

**PENGENALAN HURUF DAN ANGKA PADA CITRA BITMAP DENGAN
JARINGAN SARAF TIRUAN METODE PROPAGASI BALIK**

Naskah Publikasi



disusun oleh
Zul Chaedir
05.11.0999

Kepada
**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
AMIKOM
YOGYAKARTA
2010**

NASKAH PUBLIKASI

**Pengenalan Huruf dan Angka pada Citra Bitmap dengan
Jaringan Saraf Tiruan Metode Propagasi Balik**

disusun oleh

Zul Chaedir

05.11.0999

Dosen Pembimbing


Heri Sismoro, M.Kom

NIK. 190302057

19 April 2010

**Ketua Jurusan
Teknik Informatika**



Abas Ali Pangera, Ir, M.Kom

NIK : 190302008

RECOGNITION OF LETTER AND NUMBER AT IMAGE BITMAP BY ARTIFICIAL
NETWORK NERVE OF METHOD PROPAGASI RETURN

Zul Chaedir

Jurusan Teknik Informatika

YOGYAKARTA AMIKOM STMIK

ABSTRACT

Intention of this research is how to apply artificial nerve network to recognize of number or letter at a image bitmap. Network type to be used by artificial nerve network is perceptron of multi-layer with method of propagasi return

This useful research also in researcher same area that is artificial nerve network area. And can be used as reference and study of artificial nerve network implementation for the recognition of patterns at image bitmap

Keyword : artificial Network Nerve (SJT), Perceptron, Propagasi return, image bitmap

1. Pendahuluan

Mengenali sebuah benda atau obyek merupakan salah satu kemampuan manusia yang memainkan peranan penting dalam kehidupan. Ketika seseorang melihat suatu obyek, ia akan mengumpulkan semua informasi tentang obyek tersebut dan membandingkan dengan informasi yang ada di dalam ingatannya, dan jika ditemukan kesamaan atau mendekati maka obyek akan dikenali.

Pengenalan pola dapat dibagi menjadi 2 (dua), yaitu: mengenali obyek konkret dan mengenali obyek abstrak. Mengenali obyek konkret membutuhkan pengenalan-pengenalan dan informasi yang bersifat fisik, misalnya mengenali pola sidik jari atau mengenali karakter pada sebuah citra. Sedangkan mengenali obyek abstrak membutuhkan pengenalan-pengenalan dan informasi berupa diskusi atau pendapat, dalam kata lain hal ini merupakan mengenali obyek yang tidak secara fisik terlihat

Penelitian yang akan dilakukan adalah menggunakan pengolahan citra digital untuk membaca elemen pixel dalam citra bitmap dan hasil dari pembacaan tersebut digunakan sebagai masukan pada proses pelatihan jaringan maupun proses pengujian dari jaringan syaraf tiruan dengan metode Propagasi Balik (*Back Propagation*).

2. Landasan Teori

2.1 Jaringan Saraf Tiruan

Jaringan saraf tiruan merupakan sistem pemrosesan informasi yang mempunyai karakteristik menyerupai jaringan saraf manusia. Jaringan saraf tiruan tercipta dalam bentuk generalisasi model matematis yang bersesuaian dengan pemahaman manusia.¹

Pendekatan yang dilakukan adalah mengikuti jalan yang berbeda dari metode komputasi tradisional untuk memecahkan masalah. Sejak komputer konvensional menggunakan pendekatan berdasarkan algoritma, apabila salah satu langkah tertentu yang harus diikuti oleh komputer tidak diketahui, maka komputer tidak dapat memecahkan masalah selanjutnya. Yang artinya, metode komputasi tradisional hanya dapat memecahkan masalah-masalah yang kita sudah memahaminya dan mengetahui bagaimana harus memecahkannya.

Dalam hal ini Jaringan Saraf Tiruan mempunyai kemampuan lebih karena Jaringan Saraf Tiruan mampu memecahkan permasalahan-permasalahan yang kita belum mengetahui bagaimana cara memecahkannya. Jaringan Saraf Tiruan mempunyai kemampuan untuk menyesuaikan diri, belajar, menyamaratakan dan mengelola data. Banyak sekali jenis jaringan antara lain: *Perceptron*, *Adaline*, *Madaline*, *Kohonen*, *Backpropagation*, dan lain sebagainya. Diantara sekian banyak

¹ Arief Hermawan, Jaringan Saraf Tiruan – Teori dan Aplikasi, Hal. 3

jenis jaringan, *Backpropagation* atau Propagasi Balik merupakan salah satu jaringan yang paling umum digunakan, yang sangat sederhana dalam penerapannya dan efektif.

