

Sistem Pengenalan Nomor Plat Kendaraan Berbasis Foto Digital Dengan Metode Moment Invariant dan Jaringan Syaraf Tiruan Menggunakan Algoritma Backpropagation

Zaiful Bahri, Sukamto dan Joko Risanto

Jurusan Matematika FMIPA Universitas Riau

E-mail: zaiful_br@yahoo.co.id

Abstrak. Dalam tulisan ini dibahas mengenai pengenalan nomor plat kendaraan berbasis foto digital dengan metode moment invariant dan jaringan syaraf tiruan menggunakan metode algoritma backpropagation. Metode moment invariant dan jaringan syaraf tiruan digunakan untuk ekstraksi citra karakter yang kemudian diimplementasikan dengan Matlab untuk mengenal nomor plat kendaraan menggunakan algoritma backpropagation dengan multi layer.

Kata Kunci : Moment invariant, nomor plat kendaraan, citra karakter, jaringan syaraf tiruan, algoritma backpropagation, multi layer.

PENDAHULUAN

Sistem pengenalan pola plat nomor kendaraan merupakan suatu aplikasi proses *image* yang kompleks dalam hal mana pengenalan karakter-karakter pada sebuah plat mobil berdasarkan pada syarat dan instruksi-instruksi yang diberikan.

Tulisan ini akan fokus pada penelitian dengan menerapkan jaringan neural propagasi balik untuk mengenal pola nomor plat mobil sebagai suatu alternatif solusi untuk mengenal pola karakter dengan ekstraksi ciri menggunakan Moment Invariant dan Jaringan Syaraf Tiruan. Pola karakter yang diekstrak tersebut disimpan di dalam database. Pola karakter yang tersimpan ini, kadang kala tidak sesuai dengan pola karakter yang akan dikenal. Salah satu faktor yang menyebabkan kegagalan dalam mendapatkan akurasi atau ketepatan 100% dalam mengenal pola adalah tak mampunya sistem dengan berbagai metode yang diterapkan untuk mengenal pola karakter-karakter, sedemikian hingga dalam kasus pengenalan 'B' dan '8', 'O' dan '0', 'G' dan '6'[10],

karakter 'B' atau '3' sebagai '8', karakter '5' sebagai '6', karakter '6' sebagai 'G', karakter 'A' sebagai '4' dan karakter 'I' sebagai '1'[7], karakter '3' dengan karakter '8', '4' dengan 'A', '8' dengan 'B' dan 'D' dengan '0'[8] dan '8' dengan 'B' atau '0' dengan 'D'[4].

METODE PENELITIAN

Citra digital adalah suatu fungsi identitas cahaya dua dimensi $f(x,y)$, dimana x dan y menunjukkan koordinat spasial. Nilai $f(x,y)$ pada tiap titik menunjukkan tingkat keabuan (*gray level*) citra pada titik tersebut [6].

Citra digital dapat berupa citra dalam mode keabuan atau citra berwarna (*color*). Setiap citra direpresentasikan dalam bentuk matrik berukuran $m \times n$, dimana m menunjukkan banyaknya elemen baris dan n untuk jumlah kolom pada matriks tersebut.

$$f(x,y) = \begin{pmatrix} f(x_1,y_1) & \dots & f(x_1,y_n) \\ \vdots & \ddots & \vdots \\ f(x_m,y_1) & \dots & f(x_m,y_n) \end{pmatrix} \quad (1)$$

Gambar 1: Representasi citra digital berukuran $m \times n$.



Tiap sel matrik disebut *picture element* disingkat dengan *pixel* yang mewakili tingkat keabuan atau intensitas warna. Pada citra digital dengan format 8 bit akan memiliki 256 (2^8) intensitas warna. Nilai ini berkisar antara 0 sampai dengan 255 dengan nilai 0 menunjukkan intensitas paling gelap dan nilai 255 menunjukkan intensitas paling terang.

Ekstraksi Ciri

Ekstraksi ciri merupakan suatu proses pelatihan. Di mana sebuah sistem bisa memuat sebuah citra, preproses citra, mengekstrak ciri citra, menghitung 'jarak' antara ciri citra yang diekstrak dan mengetahui vektor yang tersimpan dalam pustaka model citra, dan mengenali citra berdasarkan derajat kesamaan antara citra yang dimuat dan model citra.

