

Pengenalan Karakter Plat Kendaraan Bermotor berbasis Citra dengan menggunakan Metode *Canny* dan Algoritma *Backpropagation*

Andy Haryoko^{1,2,a*}, Sholeh Hadi Pramono^{1,b}

¹Program Studi Magister Teknik, Universitas Brawijaya, Malang, Indonesia

²Jurusan Teknik Informatika, Universitas PGRI Ronggolawe, Tuban, Indonesia

^aandyharyoko@gmail.com, ^bsholehpramono@gmail.com

Abstract. *Transportation is important needs in daily life. However, there are so many problems in transportation system in our country. The one is public transportation. To overcome it, the government implement KIR. But this KIR has several weakness. One of them is manual data. As cosequences, human error in listing can came out as the process goes on. Pattern recognition can be used to implement automatic number plate identification in this system. One of the method is canny filter. Canny filter is uses to obtain a good image in the character image acquisition. Characters based with 12X7 pixels are be converted into binary as input for Multi Layer Perceptron with 3 layers node number of each node 84, 50, 36. Artificial neural network is trained with back propagation algorithm with a learning rate parameter 0.3 and momentum 0.9. The training process will be terminated when the iteration reaches a maximum value of 10,000 or MSE (Mean Square Error) 0.0001. Recognition rate for numeral character is 100%, however recognition rate for letter character is little bit worser, 86,87%. So overall performance is 94,29% for the whole characters.*

Keywords. *Canny Filter, Digital Image Processing, Licences Plate Identifcation, Multi Layer Perceptron, Neural Network*

Abstrak. Transportasi merupakan kebutuhan yang sangat penting dalam kehidupan sehari-hari. Namun, kenyataannya banyak permasalahan yang ditimbulkan dari transportasi itu sendiri. Salah satu permasalahan yang timbul adalah pada angkutan umum. Salah satu upaya yang dilakukan pemerintah untuk mengatasi permasalahan tersebut adalah dengan pengadaan Uji Kir. Hanya saja Uji Kir itu sendiri memiliki beberapa kelemahan. Salah satu kelemahannya adalah sistem pendataan yang masih manual. Sehingga, Kesalahan manusia dalam pencatatan kendaraan dan waktu yang dibutuhkan dalam proses tersebut merupakan kendala yang bisa saja terjadi selama proses tersebut. Salah satu metode yang dapat diterapkan untuk mengatasi masalah di atas adalah metode pengenalan Tanda Nomor Kendaraan Bermotor (TNKB) yang melekat pada kendaraan bermotor tersebut. Pengenalan pola dapat digunakan sebagai identifikasi pengenalan plat nomor kendaraan pada sistem yang diajukan pada penelitian ini. Filter canny digunakan untuk mendapatkan citra yang baik dalam pengambilan citra karakter. Karakter disampling dengan ukuran 12X7 untuk dikonversi kedalam bentuk biner sebagai input neuron dalam jaringan syaraf tiruan arsitektur *Multi Layer Perceptron* yang digunakan menggunakan 3 Layer dengan jumlah masing masing node 84, 50, 36. Jaringan syaraf tiruan dilatih dengan algoritma *backpropagation* dengan parameter *learning rate* 0.3 dan *momentum* 0.9. Proses training akan dihentikan apabila iterasi mencapai nilai maksimal 10.000 atau MSE (*Mean Square Error*) 0.0001. Dari hasil pengujian yan dilakukan sistem mampu mengenali karakter angka 100% sedangkan huruf 86,87% jadi kehandalan sistem mengenali karakter secara keseluruhan adalah 94,29%.

Kata kunci. *Fiter Canny, Jaringan Syaraf Tiruan, Multi layer perceptron (MLP), Pengenalan karakter Tanda Nomor Kendaraan Bermotor (TNKB), Pengolahan Citra Digital*

Latar Belakang

Transportasi merupakan kebutuhan yang sangat penting dalam kehidupan sehari-hari. Hal ini dikarenakan hampir semua kegiatan manusia tidak lepas dari proses transportasi (Manuel D.