2.2. Metode Propagasi Balik

Jaringan saraf tiruan metode Propagasi Balik memuat satu atau lebih lapisan (*layer*) yang masing-masing lapisan terhubung ke lapisan selanjutnya. Lapisan pertama disebut lapisan masukan (*input layer*) yang menerima data-data pembelajaran atau pola yang akan dikenali oleh jaringan, dan lapisan terakhir disebut lapisan keluaran (*output layer*) yang menyimpan nilai hasil identifikasi berdasarkan data-data masukan. Lapisan yang ada di antara lapisan masukan dan lapisan keluaran adalah lapisan tersembunyi (*hidden layer*) yang digunakan untuk melakukan propagasi maju hasil keluaran lapisan sebelumnya ke lapisan selanjutnya dan melakukan propagasi balik untuk kesalahan lapisan selanjutnya ke lapisan sebelumnya.

Pada gambar berikut ini adalah arsitektur jaringan saraf tiruan dengan satu lapisan masukan atau lapisan X, satu lapisan tersembunyi atau lapisan Z, dan satu lapisan keluaran atau lapisan Y. Tiap-tiap node di lapisan X terhubung ke tiap-tiap node di lapisan Z, dan tiap-tiap node di lapisan Z terhubung ke tiap-tiap node di lapisan Y. Penghubung-penghubung dari tiap-tiap node memiliki bobot (weights) yaitu V dan W .²

² Fausett, Laurent, *Fundamentals of Neural Networks: Architectures, Algorithms, and Applications*, Prentice-Hall, Inc., New Jersey, 1994

2.2.1. Fungsi Aktivasi

Dalam metode backpropagation fungsi aktivasi yang dipakai harus memenuhi beberapa persyaratan, yaitu: kontinyu, terdiferensial dengan mudah, dan merupakan fungsi yang tidak turun. Salah satu fungsi yang memenuhi ketiga syarat tersebut sehingga sering dipakai adalah fungsi sigmoid biner yang memiliki range (0,1). Rumusnya adalah:

$$f(x) = 1 / (1 + e^{-x}), \text{ dengan turunan } f'(x) = f(x) (1 - f(x))$$

Fungsi lain yang sering dipakai adalah fungsi sigmoid bipolar yang bentuk fungsinya mirip dengan fungsi sigmoid biner, hanya saja mempunyai range (-1,1). Rumusnya adalah:

$$f(x) = (2 / (1 + e^{-x})) - 1, \text{ dengan turunan } f'(x) = ((1 + f(x)) (1 - f(x))) / 2$$

Fungsi sigmoid memiliki nilai maksimum = 1. Maka untuk pola yang targetnya lebih dari 1, pola masukan dan keluaran harus terlebih dulu ditransformasi sehingga semua polanya memiliki range yang sama seperti fungsi sigmoid yang dipakai. Alternatif lain adalah menggunakan fungsi aktivasi sigmoid hanya pada layar yang bukan layar keluaran. Pada layar keluaran fungsi aktivasi yang digunakan adalah fungsi identitas. Rumusnya adalah:

$$f(x) = x$$

2.2.2. Momentum

Pada standar Backpropagation, perubahan bobot didasarkan atas gradient yang terjadi untuk pola yang dimasukkan saat itu. Modifikasi yang dapat dilakukan adalah melakukan perubahan bobot yang didasarkan atas arah gradient pola terakhir dan pola sebelumnya (disebut momentum) yang dimasukkan. Jadi tidak hanya pola masukan terakhir saja yang diperhitungkan.

Penambahan momentum dimaksudkan untuk menghindari perubahan bobot yang mencolok akibat adanya data yang sangat berbeda satu dengan yang lain. Apabila beberapa data terakhir yang diberikan ke jaringan memiliki pola serupa (berarti arah gradien sudah benar), maka perubahan bobot dilakukan secara cepat. Namun apabila data terakhir yang dimasukkan memiliki pola yang berbeda dengan pola sebelumnya, maka perubahan bobot dilakukan secara lambat.

