang telah disiapkan dengan menulis di sebuah kertas atau mengambil file gambar yang telah ada di *hardisk*. Kemudian dilakukan proses pengolahan citra agar bisa melakukan segmentasi karakter, normalisasi, greyscale, binerisasi. Kemudian dari data ini dilakukan proses pelatihan dengan menggunakan jaringan syaraf tiruan (*Backpropagation*). Arsitektur jaringan dengan menggunakan jumlah *input* layer 256, jumlah *hiden* layer 3 dan *output* layer 1 neuron. Didapat keluaran berupa bobot, hasil bobot tersebut dibandingkan dengan array yang sudah ada untuk pengenalan karakter huruf, sehingga dapat membaca huruf aksara carakan madura.

Dari hasil penelitian yang didapat yaitu sebuah sistem pengenalan karakter aksara carakan Madura. Sistem ini telah dilakukan pengujian dengan arsitektur JST *Backpropagation* dengan jumlah data *training* sebanyak 400 yang mewakili 20 karakter. Hasil pengujian menunjukkan nilai keakuratan pengenalan karakter sebesar 78,333 %.

Kata kunci: Image Processing, Pengenalan Aksara Carakan Madura ,JST (Backpropagation).

ABSTRACT

Image processing is a field that is growing very rapidly in line with the current technological advances.

One of the cases that could apply that field is the introduction of characters Carakan Madura, the recognition process by taking a picture using a webcam or load an image that has been prepared by writing on a paper or take an existing image file on the hard drive. Then do the image processing in order to perform segmentation of characters, normalization, greyscale, binerisasi. Then this data is done with the training process using artificial neural networks (backpropagation). Network architecture using the input layer number 256, number 3 hidden layer and output layer 1 neurons. Can be output in the form of weights, weight results are compared with existing array for character recognition letters, so it can read the letter alphabet Carakan Madura.

From the results obtained, namely an alphabet character recognition system Carakan Madura. This system has been tested with Backpropagation neural network architecture with the amount of training data 400 representing 20 characters. The results show the value of character recognition accuracy of 78.333%.

Keywords: ImageProcessing, Introduction toCharacterCarakanMadura, JST(Backpropagation).

PENDAHULUAN

1. Latar Belakang

Seiring dengan perkembangan dan kemajuan teknologi yang semakin pesat, semua dituntut untuk menjadi lebih praktis. Dari hari ke hari semakin banyak perkembangan teknologi yang dijumpai dalam berbagai bidang. Salah satunya yaitu perkembangan mengenai proses pengolahan citra (*Image Processing*).

Dalam perkembangan proses pengolahan citra, banyak yang bisa dilakukan misalnya pengenalan huruf aksara carakan

Madura yang mana pada masa sekarang hampir dilupakan. Aksara Carakan Madura (aksara kecil) ada 20 macam banyaknya, yang dikasih nama aksara gajang karena masih belum ada pasangannya.

Pengenalan aksara carakan Madura diproses melalui pengolahan citra yaitu segmentasi, normalisasi, grayscale, dan binerisasi. Kemudian diolah menggunakan jaringan saraf tiruan. Jaringan saraf tiruan merupakan sebuah sistem pemroses informasi yang memiliki performa karakteristik tertentu seperti jaringan saraf biologi.

Pada penelitian ini metode yang digunakan adalah jaringan saraf tiruan *backpropagation*. Dengan menggunakan jaringan saraf tiruan metode *backpropagation* diharapkan dapat dibuat sistem komputer yang mampu mengenali pola huruf.

2. Rumusan Masalah

Berdasarkan latar belakang masalah diatas, penulis merumuskan masalah yaitu:

- a. Bagaimana merancang dan membuat aplikasi pengenalan aksara carakan Madura dengan menggunakan metode *backpropagation*?
- b. Berapa tingkat keberhasilan dalam pengenalan karakter aksara carakan Madura menggunakan metode *backpropagation*?

3. Batasan Masalah

Dalam perancangan ini, agar tidak menyimpang dari tujuan yang ingin di capai, maka pembahasan masalah di atas di batasi oleh beberapa hal, yaitu :

- a. Pembahasan hanya meliputi pengenalan 20 karakter aksara carakan Madura dan proses pengenalan hanya berdasarkan aksara asli tidak ada tambahan pasangan atau gatungan.
- b. Ukuran citra yang digunakan untuk proses identifikasi setelah di normalisasi memiliki ukuran 16 x 16 piksel.
- c. Proses input data dengan cara *capture* menggunakan *webcam* atau *load* gambar dari hardisk.
- d. Perangkat lunak hanya dapat memproses citra dengan format *Bitmap* (.bmp) dan *jpg*.
- e. Metode yang diterapkan adalah *backpropagation* dan Bahasa pemrograman yang digunakan adalah Visual Basic 6.0.
- f. Fungsi aktivasi yang digunakan adalah fungsi sigmoid biner
- g. Operasi citra yang digunakan yaitu segmentasi, normalisasi, grayscale, dan binerisasi.

KAJIAN TEORI

1. AI (*Artificial Intelligence*)

a. Sejarah AI (*Artificial Intelligence*)

Di awal abad 20, seorang penemu Spanyol, Torres Quevedo, membuat sebuah mesin yang dapat men"skak-mat" raja lawannya dengan sebuah ratu dan raja. Perkembangan secara sistematis kemudian dimulai segera setelah diketemukannya komputer digital. Artikel ilmiah tentang kecerdasan buatan ditulis

oleh Alan Turing pada tahun 1950, dan kelompok riset pertama dibentuk tahun 1954 di Carnegie Mellon University oleh Allen Newell dan Herbert Simon. Namun bidang kecerdasan buatan baru dianggap sebagai bidang tersendiri di konferensi Dartmouth tahun 1956, dimana sepuluh peneliti muda memimpikan mempergunakan komputer untuk memodelkan bagaimana cara berfikir manusia. Hipotesis mereka adalah "Mekanisme berfikir manusia dapat secara tepat dimodelkan dan disimulasikan pada komputer digital", dan ini yang menjadi landasan dasar kecerdasan buatan.

b. Kecerdasan Buatan (*Artificial Intelligence*)

Kecerdasan buatan (*Artificial Intelligence*) merupakan salah satu bagian ilmu komputer yang membuat agar mesin (komputer) dapat melakukan pekerjaan seperti dan sebaik yang dilakukan oleh manusia (**Kusumadewi, 2003**).

Kecerdasan buatan (*Artificial Intelligence*) adalah suatu teknologi atau teknik yang mensimulasikan kecerdasan manusia, yaitu bagaimana mendefinisikan dan mencoba menyelesaikan persoalan menggunakan komputer dengan meniru bagaimana cara manusia menyelesaikan persoalan yang ada (**Krose, 1996**).

Teknik yang digunakan dalam kecerdasan buatan memungkinkan dibuatnya sebuah program yang setiap bagiannya mengandung langkah-langkah penyelesaian masalah dan dapat diidentifikasi dengan baik untuk memecahkan sebuah atau sejumlah persoalan. Setiap potongan bagian program adalah seperti sepotong informasi dalam pikiran manusia. Jika informasi diabaikan, secara otomatis dapat mengatur cara kerjanya untuk menyesuaikan diri dengan fakta atau informasi yang baru. Tidak perlu selalu mengingat setiap potong informasi yang telah dipelajari. Hanya informasi yang relevan dengan persoalan yang dihadapi dan yang digunakan. Demikian juga dengan sistem kecerdasan buatan, setiap bagian program kecerdasan buatan dapat dimodifikasi tanpa mempengaruhi struktur seluruh programnya. Hal ini dapat menghasilkan program yang efisien dan mudah untuk dipahami. Ada 2 basis utama yang dibutuhkan untuk aplikasi kecerdasan buatan (**Krose, 1996**):

1. Basis pengetahuan (*knowledge base*): berisi fakta-fakta, teori, pemikiran dan hubungan antara satu dengan yang lain.
 2. Motor inferensi (*inference engine*): kemampuan menarik kesimpulan berdasarkan pengalaman.
- c. Cabang Kecerdasan Buatan
- Cabang kecerdasan buatan adalah sebagai berikut (Krose, 1996):
- a. Pencarian