Tahap preproses berguna untuk membuat image sesuai dengan algoritma ekstraksi ciri yang berbeda. Beberapa algoritma ekstraksi ciri hanya sesuai dengan *countur* citra sedangkan beberapa algoritma menghitung setiap pixel citra.

Tahap preproses termasuk di dalamnya adalah *thresholding*, binarisasi, *filtering*, dekteksi tepi, *gap filling* dan segmentasi.

Moment Invariant

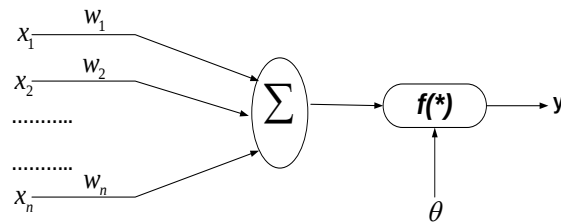
Moment invariati didefinisikan sebagai [3] dan [9] :

$$\begin{aligned}\phi_1 &= \eta_{20} + \eta_{02} \\ \phi_2 &= (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \\ \phi_3 &= (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \\ \phi_4 &= (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \\ \phi_5 &= (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\ &+ (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\ \phi_6 &= (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \\ \phi_7 &= (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\ &+ (3\eta_{21} - \eta_{30})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]\end{aligned}$$

Moment invariant adalah invariant terhadap transformasi citra termasuk di dalamnya skala, translasi dan rotasi.

Jaringan Syaraf Tiruan

Jaringan Syaraf Tiruan(JST) didefinisikan sebagai sistem komputasi yang didasarkan pada pemodelan syaraf biologi (*neuron*) melalui pendekatan dari sifat-sifat komputasi biologis (*biological computation*). JST bisa dibayangkan berupa jaringan dengan elemen pemroses sederhana yang saling terhubung. Elemen pemroses berinteraksi melalui sambungan variabel yang disebut bobot, dan bila diatur secara tepat dapat menghasilkan sifat yang diinginkan [2]. Model *neuron* sederhana disajikan pada Gambar 1.



Gambar 1: Sistem Komputasi Model Neuron.

Dan pernyataan matematisnya

$$y = f\left(\sum_{i=1}^n w_i x_i - \theta\right)$$

(3)

dengan x_i = sinyal masukan, $i = 1, 2, \dots, n$ (n = banyaknya simpul masukan)

w_i = bobot hubungan atau *synapsis* = *threshold* atau bias

$f(*)$ = fungsi aktivasi

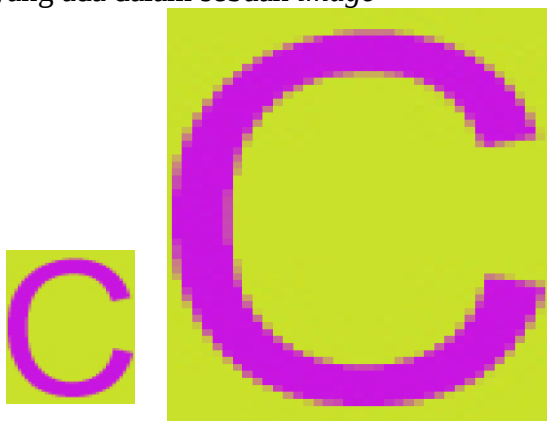
y = sinyal keluaran dari *neuron*

Proses ekstraksi dilakukan untuk mengubah ukuran *image* huruf maupun angka yang ada pada plat nomor kendaraan tersebut di mana setelah dilakukan perubahan ukuran huruf, *image* yang mempunyai nilai "1" akan dijumlahkan dan



dirata-ratakan. Selanjutnya nilai *image* ini akan dimasukkan ke dalam sebuah matriks yang berukuran $m \times n$, kemudian siap untuk diproses dengan algoritma backpropagation sebagai pengenalan pola. Ukuran huruf setelah diubah dengan fungsi *imresize* adalah 90×60 piksel. Kemudian huruf ini akan ditempatkan pada area piksel dengan ukuran 10×10 yang selanjutnya dimasukkan ke dalam sebuah matriks berukuran 9×6 sebagai input yang diperlukan untuk pelatihan ini.