Dengan penambahan momentum, bobot baru pada waktu ke $t+1$ didasarkan atas bobot pada waktu t dan $t-1$. Disini harus ditambahkan 2 variabel yang baru yang mencatat besarnya momentum untuk 2 iterasi terakhir, Jika μ adalah konstanta ($0 \leq \mu \leq 1$) yang menyatakan parameter momentum maka bobot baru dihitung berdasarkan persamaan:

$$w_{kj}(t+1) = w_{kj}(t) + (\text{Alpha} * \text{Delta}Y_k * Z_j) + \mu(w_{kj}(t) - w_{kj}(t-1))$$

dan

$$v_{ji}(t+1) = v_{ji}(t) + (\text{Alpha} * \text{Delta}Z_k * X_i) + \mu(v_{ji}(t) - v_{ji}(t-1))$$

2.3. Algoritma Pelatihan

Algoritma pelatihan Jaringan Saraf Tiruan propagasi balik pada dasarnya dibagi menjadi dua langkah, yaitu: langkah maju atau *feedforward* dan propagasi balik. Pada langkah maju perhitungan bobot hanya didasarkan pada vektor masukan saja, sedangkan pada propagasi balik, bobot-bobot dimodifikasi atau diperhalus dengan memperhitungkan nilai target yang diharapkan dan nilai keluaran yang didapat dari proses langkah maju.³

Berikut ini adalah algoritma pelatihan dengan langkah maju yaitu langkah nomor 1 sampai dengan langkah nomor 8 dan propagasi balik:

1. Siapkan data-data pelatihan, arsitektur jaringan saraf tiruan *multi-layer*
2. Tentukan nilai α (*learning rate*), Epsilon
3. Inisialisasi bobot awal dengan bilangan acak kecil
4. Selama $\Delta\text{MSE} > \text{Epsilon}$ kerjakan langkah 13 sampai:
5. Setiap node masukan X_i ($i = 1 \dots n$) meneruskan nilai vektor ke lapisan tersembunyi
6. Setiap node tersembunyi Z_j ($j = 1 \dots p$) menjumlahkan semua hasil perkalian antara vektor masukan X_i dan bobotnya V_{ij} dan menyimpan hasil penjumlahan di Z_{in_j} .

³ Ibid

n

$$Z_{in_j} = V_{0j} + \sum X_i V_{ij}$$

i = 1

Kemudian hitung keluaran dengan menggunakan fungsi aktivasi:

$$Z_j = f(Z_{in_j}) = 1 / (1 + e^{-Z_{in_j}})$$

7. Setiap node tersembunyi Z_j ($j = 1 \dots p$) meneruskan nilai vektor ke lapisan keluaran.
8. Setiap node keluaran Y_k ($k = 1 \dots m$) menjumlahkan semua hasil perkalian antara vektor tersembunyi Z_j dan bobotnya W_{jk} dan menyimpan hasil penjumlahan di Y_{in_k} .

p

$$Y_{in_k} = W_{0k} + \sum Z_j W_{jk}$$

j = 1

Kemudian hitung keluaran dengan menggunakan fungsi aktivasi:

$$Y_k = f(Y_{in_k})$$

9. Untuk setiap node keluaran Y_k ($k = 1 \dots m$) bersama nilai target yang diinginkan, hitung delta atau besar faktor kesalahannya.

$$\Delta Y_k = (T_k - Y_k) Y_k (1 - Y_k)$$

10. Hitung delta perubahan bobot W_{jk} ($j = 1 \dots p, k = 1 \dots m$)

$$\Delta W_{jk} = \acute{\alpha} \Delta Y_k Z_j$$

11. Untuk setiap node tersembunyi Z_j ($j = 1 \dots p$) bersama dengan nilai vektor keluaran Y_k ($k = 1 \dots m$), hitung delta atau besar kesalahannya.

m

$$\Delta \text{net}_j = \sum_{k=1}^m \Delta Y_k W_{jk}$$

k = 1

$$\Delta Z_j = \Delta \text{net}_j Z_j (1 - Z_j)$$

12. Hitung delta perubahan bobot V_{ij} ($i = 1 \dots n, j = 1 \dots p$)

$$\Delta V_{ij} = \acute{\alpha} \Delta Z_j X_i$$

13. Perbaharui bobot-bobot dengan menjumlahkan bobot lama dengan delta perubahan bobot.

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

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

Tujuan utama penggunaan Backpropagation adalah mendapatkan keseimbangan antara pengenalan pola pelatihan secara benar dan respon yang baik untuk pola lain yang sejenis. Jaringan dapat dilatih terus menerus hingga semua pola pelatihan dikenali dengan benar. Akan tetapi hal itu tidak menjamin jaringan akan mampu mengenali pola pengujian dengan tepat. Jadi tidaklah bermanfaat untuk meneruskan iterasi hingga semua kesalahan pelatihan = 0