Program AI seringkali harus mengevaluasi kemungkinan yang jumlahnya banyak sekali, misalnya kemungkinan langkah dalam permainan catur atau penyimpulan dari program untuk membuktikan suatu teori.
 - b. Pengenalan Pola

Pengenalan pola merupakan sebuah sistem atau teknik yang dapat mengenali pola dari suatu karakter, tulisan dan gambar.
 - c. Representasi

Representasi dalam kecerdasan buatan yaitu bagaimana mempresentasikan/menuliskan fakta-fakta yang ada ke dalam symbol-simbol atau bahasa logika matematis.
 - d. Perencanaan

Program perencanaan bermula dari fakta-fakta umum, terutama fakta mengenai efek dari suatu aksi, fakta tentang situasi yang khusus, dan suatu pernyataan tentang tujuan. Dari sini kemudian dibuat sebuah strategi untuk mencapai tujuan tersebut. Secara umum, biasanya strategi tersebut berupa urutan-urutan aksi.
 - e. Epistemologi

Empistemologi yaitu sebuah studi yang mempelajari tentang sumber, sifat, dan keterbatasan pengetahuan yang digunakan untuk pemecahan masalah.
 - f. Ontologi

Ontologi adalah ilmu tentang keberadaan dan realitas.
 - g. Heuristik

Heuristik adalah suatu cara atau teknik untuk mencoba menemukan ide.
2. Pengolahan Citra Digital
 1. Pengertian Pengolahan Citra Digital

Pengolahan citra (*Image Processing*) merupakan suatu sistem dimana proses dilakukan dengan masukan berupa citra (*image*) dan hasilnya juga berupa citra

(*image*). (Achmad, Basuki, Palandi, dkk, 2005).

Pada awalnya pengolahan citra ini dilakukan untuk memperbaiki kualitas citra, namun dengan berkembangnya dunia komputasi yang ditandai dengan semakin meningkatnya kapasitas dan kecepatan proses komputer serta munculnya ilmu-ilmu komputasi yang memungkinkan manusia dapat mengambil informasi dari suatu citra, maka *image processing* tidak dapat dilepaskan dengan bidang *computer vision*. (Achmad, Basuki, Palandi, dkk, 2005).

Sesuai dengan perkembangan komputer itu sendiri, pengolahan citra mempunyai dua tujuan utama, yaitu sebagai berikut:

- a. Memperbaiki kualitas citra, dimana citra yang dihasilkan dapat menampilkan informasi secara jelas. Hal ini berarti manusia sebagai pengolah informasi (*human perception*).
- b. Mengekstraksi informasi ciri yang menonjol pada suatu citra, dimana hasilnya adalah informasi citra dimana manusia mendapatkan informasi ciri dari citra secara numerik atau dengan kata lain *computer* (mesin) melakukan interpretasi terhadap informasi yang ada pada citra melalui besaran-besaran data yang dapat dibedakan secara jelas (berupa besaran numerik).

Dalam perkembangan lebih lanjut, *image processing* dan *computer vision* digunakan sebagai pengganti mata manusia dengan perangkat *input image capture* seperti kamera, scanner, dan mesin komputer dijadikan sebagai otak yang mengolah informasi. Sehingga muncul beberapa pecahan bidang yang menjadi penting dalam *computer vision* antara lain: *pattern recognition* (pengenalan pola), *biometric* (pengenalan identifikasi manusia berdasarkan ciri-ciri biologis yang tampak pada badan manusia), *content based image and video retrieval* (mendapatkan kembali citra atau video dengan informasi tertentu), *video editing* dan lain-lain. (Achmad, Basuki, Palandi, dkk, 2005).

Citra digital mengandung sejumlah elemen-elemen dasar. Elemen-elemen dasar tersebut dimanipulasi dalam pengolahan citra dan dieksploitasi lebih lanjut dalam *computer version*. Elemen-elemen yang penting diantaranya:

1. Kecerahan (*Brightness*) adalah kata lain untuk intensitas cahaya. Kecerahan pada sebuah titik (*pixel*) di dalam citra bukanlah intensitas yang asli, tetapi sebenarnya adalah intensitas rata-rata dari suatu area yang melingkupnya.
2. Kontras (*Contrast*) menyatakan bahwa dimana gambar tersebut terang (*Lightness*) atau gelap (*Darkness*). Citra dengan kontras rendah dicirikan oleh sebagian besar komposisi citranya adalah terang atau sebagian besar gelap. Pada citra dengan kontras yang baik, komposisi gelap dan terang tersebar secara merata.
3. Kontur (*Contour*) adalah keadaan yang ditimbulkan oleh perubahan intensitas pada piksel-piksel yang bertetangga. Karena adanya perubahan intensitas inilah mata kita mampu mendeteksi tepi-tepi (*edge*) obyek didalam citra.
4. Warna (*Color*) adalah persepsi yang dirasakan oleh system visual mampu terhadap panjang gelombang cahaya yang di pantulkan oleh obyek. Setiap warna mempunyai panjang gelombang yang berbeda. Warna-warna yang diterima oleh mata merupakan hasil kombinasi cahaya dengan panjang gelombang yang berbeda. Penelitian memperlihatkan bahwa kombinasi warna yang memberikan rentang warna yang paling lebar adalah *red* (R), *green* (G), dan *blue* (B).
5. Bentuk (*shape*) adalah properti intrinsik dari objek tiga dimensi, dengan pengertian bahwa *shape* merupakan properti intrinsik utama untuk sistem visual manusia. Pada umumnya citra yang dibentuk oleh mata merupakan citra *dwimatra* (dua dimensi), sedangkan objek yang dilihat umumnya berbentuk *trimatra* (tiga dimensi). Informasi bentuk objek dapat diekstraksi dari citra pada permulaan prapengolahan dan segmentasi citra.
6. Tekstur (*Texture*) Pada hakekatnya system visual manusia tidak menerima informasi citra secara terpisah pada setiap titik, tetapi suatu citra dianggapnya sebagai satu kesatuan. Jadi definisi kesamaan suatu obyek perlu dinyatakan dalam bentuk kesamaan dari satu himpunan parameter citra (*Brightness, Color, Size*). Atau dengan kata lain dua buah citra tidak dapat disamakan hanya dengan satu parameter saja.

7. Waktu dan Pergerakan respon suatu sistem visual tidak hanya berlaku pada faktor ruang, tetapi juga pada faktor waktu. Sebagai contoh, bila citra-citra diam ditampilkan secara cepat, akan berkesan melihat citra yang bergerak.
8. Deteksi dan Pengenalan dalam mendeteksi dan mengenali suatu citra, ternyata tidak hanya sistem visual manusia saja yang bekerja, tetapi juga ikut melibatkan ingatan dan daya pikir manusia.

2. *Capture*

Proses *capture* adalah proses pengambilan gambar melalui kamera. Teknik pengambilan gambar yang digunakan adalah meletakkan resistor yang ingin diketahui nilainya pada tempat yang sudah disediakan di depan kamera. Agar mempermudah dalam pengambilan gambar dan pengolahannya, maka kamera yang digunakan adalah kamera digital atau sebuah *webcam*.



GambarCitra *Capture*

3. Derajat Keabuan (*Gray Scale*)

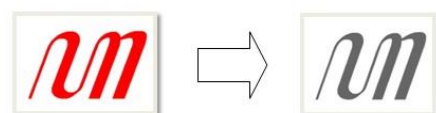
Proses mengubah citra warna menjadi citra keabuan digunakan dalam pemrosesan citra yang bertujuan untuk menyederhanakan model citra.