Setelah proses ekstraksi ciri dilakukan dengan Moment Invariant dan JST, plat nomor mobil akan disegmentasi ke dalam karakter-karakter terpisah dengan menandai setiap karakter menggunakan sebuah fungsi label dalam Matlab. Pelabelan ini mampu untuk mengenal berapa banyak karakter yang ada dalam sebuah *image*



Gambar 3a Sebelum diekstrak 3b. Sesudah diekstrak

Untuk masing-masing karakter yang sudah diberi label, baris maksimum dan minimum yang berhubungan dengan kolom digunakan untuk menentukan piksel karakter awal. Nilai-nilai ini ditandai sebagai vektor [“kolom baris” “lebar tinggi”]. Bentuk ini akan merubah ukuran huruf menjadi ukuran yang seragam dengan fungsi *imresize* 90×60 piksel agar pengenalan pola lebih efektif. Perubahan

ukuran ini dapat dilihat pada gambar 3a dan 3b.

Algoritma Propagation

Metode propagasi balik merupakan metode jaringan syaraf tiruan yang sangat baik dalam menangani masalah pengenalan pola-pola kompleks. Metode ini merupakan metode jaringan syaraf tiruan yang populer. Beberapa contoh aplikasi yang melibatkan metode ini adalah pengompresan data, pendekteksian virus komputer, pengidentifikasian objek, sintesis suara dari teks, dan lain-lain.

Cara kerja jaringan propagasi balik adalah sebagai berikut: Mula-mula jaringan diinisialisasi dengan bobot yang diset dengan bilangan acak. Kemudian contoh pelatihan dimasukkan ke dalam jaringan. Keluaran jaringan berupa sebuah vektor output target. Selanjutnya vektor output aktual jaringan dibandingkan dengan vektor output target untuk mengetahui apakah output jaringan sudah sesuai dengan harapan atau output aktual. Berikut langkah-langkah algoritma propagasi balik.

1. Inisialisasi bobot-bobot

Tentukan nilai pembelajaran (α).

Tentukan pula nilai toleransi *error* atau nilai ambang (bila menggunakan nilai ambang sebagai kondisi berhenti); atau set maksimum *epoch* (bila menggunakan banyak *epoch* sebagai kondisi berhenti)

2. **While** kondisi berhenti tidak terpenuhi **do** langkah ke-3 sampai langkah ke-10

3. Untuk setiap pasangan pola pelatihan, lakukan langkah ke-4 sampai dengan langkah ke-9.

Tahap (feedforward) umpan maju

1. Setiap unit input x_i ($i = 1, 2, \dots, n$; dari unit ke-1 sampai unit ke- n pada lapisan input) mengirimkan sinyal input ke

semua unit yang ada di lapisan atasnya(ke lapisan tersembunyi)

2. Pada setiap unit di lapisan tersembunyi z_j , (dari unit ke-1 sampai unit ke-p; $i = 1, 2, \dots, n; j = 1, 2, \dots, p$) sinyal output lapisan tersembunyinya dihitung dengan menerapkan fungsi aktivasi terhadap penjumlahan sinyal input berbobot x_i :

$$z_j = f(v_{0j} + \sum_{i=1}^p x_i v_{ij})$$

kemudian dikirim ke semua unit di lapisan atasnya.

3. Setiap unit di lapisan output y_k (dari unit ke-1 sampai unit ke-m; $i = 1, 2, \dots, n; k = 1, 2, \dots, m$) dihitung sinyal outputnya dengan menerapkan fungsi aktivasi terhadap penjumlahan sinyal-sinyal input berbobot z_j bagi lapisan ini :

$$y_k = f(w_{0k} + \sum_{j=1}^m z_j w_{jk})$$

Tahap Propagasi Balik

1. Setiap unit output y_k (dari unit ke-1 sampai unit ke-m; $j = 1, 2, \dots, p; k = 1, 2, \dots, m$) menerima pola target t_k , lalu informasi kesalahan lapisan output (δ_k) dihitung. (δ_k) dikirim ke lapisan di bawahnya dan digunakan untuk menghitung besar koreksi bobot dan bias (Δw_{jk} dan Δw_{0k}) antara lapisan tersembunyi dengan lapisan output :

$$\delta_k = (t_k - y_k) f'(w_{0k} + \sum_{j=1}^m z_j w_{jk})$$

di mana :

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

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

2. Pada setiap unit di lapisan tersembunyi (dari unit ke-1 sampai unit ke-p;

$i = 1, 2, \dots, n; j = 1, 2, \dots, p; k = 1, 2, \dots, m$) dilakukan perhitungan informasi kesalahan lapisan tersembunyi (δ_j). δ_j kemudian digunakan untuk menghitung besar koreksi bobot dan bias (Δv_{ij} dan Δv_{0j}) antara lapisan input dan lapisan tersembunyi.

$$\delta_j = (\sum_{k=1}^m \delta_k w_{jk}) f'(v_{0j} + \sum_{i=1}^n x_i v_{ij})$$

di mana :

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

$$\Delta v_{0k} = \alpha \delta_j$$

Tahap Peng-Update-an Bobot dan Bias

1. Pada setiap unit output y_k (dari unit ke-1 sampai ke-m) dilakukan peng-update-an bobot dan bias

($j = 0, 1, 2, \dots, p; k = 1, 2, \dots, m$) sehingga bias dan bobot yang baru menjadi :

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

2. Dari unit ke-1 sampai unit ke-p di lapisan tersembunyi juga dilakukan peng-update-an nilai bobot dan biasnya ($i = 0, 1, 2, \dots, n; j = 1, 2, \dots, p$) :

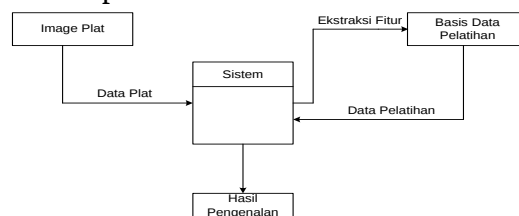
$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij}$$

3. Test Kondisi Berhenti.

HASIL DAN PEMBAHASAN

Diagram Konteks

Konteks diagram adalah diagram yang menggambarkan data yang masuk dan keluar dari sistem. Berikut adalah diagram konteks untuk sistem pengenalan pola nomor plat kendaraan :