Untuk mengkonversi dari citra warna ke *grayscale* dapat dilakukan dengan berbagai macam metode, salah satunya adalah perhitungan skala keabuan (*grayscale*) menggunakan metode rata-rata seperti yang dirumuskan pada persamaan berikut ini.

$$Grey = \frac{R + G + B}{3}$$

Dimana *Grey* merupakan nilai keabuan pada suatu piksel, sedangkan R (*Red*), G (*Green*), dan B (*Blue*) merupakan tiga warna dasar yang dimiliki tiap piksel pada citra *true color* (citra warna). Perhitungan nilai skala keabuan (*grayscale*) juga dapat dilakukan dengan perubahan komposisi seperti pada persamaan berikut ini.

$$Grey = 0.42 * R + 0.32 * G + 0.28 * B$$



GambarCitra *Grayscale*

4. Thresholding

Thresholding digunakan untuk mengatur derajat keabuan yang ada pada suatu citra. Proses *thresholding* mengelompokkan nilai derajat keabuan setiap piksel ke dalam dua kelas, hitam dan putih (Munir, 2004).

Teknik *thresholding* ini memisahkan bagian gambar yang sesuai dengan obyek yang diteliti dan latar belakangnya. Metode ini bertujuan agar proses pengolahan citra dapat dilakukan dengan lebih mudah. Proses *thresholding* ini pada dasarnya adalah proses pengubahan kuantisasi pada citra, sehingga untuk melakukan *thresholding* dengan derajat keabuan dapat digunakan

$$\text{Rumus . } x = b.\text{int}\left(\frac{256}{a}\right)$$

x adalah nilai derajat keabuan *thresholding* a adalah derajat keabuan yang diinginkan

Fungsi Thresholding secara umum dinyatakan dengan persamaan berikut ini.

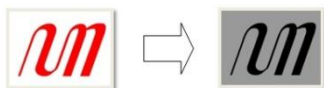
$$f'(i, j) \begin{cases} 1, f(i, j) \leq x \\ 0, \text{lainnya} \end{cases}$$

$f(i, j)$ = Citra awal

$f'(i, j)$ = Citra hasil *thresholding*

x = Nilai ambang *thresholding*

Dengan operasi *thresholding*, obyek dibuat berwarna gelap (1 atau hitam) sedangkan latar belakang berwarna terang (0 atau putih).



Gambar Citra *Threshold*

5. Segmentasi Karakter

Segmentasi atau pemenggalan karakter dilakukan untuk memisahkan karakter-karakter pada suatu citra. Segmentasi karakter perlu dilakukan sebelum proses pengenalan karakter, citra yang akan disegmentasi berupa citra biner (hitam-putih).

Proses segmentasi dapat dilakukan dengan berbagai metode, salah satunya dengan menghitung jumlah piksel putih pada suatu citra secara mendarat dan juga secara menurun (Munir, 2004).

Segmentasi karakter merupakan sebuah proses dimana dalam proses

tersebut terjadi pemisahaan objek-objek pada suatu gambar yang telah dipilih. Secara umum segmentasi dapat dibagi menjadi tiga kelompok yaitu:

- Segmentasi berdasar klasifikasi (classification based segmentation), segmentasi ini merupakan segmentasi yang dilakukan dengan mencari kesamaan dari ukuran tertentu pada nilai piksel yang terdapat pada suatu gambar. Salah satu proses yang paling mudah menggunakan segmentasi ini adalah dengan cara *thresholding*.
- Segmentasi berdasar tepi (edge base segmentation), sedangkan untuk segmentasi berdasarkan tepi bertujuan untuk mendapatkan batas tepi antar objek yang berguna sebagai pembatas antar segmen.
- Segmentasi berdasar daerah (region based segmentation), untuk segmentasi berdasarkan daerah dilakukan untuk mencari daerah yang di duga sebagai objek berdasarkan kumpulan pixel yang memiliki kesamaan yang dimulai dari suatu titik ke titik yang lain yang ada di sekitarnya.



Gambar 2.7 Citra Segmentasi Karakter

6. Normalisasi (*Scaling*)

Scaling adalah fungsi yang mengubah ukuran suatu gambar, dimana *scaling* biasanya cenderung merupakan sebutan untuk perbesaran ukuran gambar dan *shrink* cenderung merupakan sebutan untuk memperkecil ukuran gambar. (Munir, 2004).



Gambar Citra Normalisasi Karakter

7. Citra Biner

Citra biner adalah sebuah citra hitam putih yang fungsinya untuk pengenalan pola, yaitu proses pengambilan piksel pada gambar dimana hanya ada dua warna yaitu hitam dan putih. Pengambilan nilai tersebut untuk mendapatkan pola pada gambar tersebut.

Setiap piksel dalam citra bernilai 0 dan 1. (warna hitam 1, putih 0).

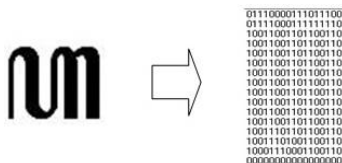
1. Model citra cahaya jika ada cahaya (=0) maka warna putih.

2. Model citra cahaya jika tidak ada cahaya (=1) maka warna hitam.
3. Model citra tinta / cat jika ada cat (=1) maka warna hitam.
4. Model citra tinta / cat jika tidak ada cat (=0) maka warna putih.
5. Setiap titik membutuhkan media penyimpanan 1 bit.

Sebenarnya prosesnya sama dengan threshold yaitu mengubah kuantisasi citra. Untuk citra dengan derajat keabuan 256, maka nilai tengahnya adalah 128 sehingga untuk mengubah citra biner dapat dituliskan :

Jika $x < 128$ maka $x = 0$, jika tidak maka $x = 255$

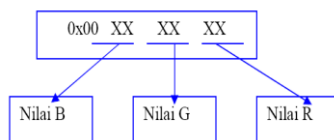
Sumber : (Achmad, Basuki, Palandi, dkk, 2005).



Gambar Citra biner Karakter

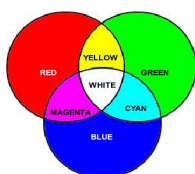
8. Representasi Warna RGB (Red, Green, Blue)

Dasar dari pengolahan citra adalah pengolahan warna RGB pada posisi tertentu. Dalam pengolahan citra, warna direpresentasikan dengan nilai hexadecimal dari 0x00000000 sampai dengan 0x00ffffff. Warna hitam dengan nilai 0x00000000, dan warna putih bernilai 0x00ffffff.



Gambar Warna RGB dalam Hexadesimal

Nilai warna pada gambar 2.10 merupakan gabungan antara 3 warna dasar yaitu merah (Red), hijau (Green), biru (Blue), seperti tampak pada gambar 2.11 untuk memperoleh warna lain selain warna primer, perlu menggabungkan skala kecerahan dari tiap warna.

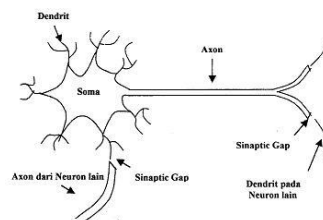


Gambar Komposisi warna RGB

Sumber : (Ahmad dan Firdausy, 2005).

3. Jaringan Syaraf Tiruan

Pada tahun 1940-an ilmuwan menemukan bahwa psikologis dari otak sama dengan metode pemrosesan yang dilakukan oleh peralatan komputer. Pada tahun 1943 seorang ahli saraf Warren McCulloch dan seorang ahli logika Walter Pitts merancang model formal yang pertama kali sebagai perhitungan dasar *neuron*. Jaringan saraf tiruan merupakan model yang meniru cara kerja jaringan *neuron* biologis. Penelitian yang berlangsung pada tahun 1950-an dan 1960-an mengalami hambatan karena minimnya kemampuan komputer pada saat itu. Sekitar tahun 1970-an penelitian di bidang ini terhenti sama sekali. Baru pada pertengahan tahun 1980-an dapat dilanjutkan lagi, karena sarana yang dibutuhkan telah tersedia. *Neuron* tiruan dirancang untuk menirukan karakteristik *neuron* biologis. (Siang, Jek, 2005).