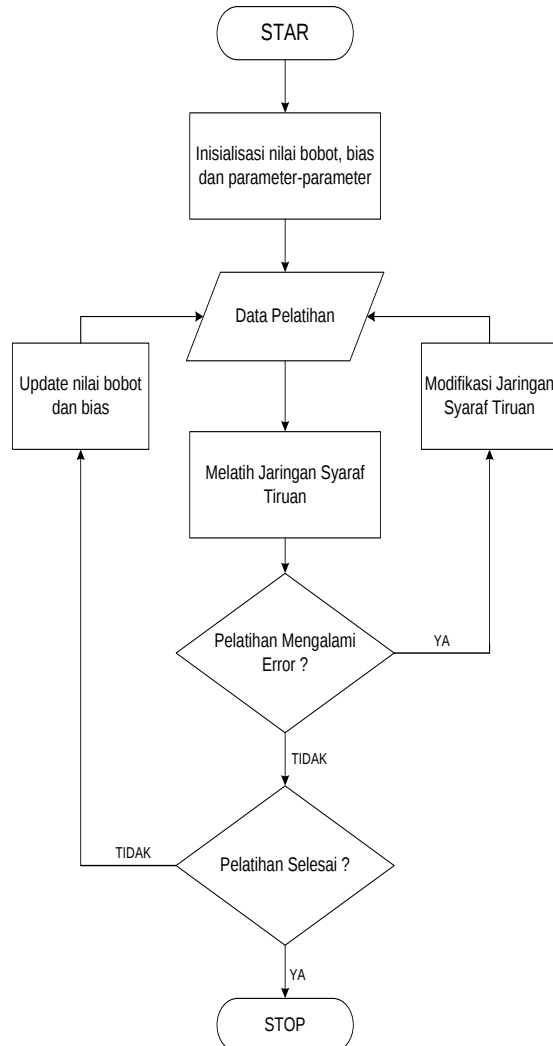


Gambar 4 Diagram Konteks



Semirata 2013 FMIPA Unila | 575

Zaiful Bahri dkk: Sistem Pengenalan Nomor Plat Kendaraan Berbasis Foto Digital Dengan Metode Moment Invariant dan Jaringan Syaraf Tiruan Menggunakan Algoritma Backpropagation



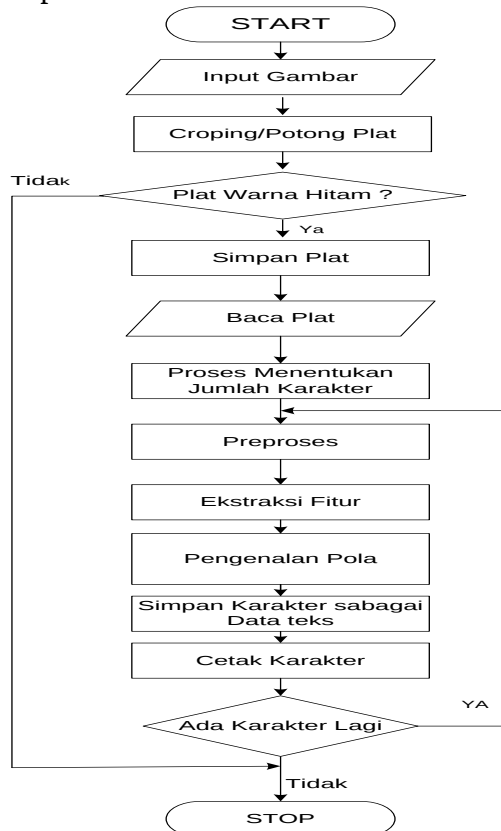
Gambar 6 Diagram Alir Pelatihan **Arsitektur Backpropagation**

Pembangunan simulator jaringan syaraf tiruan Backpropagation [1] merupakan gambaran arsitektur jaringan syaraf tiruan yang akan digunakan dalam tulisan ini. Nilai bias terdiri dari b_1, b_2 dan b_3 . Nilai input terdiri dari $X_1, X_2, \dots, X_i (i = 1, 2, \dots, 69)$. Nilai bobot dari lapisan input ke lapisan tersembunyi 1 adalah $u_{i,j}$, di mana $(i = 1, 2, \dots, 69)$ dan $(j = 1, 2, \dots, 324)$. Nilai input dari lapisan tersembunyi 1 ke lapisan tersembunyi 2 adalah A_j , di mana $(j = 1, 2, \dots, 324)$. Nilai

bobot dari lapisan tersembunyi 1 ke lapisan tersembunyi 2 adalah v_{kj} , di mana $(k = 1, 2, \dots, 324)$ dan $(j = 1, 2, \dots, 324)$. Nilai input dari lapisan tersembunyi 2 ke lapisan output adalah B_k di mana $(k = 1, 2, \dots, 324)$. Nilai bobot dari lapisan tersembunyi 2 ke lapisan output adalah w_{lk} di mana $(l = 1, 2, \dots, 36)$ dan $(k = 1, 2, \dots, 324)$.

Pembuatan Sistem

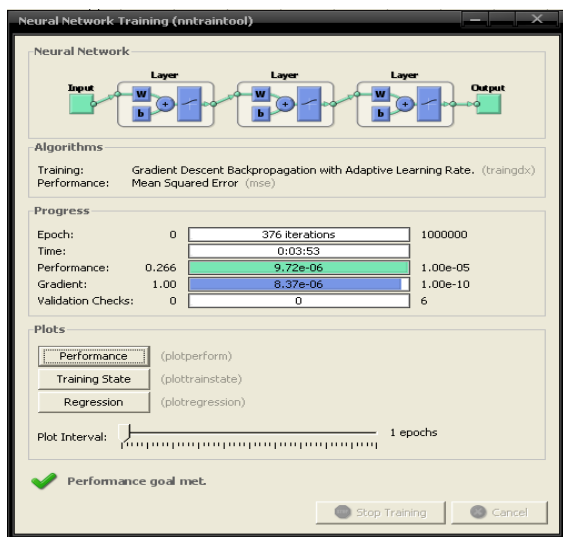
Dalam pembuatan sistem menggunakan Matlab, pengenalan pola nomor plat kendaraan dengan algoritma jaringan propagasi balik ini diperlukan beberapa tahap.



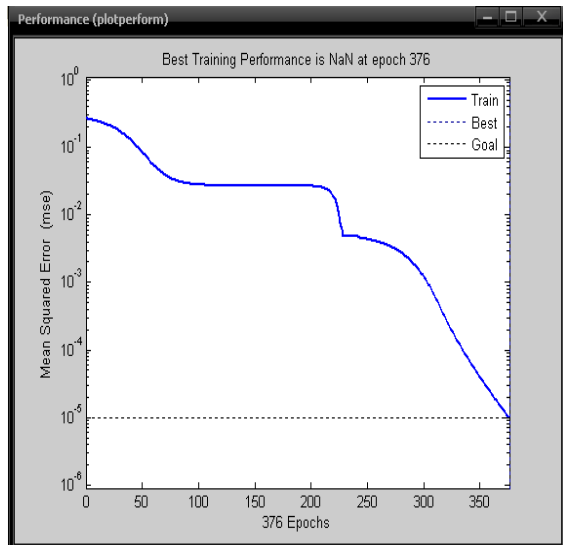
Gambar 7. Diagram Alir Aplikasi

Sebelum sistem pengenalan pola digunakan, maka terlebih dahulu dilakukan pelatihan terhadap beberapa data pelatihan. Data pelatihan yang digunakan adalah data pelatihan Gambar 5. Dengan hasil sebagai berikut :



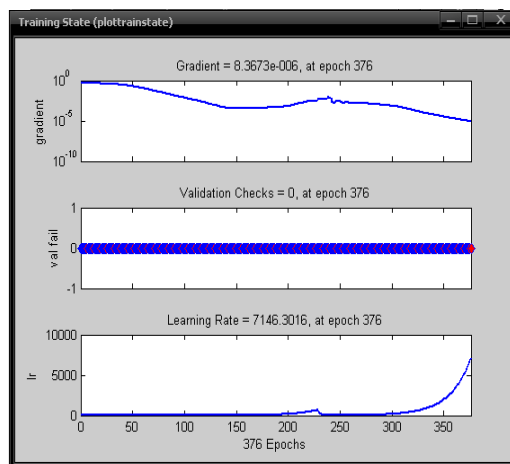


Gambar 8. Hasil Proses Data Pelatihan



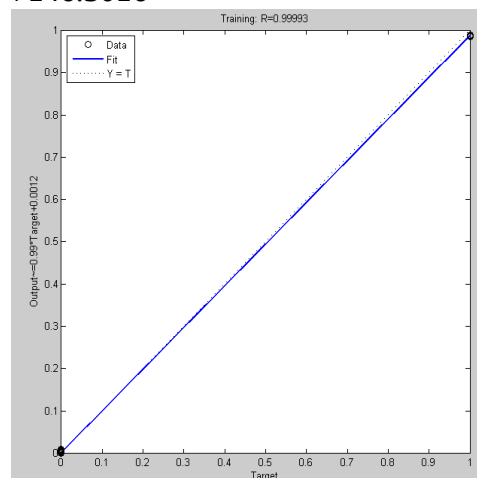
Gambar 9. Performa Jaringan Data Pelatihan

Dari proses data pelatihan 1 terlihat bahwa algoritma pelatihan yang digunakan adalah *traingdx*, performa Error Kuadrat Rata-Rata/*Mean Square Error*(MSE) $9.72e-06$, *epoch* 376 iterasi, waktu pelatihan 3 menit 53 detik dan *gradient* $8.37e-06$. Dari gambar 9 terlihat bahwa jaringan mencapai konvergensi pada iterasi ke 376 dengan *mse* $9.27e-06$