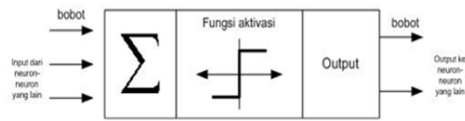


Gambar Jaringan Syaraf Biologi

Prinsip dasar jaringan saraf tiruan adalah: "Jaringan saraf tiruan merupakan sebuah sistem yang terdiri dari unit-unit yang saling berhubungan, dimana setiap unit ini disebut sebagai *Artificial Neuron*". Prinsip ini diterapkan berdasarkan asumsi yang diambil dari jaringan saraf biologis yang dimiliki otak manusia. Adapun asumsi-asumsi tersebut meliputi :

- a. Pemrosesan informasi terjadi pada elemen-elemen sederhana yang disebut *neuron*.
- b. Sinyal-sinyal yang mengalir *neuron-neuron* melewati hubungan *links*.
- c. Setiap *links* penghubung memiliki bobot yang bersesuaian, yang dalam suatu jaringan saraf menggandakan sinyal yang ditransmisikan.
- d. Setiap *neuron* menerapkan suatu fungsi aktivasi (biasanya non linier) ke *input* jaringan untuk menentukan sinyal *outputnya*.

Jadi, *Artificial Neuron* (AN) merupakan model matematika dari *neuron* yang dimiliki oleh otak manusia, dimana pada otak manusia terjadi proses *electrochemical* sedangkan pada jaringan saraf tiruan proses komputasi adalah dengan sebuah aktivasi.



Gambar Struktur *Neuron* Jaringan Syaraf Tiruan

Neuron buatan ini sebenarnya mirip dengan sel neuron biologis. Neuron-neuron buatan tersebut bekerja dengan cara yang sama pula dengan neuron-neuron biologis. Informasi (*input*) akan dikirim ke neuron dengan bobot kedatangan tertentu. Input ini akan diproses oleh suatu fungsi perambatan yang akan menjumlahkan nilai-nilai semua bobot yang datang. Hasil penjumlahan ini kemudian akan dibandingkan dengan suatu nilai ambang (*threshold*) tertentu melalui fungsi aktivasi setiap neuron. Apabila input tersebut melewati suatu nilai ambang tertentu, maka neuron tersebut akan diaktifkan, tapi kalau tidak, maka neuron tersebut tidak diaktifkan. Apabila neuron tersebut diaktifkan, maka neuron tersebut akan mengirimkan keluaran (*output*) melalui bobot-bobot keluarannya ke semua neuron yang berhubungan dengannya.

Pada jaringan syaraf, neuron-neuron akan dikumpulkan dalam lapisan-lapisan (*layer*) yang disebut dengan lapisan neuron (*neuron layer*). Biasanya neuron-neuron pada satu lapisan akan dihubungkan dengan lapisan-lapisan sebelum dan sesudahnya (kecuali lapisan *input* dan lapisan *output*). Informasi yang diberikan pada jaringan syaraf akan dirambatkan lapisan ke lapisan, mulai dari lapisan input sampai ke lapisan output melalui lapisan yang lainnya, yang sering disebut dengan nama lapisan tersembunyi (*hidden layer*). Tergantung pada algoritma pembelajarannya, bisa jadi informasi tersebut akan dirambatkan secara mundur pada jaringan. (Siang,Jek, 2005).

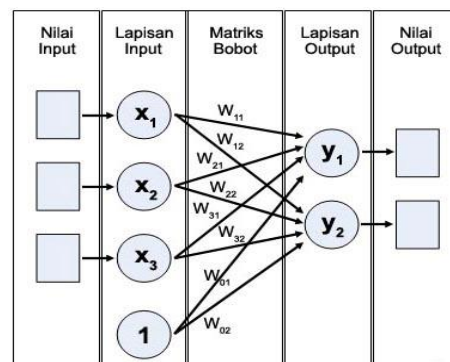
1. Arsitektur Jaringan Syaraf Tiruan

Seperti telah dijelaskan sebelumnya bahwa *neuron-neuron* dikelompokkan dalam lapisan-lapisan. Umumnya, *neuron-neuron* yang terletak pada lapisan yang sama akan memiliki keadaan yang sama. Faktor terpenting dalam menentukan perilaku suatu *neuron* adalah fungsi aktivasi dan pola bobotnya. Pada setiap lapisan yang sama, *neuron-neuron* akan memiliki fungsi aktivasi yang sama. Apabila neuron-neuron dalam suatu lapisan (misalkan lapisan tersembunyi)

akan dihubungkan dengan *neuron-neuron* pada lapisan yang lain (misalkan lapisan output), maka setiap neuron pada lapisan tersebut (misalkan lapisan tersembunyi) juga harus dihubungkan dengan setiap lapisan pada lapisan lainnya (misalkan lapisan output). (Siang, Jek, 2005).

a. Arsitektur Jaringan dengan Lapisan Tunggal (*Single Layer Net*).

Jaringan dengan lapisan tunggal hanya memiliki satu lapisan dengan bobot-bobot terhubung. Jaringan ini hanya menerima *input* kemudian secara langsung akan mengolahnya menjadi *output* tanpa harus melalui lapisan tersembunyi. Pada Gambar 2.14 tersebut, lapisan input memiliki 3 *neuron*, yaitu X_1 , X_2 , dan X_3 . Sedangkan pada lapisan *output* memiliki 2 *neuron* yaitu Y_1 dan Y_2 . *Neuron-neuron* pada kedua lapisan saling berhubungan. Seberapa besar hubungan antara dua *neuron* ditentukan oleh bobot yang bersesuaian. Semua unit akan dihubungkan dengan setiap unit *output*. (Siang,Jek, 2005).

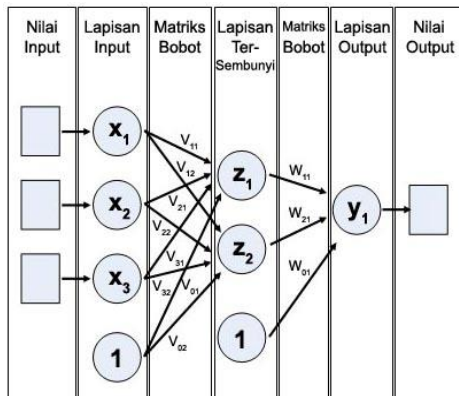


Gambar Jaringan Syaraf Dengan Lapisan Tunggal

b. Arsitektur Jaringan dengan Banyak Lapisan (*Multilayer Net*)

Jaringan dengan banyak lapisan memiliki 1 atau lebih lapisan yang terletak diantara lapisan input dan lapisan output (memiliki 1 atau lebih lapisan tersembunyi), seperti terlihat pada Gambar 2.15 umumnya, ada lapisan bobot-bobot yang terletak antara 2 lapisan yang bersebelahan. Jaringan dengan banyak lapisan ini dapat menyelesaikan permasalahan yang lebih sulit daripada lapisan dengan lapisan tunggal, tentu saja dengan pembelajaran yang lebih rumit. Namun demikian, pada banyak kasus, pembelajaran dengan banyak

lapisan ini lebih sukses dalam menyelesaikan masalah. (Siang, Jek, 2005).



Gambar Jaringan Syaraf Dengan Banyak Lapisan

4. Metode Backpropagation

Kelemahan jaringan saraf tiruan yang terdiri dari lapisan tunggal membuat perkembangannya menjadi terhenti pada sekitar tahun 1970an. Penemuan *backpropagation* yang terdiri dari beberapa lapisan membuka kembali cakrawala, terlebih setelah berhasil ditemukannya berbagai aplikasi yang dapat diselesaikan dengan *backpropagation*, membuat jaringan saraf tiruan semakin diminati orang. (Siang, Jek, 2005).

Backpropagation merupakan algoritma pembelajaran yang terawasi dan biasanya digunakan oleh *perceptron* dengan banyak lapisan untuk mengubah bobot-bobot yang terhubung dengan *neuron-neuron* yang ada pada bagian tersembunyi. *Perceptron* biasanya digunakan untuk mengklasifikasikan suatu tipe pola tertentu yang sering dikenal dengan pemisahan secara *linier*. Pada dasarnya, *perceptron* pada jaringan saraf tiruan dengan satu lapisan memiliki bobot yang bisa diatur dan suatu nilai ambang (*threshold*). Algoritma yang digunakan oleh aturan *perceptron* ini akan mengatur parameter-parameter bebasnya melalui proses pembelajaran.