Gambar 10. Tingkat Pembelajaran Data Pelatihan

Pada gambar 10 terlihat bahwa *gradient* $8.37e-06$ pada *epoch* ke 376 cek validasi 0 dan tingkat pembelajaran sebesar 7146.3016



Gambar 11. Plot Regresi Data Pelatihan

Pada gambar 11 terlihat bahwa plot regresi antara garis lurus berwarna biru dengan garis titik-titik berwarna biru hampir sama, berarti *error*-nya sangat kecil. Untuk data pelatihan ini nilai regresinya adalah 0.9999 yang mendekati 1. Data pelatihan yang baik akan menunjukkan angka mendekati 1. Berikut adalah hasil perbandingan beberapa pelatihan :

Zaiful Bahri dkk: Sistem Pengenalan Nomor Plat Kendaraan Berbasis Foto Digital Dengan Metode Moment Invariant dan Jaringan Syaraf Tiruan Menggunakan Algoritma Backpropagation

Tabel 1. Hasil dan Parameter Data Pelatihan

	Konfig1	Konfig2	Konfig3	Konfig4	Konfig5	Konfig6
Algoritma Pelatihan	traingdx					
Jumlah lapisan input	69					
lapisan tersembunyi 1	54	108	162	216	270	324
lapisan tersembunyi 2	54	108	162	216	270	324
Jumlah lapisan output	36					
Momentum	0.9					
Fungsi pelatihan	MSE					
Fungsi tujuan	0.00001					
Jumlah <i>epochs</i>	1000000					
Fungsi transfer	logsig, logsig, logsig					
Jumlah iterasi	521	495	409	389	386	376
Waktu iterasi	22 det	39 det	49det	1.04 det	1.25det	3.53det
<i>Gradient</i>	3.54e-06	4.79e-06	5.56e-06	6.19e-06	6.81e-06	8.37e-06
Tingkat pembelajaran	39952	22586	16137	12833	10919	7146
Nilai regresi	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999

Dari tabel 1 terlihat bahwa semakin banyak jumlah lapisan tersembunyinya maka jumlah iterasi semakin sedikit untuk mencapai konvergensi. Tetapi tingkat waktu yang dibutuhkan untuk iterasi tersebut semakin meningkat.

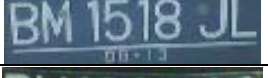
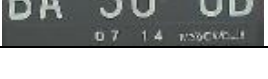
Namun demikian bisa dilihat bahwa tingkat pembelajarannya semakin kecil dan nilai regresi untuk masing-masing jumlah lapisan tersembunyi adalah tetap. Ini disebabkan data pelatihan yang digunakan adalah sama.



Gambar 12. Sistem Pengenalan Nomor Plat



Tabel 2. Hasil Beberapa Pengenalan Nomor Plat Kendaraan

No.	Plat Asli	Hasil Cropping	Pengenalan	%	Ket
1			BA 2223 SW	100	Baik
2			BM 1733 TE	100	Baik
3			BM 1909 TE	100	Baik
4			B 1183 SFF	100	Baik
5			BA 2057 TG	100	Baik
6			BM 1168 LT	100	Baik
7			BA 1440 J	100	Baik
8			BA 335 TY	100	Baik
9			BM 1518 JL	100	Baik
10			BH 1830 SG	87,5	Cukup Baik, Noise Pixel
11			BM 1026 AF	100	Baik
12			BM 1598 QF	100	Baik
14			BA 2003 W	100	Baik
15			BA 22 18 JZ	100	Baik
16			BM 1152 KL	100	Baik
18			BA 3746 JD	100	Baik
17			BM 1497 BC	100	Baik
18			BA 50 UB	100	Baik

Zaiful Bahri dkk: Sistem Pengenalan Nomor Plat Kendaraan Berbasis Foto Digital Dengan Metode Moment Invariant dan Jaringan Syaraf Tiruan Menggunakan Algoritma Backpropagation

19			BM 1143 QJ	100	Baik
20			BM 1371 QE	100	Baik

Generalisasi adalah tingkat pengenalan jaringan dalam mengenali sejumlah pola yang diberikan. Secara matematis generalisasi dapat ditulis sebagai berikut [7]:

$$\text{Generalisasi} = \frac{\text{Jumlah pola yang dikenali}}{\text{Jumlah seluruh pola}} \times 100\%$$

Hasil rata-rata pengenalan pola dari 20 data uji nomor plat kendaraan dengan jaringan backpropagation ini adalah 95 %.