Algoritma *backpropagation* menggunakan error *output* untuk mengubah nilai bobot-bobotnya dalam arah mundur (*backward*). Untuk mendapatkan error ini, tahap perambatan maju (*forward propagation*) harus dikerjakan terlebih dahulu. Pada saat perambatan maju, *neuron-neuron* diaktifkan dengan menggunakan fungsi aktivasi *sigmoid* dan *biner sigmoid*.

Pada dasarnya algoritma *backpropagation* terbagi menjadi tiga tahap, yaitu langkah maju (*freedforward*), propagasi

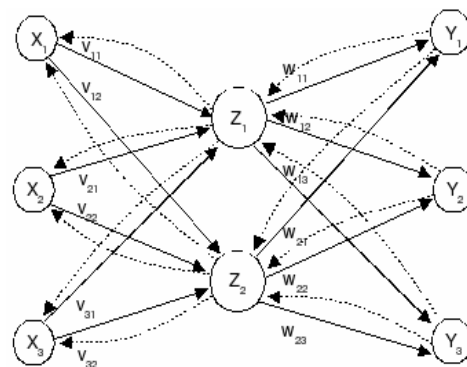
balik (*backpropagation*), dan perubahan nilai bobot. Pada saat *freedforward*, perhitungan nilai bobot *neuron* hanya didasarkan pada vektor masukan, sedangkan pada *backpropagation*, nilai bobot diperhalus dengan memperhitungkan nilai target atau keluaran.

Setiap unit *layer input* pada jaringan *backpropagation* selalu terhubung dengan setiap unit yang berbeda pada layer yang tersembunyi, demikian juga setiap unit pada layer tersembunyi selalu terhubung dengan unit layer *output*. Jaringan *backpropagation* terdiri dari banyak lapisan (*multilayer network*), yaitu :

1. Lapisan input (1 buah), yang terdiri dari 1 hingga p unit.
2. Lapisan tersembunyi (minimal 1 buah), yang terdiri dari 1 hingga p unit tersembunyi.
3. Lapisan *output* (1 buah), yang terdiri m unit *output*.

Nilai *mean square error* (MSE) pada suatu perulangan pelatihan dihitung sebagai jumlah kuadrat dari *error* (*error* = nilai keluaran - nilai masukan). Kemudian dibagi dengan jumlah seluruh data yang dipresentasikan ke jaringan syaraf tiruan. Secara matematis dapat dituliskan sebagai berikut:

$$MSE = 0,5 \times \{(tk_1 - y_{k1})^2 + (tk_2 - y_{k2})^2 + \dots + (tk_m - y_{km})^2\}$$



Gambar Arsitektur Jaringan Backpropagation

Keterangan :

X = Masukan (*input*).

J = 1 s/d n.

V = Bobot pada *hidden layer*.

W = Bobot pada *output layer*.

n = Jumlah unit pengolahan pada *hidden layer*.

b = Bias pada *hidden layer* dan *output layer*.

k = Jumlah unit pengolahan pada *output layer*.

Y = Keluaran hasil.

Algoritma *Backpropagation* :

Input : Data pelatihan, jumlah lapis, jumlah neuron, target error, learning rate

Output : Model JST yang siap mengolah data baru.

1. Bobot dengan bilangan acak, MSE, diinisialisasi
2. Selama MSE > dilakukan :
3. Untuk tiap-tiap pasangan elemen yang akan dilakukan pembelajaran, dikerjakan.

Feedforward :

4. Tiap-tiap unit input ($X_i, i = 1,2,3,\dots,n$) menerima sinyal x_i dan meneruskan sinyal tersebut ke semua unit pada lapisan yang ada di atasnya (lapisan tersembunyi).
5. Tiap-tiap unit tersembunyi ($Z_j, j = 1,2,3,\dots,p$) menjumlahkan sinyal-sinyal input dengan bobot :

$$z_in_j = v_{0j} + \sum_{i=1}^n x_i v_{ij}$$

menggunakan fungsi aktivasi untuk menghitung sinyal outputnya:

$$z_j = f(z_in_j)$$

dan kirimkan sinyal tersebut ke semua unit di lapisan atasnya (unit-unit output).

6. Tiap-tiap unit output ($Y_k, k = 1,2,3,\dots,m$) menjumlahkan sinyal-sinyal input dengan bobot :

$$y_in_k = w_{0k} + \sum_{i=1}^p z_i w_{ik}$$

Menggunakan fungsi aktivasi untuk menghitung sinyal outputnya:

$$y_k = f(y_in_k)$$

Dan sinyal tersebut dikirim ke semua unit di lapisan atasnya (unit-unit output).

Backpropagation :

7. Tiap-tiap unit output ($Y_k, k = 1,2,3,\dots,m$) menerima target pola yang berhubungan dengan pola input dari pembelajaran, informasi errornya dihitung:

$$\delta_k = (t_k - y_k) f'(y_in_k)$$

Kemudian koreksi bobot (yang nantinya akan digunakan untuk memperbaiki nilai w_{jk}) dihitung:

$$\Delta w_{jk} = \alpha \delta_k z_j$$

Koreksi bias (yang nantinya akan digunakan untuk memperbaiki nilai w_{0k}) dihitung juga:

$$\Delta w_{0k} = \alpha \delta_k$$

δ_k ini dikirim ke unit-unit yang ada di lapisan bawahnya.

8. Tiap-tiap unit tersembunyi ($Z_j, j = 1,2,3,\dots,p$) menjumlahkan delta inputnya (dari unit-unit yang berada pada lapisan atasnya):

$$\delta_in_j = \sum_{k=1}^m \delta_k w_{jk}$$

Nilai ini dikalikan dengan turunan dari fungsi aktivasinya untuk menghitung informasi error :

$$\delta_j = \delta_in_j f'(\delta_in_j)$$

kemudian koreksi bobot (yang nantinya akan digunakan untuk memperbaiki nilai v_{ij}) dihitung :

$$\Delta v_{ij} = \alpha \delta_j x_i$$

Koreksi bias (yang nantinya akan digunakan untuk memperbaiki nilai v_{0j}) dihitung juga :

9. Tiap-tiap unit output ($Y_k, k = 1,2,3,\dots,m$) memperbaiki bias dan bobotnya ($j = 0,1,2,\dots,p$) :

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta w_{jk}$$

Tiap-tiap unit tersembunyi ($Z_j, j = 1,2,3,\dots,p$) memperbaiki bias dan bobotnya ($i = 0,1,2,\dots,n$) :

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij}$$

MSE dihitung

$$MSE = 0,5 \times \{ (t_{k1} - y_{k1})^2 + (t_{k2} - y_{k2})^2 + \dots + (t_{km} - y_{km})^2 \}$$

10. Tes kondisi berhenti.

Sumber : (Siang, Jek, 2005).

Keterangan simbol - simbol yang digunakan berikut ini :

X = Vektor input.

Y = Vektor output.

δ_k = Nilai koreksi kesalahan untuk lapisan keluaran.

δ_j = Nilai koreksi kesalahan untuk lapisan bersembunyi.

α = Tingkat pembelajaran.

X_i = Unit ke-i pada lapisan masukan.

v_{j0} = Nilai bias unit ke-j pada lapisan tersembunyi.

Z_j = Unit ke-j pada lapisan tersembunyi.

w_{k0} = Nilai bias unit keluaran.

Y_k = Unit ke-k pada lapisan keluaran.

Err = Nilai kesalahan pada proses pembelajaran.

Tk = Nilai target.

Yk = Nilai keluaran hasil *feedforward*.

GambarAksara Carakan Madura

5. Aksara Carakan Madura

Selama mengenyam studi dibangku SD sampai SMP saya mendapatkan yang namanya pelajaran Bahasa Madura. Walaupun saya asli madura pelajaran ini cukup menyulitkan sehingga membuat lidah puter kiri dan puter kanan. Lebih-lebih kalau sudah mememasuki huruf asli bahasa madura yang sarat dengan aturan begini dan begitu. Kalau Jepang punya Katakana dan Hiragana, China punya Pinyin atau Hanzhi, Jawa punya Aksara Jawa Kuno (Honocoroko), maka Madura tentu punya huruf original. Namanya sudah tak asing dan aneh ditelinganya orang Madura tentunya "Akasara Carakan Madura" (Anacaraka).

Sebenarnya Aksara Carakan Madura juga disebut Aksara Jawa, karena memang mirip bentuknya. Namun Aksara Carakan Madura telah dipatenkan menjadi aksara asli Madura karena sedikit lebih variatif dari pada Aksara Jawa. Aksara ini juga digunakan di Bali dan Sunda. Bahkan ditemukan pula surat - surat dalam bahasa melayu yang menggunakan tulisan carakan. Tulisan ini berasal dari pada tulisan kawi yang mempunyai asal usul dari tulisan Brahmi di India. Anacaraka dinamakan sedemikian karena lima huruf pertamanya membentuk sebutan "a na ca ra ka". Anacaraka juga boleh merujuk kepada kelompok sistem tulisan yang berkait rapat dengan tulisan jawa dan menggunakan susunan abjad yang sama yaitu tulisan Jawa sendiri, tulisan Bali, dan tulisan Sunda sendiri. Terdapat 20 huruf dalam aksara madura, yang mana boleh disingkatkan kepada kata "ana caraka, data sawala, padha jhayanya, magha bhathanga", dalam bahasa madura bermaksud "dua utusan, meraka berbala, dua-dua sama hebat, dua-dua sama mati".

Aksara ghajang (Madura) adalah aksara yang berdiri sendiri karena belum ada pasangannya sering kali disebut dengan aksara kene' karena ditulis dengan huruf kecil (bukan huruf kapital). Aksara ini ada 20 macam yaitu:

a	na	ca	ra	ka
da / dâ	ta	sa / sâ	wa / wâ	la / lâ
pa	dha / dhâ	jâ / jhâ	ya / yâ	nya
ma	gâ / ghâ	bâ / bhâ	tha	nga

METODE PENELITIAN

1. Analisa Masalah

Sistem pengenalan tulisan merupakan sebuah sistem yang dapat melakukan pengenalan terhadap citra yang berisi tulisan atau karakter. Pada tugas akhir ini, sistem yang akan dibuat terdiri dari dua tahap, yaitu, tahap pelatihan (*training*) dan tahap pengenalan.

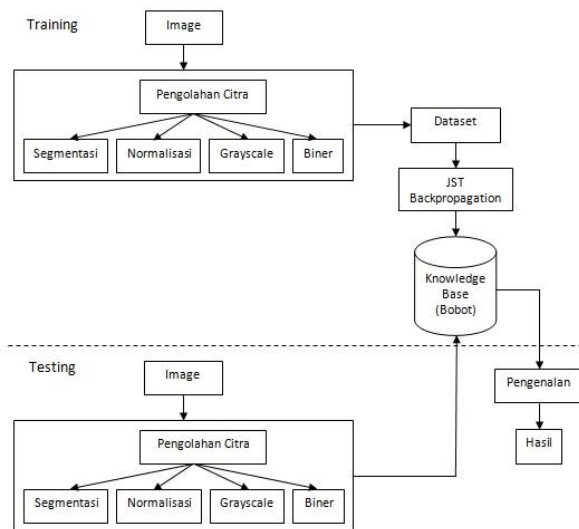
Tahap pelatihan atau pembelajaran merupakan tahap pertama sebelum dapat melakukan pengenalan, dimana dalam tahap ini sistem akan melakukan pembelajaran terhadap huruf yang dimasukkan ke dalam sistem. Huruf yang dimasukkan ke dalam sistem berupa huruf aksara carakan madura dengan proses pengambilan gambar menggunakan *webcam* atau dengan menggunakan *load* gambar.

Pada tahap pengenalan, sistem akan melakukan pengenalan terhadap citra yang di *inputkan* oleh *user* dengan cara mengambil gambar menggunakan *webcam* atau *load* gambar. Dalam tahap ini data gambar yang di *input* oleh *user* akan dicocokkan dengan data pengetahuan hasil dari tahap sebelumnya yaitu proses pembelajaran.

Dalam setiap tahap diatas, terdapat satu proses yang sangat penting terhadap hasil pengenalan maupun hasil pembelajaran. Proses yang dimaksud adalah proses pengolahan citra (*image processing*). Proses pengolahan citra ini sangat menentukan karena sebuah citra harus diproses dengan baik terlebih dahulu sebelum dilakukan perhitungan-perhitungan yang ada dalam proses pembelajaran maupun pada proses pengenalan.

2. Gambaran Umum Sistem

Untuk memahami bagaimana cara kerja sistem yang dibuat, terlebih dahulu diuraikan fungsi dari tiap diagram blok.



Gambar Blok Diagram Sistem Secara Keseluruhan

Pada gambar diatas dijelaskan mengenai gambar blok diagram sistem secara keseluruhan, yaitu proses pertama yang dilakukan input data yang berupa gambar yang kemudian di olah dengan proses pengolahan citra. Hingga menjadi dataset biner kemudian di proses dengan *training* jaringan syaraf tiruan *Backpropagation* yang nantinya data tersebut akan disimpan sebagai pengetahuan (knowledge base). Proses selanjutnya yaitu apabila user menginputkan data baru maka akan dilakukan pencocokan atau pengujian data dan akan menghasilkan sebuah nilai akurasinya.

3. Perancangan Sistem

Setelah dilakukan analisis sistem, tahap berikutnya adalah perancangan sistem, secara garis besar aplikasi ini dibagi menjadi tiga bagian, yaitu pengolahan citra digital (*preprocessing*), pembelajaran jaringan syaraf tiruan, dan pengenalan karakter.

Dalam proses pengolahan citra digital, sebuah data *input* dalam aplikasi ini berupa *bitmap* (.bmp) dan *jpg image*. Di tahap ini *image* akan diolah terlebih dahulu sebelum nantinya akan dilakukan proses pengenalan karakter satu persatu.

Pada proses pembelajaran jaringan syaraf tiruan, terlebih dahulu adalah membuat jaringan syaraf tiruan itu sendiri. Dalam pembuatannya, akan ditentukan berapa jumlah keluaran dan masukan pada jaringan tersebut, dan jumlah *hidden layer*.

Proses selanjutnya adalah pengenalan karakter. setelah citra masukan telah diolah pada proses pengolahan citra digital dan jaringan syaraf tiruan sudah dilatih, maka proses pengenalan bisa dilakukan.

4. Analisa Kebutuhan

Gambaran umum yang terdapat dalam sistem ini dibagi menjadi tiga bagian, yaitu:

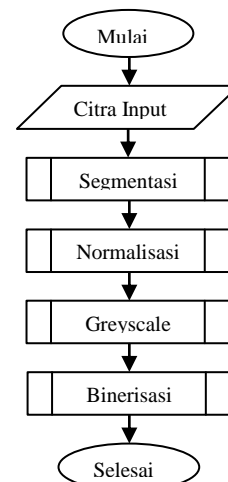
1. Analisa Kebutuhan *Input*

Dalam proses pengenalan aksara carakan madura ini, metode untuk memasukkan citra kedalam sistem yaitu dengan *capture* menggunakan *webcam*. Selain itu, citra yang akan dimasukkan kedalam sistem bisa juga diperoleh dari hasil *load* gambar yang ada didalam media penyimpanan atau *hardisk*.

1. Gambar yang ingin diproses dengan ukuran 300 x 300 piksel.
2. Type citra 24 bit (RGB).

2. Pengolahan Citra (*Image Processing*)

Seperti yang telah dijelaskan pada bahasan sebelumnya pengolahan citra merupakan satu proses yang sangat berpengaruh terhadap hasil pengenalan maupun pembelajaran huruf aksara carakan madura.



Flowchart Proses Pengolahan Citra

a. Segmentasi

Proses segmentasi digunakan untuk mencari banyaknya huruf dari setiap kata. Pada huruf dari setiap kata dilakukan proses pemotongan atau pemenggalan tiap karakter.

b. Segmentasi

Proses normalisasi digunakan untuk memperkecil citra huruf dari hasil segmentasi. Hal ini dilakukan agar proses pengidentifikasian dengan *Backpropagation* tidak berjalan terlalu lama.