KESIMPULAN

Dari hasil pengenalan pola nomor plat kendaraan di atas dapat disimpulkan bahwa Jaringan Syaraf Tiruan(JST) dengan Algoritma Backpropagation dapat mengenal pola dengan baik. Hal ini terlihat bahwa hasil pengenalan sudah mencapai 95%. 5% error lebih disebabkan adanya nomor plat yang terputus akibat adanya baut pemasangan nomor plat sehingga saat dieskrak hurufnya putus. Semakin banyak jumlah node tersembunyinya maka semakin lama proses iterasinya. Tetapi Jaringan Semakin cepat Belajar. Sehingga pengenalan akan lebih akurat. Dalam penelitian ini plat nomor yang digunakan adalah plat nomor normal. Artinya plat nomor tidak dalam keadaan terotasi atau skew. Untuk itu penelitian selanjutnya diharapkan untuk dapat mengenal pola nomor plat yang dalam keadaan terotasi atau skew. Dalam penelitian ini juga menggunakan *cropping* manual, sehingga diharapkan kepada peneliti untuk dapat meng-*cropping* nomor

plat secara otomatis dengan menentukan posisi plat terlebih dahulu.

UCAPAN TERIMA KASIH

Penulis ingin mengucapkan terima kasih kepada Lembaga Penelitian Universitas Riau yang telah mendanai penelitian laboratorium ini. Penulis juga mengucapkan terima kasih kepada Kepala Laboratorium Komputasi Jurusan Matematika FMIPA Universitas Riau yang telah memberikan kesempatan untuk melakukan penelitian. Tak lupa penulis juga mengucapkan banyak terima kasih kepada rekan kerja Sukanto dan Joko Risanto yang banyak memberikan masukan dalam penelitian ini.

DAFTAR PUSTAKA

- Anam Sarawat, Islam Md. Shohidul, Kashem M.A, Islam M.N, Islam M.R, Islam M.S. *Face Recognition Using Genetic Algorithm and Back Propagation Neural Network*. Proceedings of the International MultiConference of Engineers and Computer Scientists 2009 Vol I IMECS 2009, March 18 - 20, 2009, Hong Kong
- Fausett L. 1994. *Fundamentals of Neural Networks (Architectures, Algorithms, and Applications)*. Prentice-Hall, New Jersey.
- Flusser, J, Suk, T and Zitova, B, 2009. *Moments and Moment Invariants in Pattern Recognition*. A John Wiley and Sons, Ltd, Publication, United Kingdom.



- McAndrew, A., (2004). "Introduction to Digital Image Processing With Matlab", Thomson Course Technology:USA, pp353
- Mohamed, Nawwaf, Cheng and Ching, (2007). "Character Recognition Systems., A Guide for Student and Practitioners", John Wiley and Sons. Inc, Publication
- R. Gonzales and R. E. Wood. 2002. *Digital Citra Processing*, Second Edition. Prentice Hall, Inc., New Jersey
- Seu et al (1999). "Practical Algorithms for Image Analysis: Description, Examples and Code, edisi kebrp, Cambridge University:USA.
- Shen-Zheng Wang dan His-Jian Lee (2007), "A Cascade Framework for Real Time Statistical Plate Recognition System", IEEE Transaction On Information Forensics and Security, 2(2).
- S. Pakchalakis and O. Lee,1999. *Pattern recognition in gray scale images using moment based invariants feature*, Image Processing and its Applcation, IEEE Conference Publication No.pp.245-249
- T, Suk and J. Flusser. 2002. *Blur and Affine Moment Invariants*, ICPR2002, 16th International Conference on Pattern Recognition, Vol.4, pp.339-342.