Yang pertama kali dilakukan adalah menentukan skala dari hasil

citra hasil segmentasi agar ukuran yang diinginkan, yaitu 16x16 piksel. Dengan demikian, ukuran citra hasil *cropping* yang awalnya 300x300 piksel setelah dikalikan skala menjadi 16x16 piksel.

c. *Grayscale*

Pada proses ini, citra hasil normalisasi disederhanakan dengan cara mencari rata-rata nilai RGB. Hal ini perlu dilakukan sebelum memulai proses ekstraksi. Cara mencari nilai rata-rata RGB adalah dengan menjumlahkan warna *red*, *green*, dan *blue* dalam satu piksel dibagi tiga.

Selanjutnya hasil rata-rata tersebut diset menjadi nilai RGB yang baru.

d. Binerisasi

Tahap pra proses yang terakhir adalah memproses hasil *grayscale* menjadi citra biner (citra yang mempunyai dua nilai, yaitu nilai 0 dan 1). Citra biner ini sangat diperlukan sebagai input untuk Jaringan Syaraf Tiruan *Backpropagation*. Nantinya biner ini yang akan disimpan ke dalam *database* dan digunakan sebagai bobot koneksi untuk mengenali pola yang baru.

3. Analisa Kebutuhan Output

Adapun proses dalam perancangan output yaitu berupa hasil yang diperoleh dari proses pengolahan citra (*image processing*) yang kemudian akan dilanjutkan proses perhitungan sistem pembelajaran algoritma jaringan syaraf tiruan *backpropagation* agar dapat mengenali karakter aksara carakan madura dengan baik.

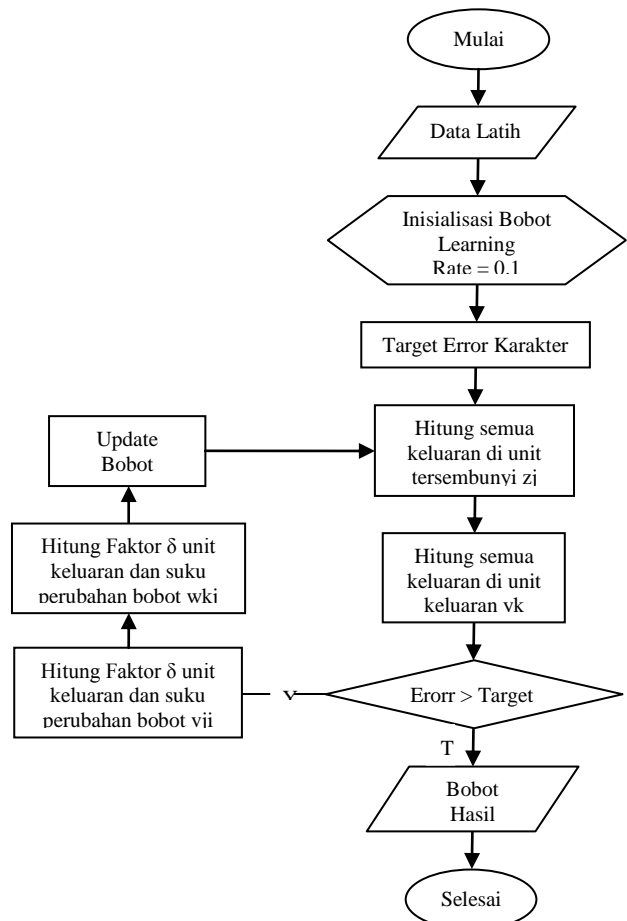
5. Proses Jaringan Syaraf Tiruan

1. Algoritma Pelatihan *Backpropagation*

Berikut tahap-tahap pelatihan *backpropagation* pada aplikasi ini :

1. Mulai.
2. Data pelatihan
3. Inialisasi bobot, *learning rate* = 0,1.
4. Target error karakter
5. Hitung semua keluaran di unit tersembunyi zj.
6. Hitung semua keluaran di unit keluaran yk.
7. Hitung MSE
8. Periksa apakah kuadrat error > target, jika ya maka lakukan langkah ke 9-12, jika tidak maka lakukan langkah ke-

- 12.
9. epoch + 1
10. Hitung faktor δ unit keluaran dan suku perubahan bobot w_{kj}
11. Hitung faktor δ unit tersembunyi dan suku perubahan bobot v_{ji}
12. Update bobot.
13. Simpan bobot akhir.
14. Selesai.



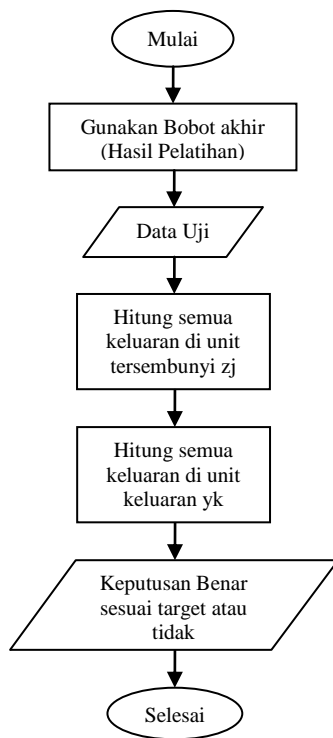
Flowchart Pelatihan

2. Algoritma Pengujian *Backpropagation*

Berikut ini algoritma proses pengujian, yaitu:

1. Mulai.
2. Gunakan bobot akhir pelatihan.
3. Masukkan data pengujian.
4. Hitung semua keluaran di unit tersembunyi zj.
5. Hitung semua keluaran di unit keluaran yk.
6. Menentukan keputusan, benar sesuai target atau tidak.
7. Selesai.

Gambaran aliran proses kerja dari algoritma pengujian di atas dapat ditunjukkan oleh Gambar pada *flowchart* berikut:



Flowchart Algoritma Pengujian

3. Target Error Karakter

Target Error untuk 20 karakter ini ditentukan dari 0 sampai 1. Fungsi yang dipakai adalah fungsi sigmoid biner yang memiliki range (0-1). Bobot target masing-masing karakter dapat dilihat pada tabel berikut :

No	Karakter	Target Error
1	A	0.05
2	Na	0.10
3	Ca	0.15
4	Ra	0.20
5	Ka	0.25
6	Da	0.30
7	Ta	0.35
8	Sa	0.40
9	Wa	0.45
10	La	0.50
11	Pa	0.55
12	Dha	0.60
13	Ja	0.65
14	Ya	0.70
15	Nya	0.75
16	Ma	0.80
17	Ga	0.85
18	Ba	0.90
19	Ta	0.95
20	Nga	1

Tabel Target Error Tiap Karakter

3. Perancangan Database

Database diperlukan untuk menyimpan pola yang didapatkan dari proses pelatihan jaringan. Database yang digunakan untuk menyimpan pola adalah Microsoft Access 2007. Ada 2 tabel yang digunakan untuk proses ini, yaitu tabel rs_pembelajaran dan rs_karakter. Di tabel rs_pembelajaran, data yang disimpan yaitu id, pola, bobot_output. Id digunakan sebagai identitas pemilik pola, id ini bersifat *foreign key*, pola_h adalah pola biner secara horizontal dari huruf tersebut, sementara bobot_output disimpan setelah proses pembelajaran. Di tabel karakter ada id dan target. Id digunakan sebagai identitas pemilik pola, id ini bersifat *primary key*, target digunakan untuk proses pembelajaran. Desain database pada aplikasi ini adalah sebagai berikut :

a. Tabel rs_pembelajaran

Tabel Desain rs_pembelajaran

Field Name	Tipe Data
id	Text
Pola	Memo
Bobot_output	Number

b. Tabel rs_karakter

Tabel Desain rs_karakter

Field Name	Tipe Data
id	Text
target	Number

HASIL DAN PEMBAHASAN

1. Kebutuhan Sistem

a. Perangkat Keras (Hardware)

Perangkat keras (Hardware) merupakan komponen-komponen elektronik yang saling berinteraksi sehingga dapat digunakan untuk memasukkan data, mengolah dan menghasilkan keluaran sesuai dengan data. Pembuatan aplikasi berbasis dekstop diperlukan perangkat keras sebagai unsur pendukung dalam melaksanakan pengolahan data.

Perangkat yang digunakan adalah :

1. Prosessor Intel(R) Core(TM) i3 CPU @ 2,40Ghz
2. Memori DDR3 2 Gb
3. Hardisk 500 Gb

b. Perangkat Lunak (*Software*)

Perangkat lunak (*Software*) adalah salah satu faktor penentu utama dari sistem komputer karena tanpa adanya perangkat lunak komputer tidak dapat dioperasikan. Perangkat lunak (*Software*) merupakan kumpulan satu atau lebih program dan data yang saling berhubungan dan ketergantungan membentuk satu paket program yang berfungsi untuk mengoptimalkan kerja suatu sistem komputer. Perangkat (*Software*) yang digunakan antara lain sebagai berikut :

1. Sistem Operasi *Microsoft Windows 7 Ultimate 32 Bit*
2. *Microsoft Visual Basic 6.0*
3. *Microsoft Access 2007*

2. Implementasi Sistem

Dalam implementasi program ini penulis membuat aplikasi pengenalan aksara carakan Madura dengan menggunakan *Microsoft Visual Basic 6.0* dan database yang digunakan adalah *Microsoft Access*. Dalam aplikasi ini terdapat beberapa tampilan antar muka, diantaranya adalah form menu utama, form pengenalan, form pembelajaran, form panduan, dan form *about*, form *exit*.

1. Form Menu Utama

Form menu utama adalah form pertama kali aplikasi dijalankan. Pada form ini terdapat beberapa menu didalamnya yaitu Menu Proses *Backpropagation*, Menu Bantuan, Menu *Exit*. Berikut gambar tampilan form menu utama:

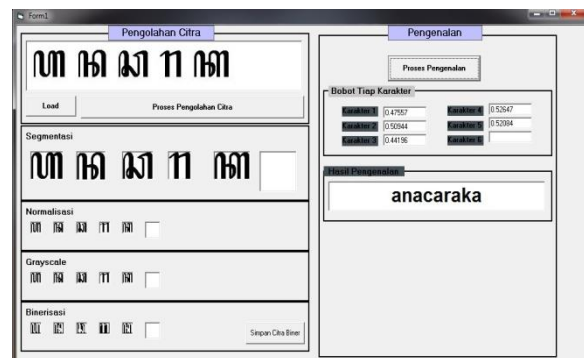


Gambar Form Menu Utama

2. Form Pengenalan Aksara Carakan Madura

Form pengenalan aksara carakan Madura adalah form untuk proses pengolahan citra dan pengenalan karakter dari data yang akan dilakukan pengujian.

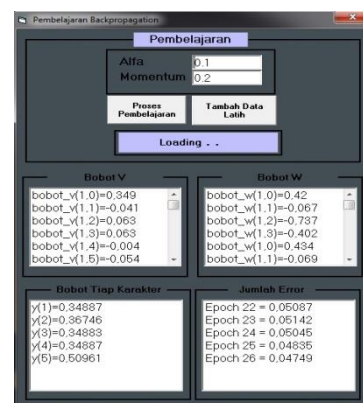
Berikut tampilan form aksara carakan Madura.



Gambar Form Pengenalan

3. Form Pembelajaran

Form pembelajaran adalah form untuk proses pembelajaran karakter atau data latih dari beberapa pola citra yang sudah ada didalam database. Berikut tampilan form pembelajaran :



Gambar Form Pembelajaran

Tombol data latih ini untuk membuka form tambah data latih. Form dapat digunakan untuk proses penambahan citra untuk pembelajaran *backpropagation*. Berikut tampilan form data latih :



Gambar Form Tambah Data Latih

3. Hasil dan Pengujian

Pengujian ini dilakukan untuk mengetahui tingkat akurasi jaringan syaraf tiruan *backpropagation* dalam mengenali karakter dari beberapa data pengujian dengan menggunakan 20 citra karakter yang sudah di training.

Pengujian (Tiap Karakter)				
No.	Huruf Aksara Madura	Hasil Pengenalan	Terdeteksi	
			Benar	Salah
1	ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ	atadhagamapa	6	0
2	ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ	athagapatada	6	0
3	ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ	dagapaabaya	4	2
4	ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ	dangawapaadha	4	2
5	ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ	dasaralawaja	4	2
6	ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ	dharalathamapa	5	1
7	ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ	gadhadalapaya	1	5
8	ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ	gamadabanyaja	6	0
9	ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ	paarabamadha	6	0
10	ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ ᮘᮞ	thadasadhalapa	5	1
Total Terdeteksi			47	13
Persentasi Keberhasilan			78,333%	21,666%

Tabel Hasil Pengenalan

Contoh perhitungan persentasi tabel diatas sebagai berikut :

$$X = (\text{Total Hasil Pengenalan} / 100\%) * \text{Hasil Terdeteksi Benar/Salah.}$$

Total Hasil Pengenalan = 60 Karakter

Jumlah Terdeteksi Benar = 47 Karakter

Jumlah Terdeteksi Salah = 13 Karakter

Dengan hasil perhitungan diatas, diperoleh kesimpulan bahwa sistem memiliki nilai akurasi dalam mengenali karakter Aksara Carakan Madura sebesar 78,333 % dan nilai kesalahan yang dimiliki sebesar 21,666 %.

PENUTUP

1. Kesimpulan

Setelah melakukan implementasi dan pengujian sistem maka diperoleh kesimpulan sebagai berikut.

- Aplikasi pengenalan aksara carakan Madura melalui proses pengolahan citra, susunannya adalah ambil citra, segmentasi, normalisasi, greyscale, dan binerisasi. Susunan tersebut dilakukan agar proses pembelajaran maupun pengenalan karakter huruf dapat dilakukan.
- Tahap-tahap pengolahan citra melalui proses dengan mengambil gambar menggunakan *webcam* atau *load* gambar pada sebuah *picture box* yang berukuran 300 x 300. Kemudian citra tersebut disegmentasikan untuk proses memotong atau memecah citra input menjadi per karakter huruf aksara carakan Madura. Agar proses *backpropagation* lebih cepat untuk proses pengnalan maka citra dinormalisasikan, sehingga citra tersebut menghasilkan ukuran 16 x 16 *pixel*.
- Berdasarkan pengujian sistem, dengan jumlah input = 256 neuron, *hidden layer* = 3 neuron, *output layer* = 1 neuron, dan *learning rate* = 0,2 dapat mengenali huruf aksara carakan Madura sebesar 78,333 %.

2. Saran

Adapun saran untuk pengembangan sistem ini adalah sebagai berikut:

- Dikembangkan penelitian lebih mendalam dan variasi algoritma pelatihan supaya didapatkan hasil yang optimal dengan waktu pelatihan yang lebih singkat.
- Komposisi pembagian data yang lain perlu dicoba untuk memperoleh hasil yang lebih optimal.
- Lapisan tersembunyi lebih dari satu dan parameter yang digunakan di set lebih tepat lagi agar jaringan dapat mengenali semua pola data input, dan dapat memprediksi dengan akurat.
- Karena banyak karakter yang memiliki ciri yang mirip, perlu ditambahkan data latih agar pengenalan bisa lebih akurat.

DAFTAR PUSTAKA

- [1] Achmad, Basuki, Palandi, dkk. Pengolahan Citra Digital Menggunakan Visual Basic. Graha Ilmu. Yogyakarta.2005
- [2] Ahmad, Balza dan Firdausy, Kartika. Teknik Pengolahan Citra Digital Menggunakan Delphi. Ardi Publishing. Yogyakarta.2005
- [3] Badrudin, Imam. Pengenalan Wajah Dengan Menggunakan PCA dan Backpropagation Untuk Sistem Absensi. STMIK ASIA. Malang.2012
- [4] Bertalya, Teori Pengolahan Citra Digital. Yogyakarta. Andi. 2002
- [5] Daryanto. Belajar Komputer Visual Basic. Rama Media. Bandung.2003
- [6] Krose, Ben dan Patrick Van Der Smagt. An Introduction to Neural Networks. The University of Amsterdam. 8th edition. 1996
- [7] Kusumadewi, Sri. Artificial Intelligence (Teknik dan Aplikasi). Graha Ilmu. Jakarta. 2003.