Rossetti et al, 2012). Transportasi adalah suatu proses pemindahan manusia atau barang dari suatu tempat ke tempat lain dengan menggunakan suatu alat bantu kendaraan darat, kendaraan laut, maupun kendaraan udara, baik umum maupun pribadi dengan menggunakan mesin atau tidak menggunakan mesin (Ying Wen et al, 2012). Dalam rangka optimasi manfaat transportasi bagi kepentingan manusia, banyak pihak yang terlibat dalam operasi transportasi. Bukan pihak pemerintah saja yang terlibat, akan tetapi pihak swasta juga. Transportasi memegang peranan penting dalam dinamika masyarakat bahkan dinamika negara dan bangsa, baik dalam kehidupan sehari-hari, kehidupan budaya, kehidupan politik, terutama dalam kehidupan sosial ekonomi (David J. Roberts et al, 2012).

Perkembangan suatu masyarakat atau daerah tergantung pada perkembangan transportasi atau sebaliknya, perkembangan transportasi suatu negara dan masyarakat tergantung pada perkembangan aktivitas atau kegiatan perdagangan dan bisnis dari suatu negara atau masyarakat tersebut.

Berdasarkan pentingnya transportasi untuk kehidupan masyarakat serta peranan berbagai pihak bukan berarti masalah tersebut sudah dapat diatasi dengan baik. Kenyataannya banyak permasalahan yang ditimbulkan dari transportasi itu sendiri. Salah satu permasalahan yang timbul adalah adanya angkutan umum yang memprihatinkan padahal angkutan umum memegang peranan yang cukup penting dalam kehidupan masyarakat. Permasalahan angkutan umum yang memprihatinkan ini misalnya kondisi mesin yang sudah tua sehingga sering mogok, badan dan kursi angkutan umum yang sudah reyot, atau atap kendaraan umum yang bocor.

Salah satu upaya yang dilakukan pemerintah untuk mengatasi permasalahan tersebut adalah dengan pengadaan kir atau keur dalam bahasa Belanda yang dilaksanakan oleh Balai Pengujian Kendaraan Bermotor atau uji KIR. Uji KIR adalah serangkaian kegiatan menguji dan/atau memeriksa bagian-bagian, kendaraan bermotor, kereta gandengan, kereta tempelan, dan kendaraan khusus dalam rangka pemenuhan terhadap persyaratan teknis dan laik jalan.

Hanya saja uji itupun memiliki beberapa kelemahan. Salah satu kelemahan dari uji KIR kelayakan kendaraan bermotor di Indonesia yang harus dihadapi oleh pemerintah saat ini adalah dengan masih diterapkannya system manual dalam proses pendataan. Kesalahan manusia dalam pencatatan kendaraan dan waktu yang dibutuhkan dalam proses tersebut merupakan kendala yang bisa saja terjadi selama proses tersebut.

Salah satu metode yang dapat diterapkan untuk mengatasi masalah di atas adalah metode pengenalan Tanda Nomor Kendaraan Bermotor (TNKB) yang melekat pada kendaraan bermotor tersebut. Metode yang dapat dipergunakan untuk mengenali kendaraan bermotor antara lain *object detection*, *image processing*, atau *pattern recognition*. Metode pengenalan plat ini dikenal dengan istilah Automatic Plate License Plate Recognition (ALPR) atau dikenal juga dengan sebutan *automatic vehicle identification*, *car plate recognition*, *automatic number plate recognition*, dan *optical character recognition (OCR)*. (Shan Duet al, 2013)

Walaupun metode pengidentifikasian plat itu sendiri memiliki beberapa kendala yang harus mendapat perhatian namun dengan adanya sistem data dan administrasi nomor kendaraan yang baik maka secara tidak langsung akan berpengaruh pada kenyamanan dan keamanan berkendara, mengurangi tingkat kecelakaan, dan menekan angka kejahatan atau kriminalitas pada kendaraan bermotor.

Tinjauan Pustaka

Deteksi Tepi Dengan *Canny*

Canny Edge Detector menggunakan fungsi Gaussian dalam operasinya (Thomas B Moeslund, 2009). Tahapan yang dilakukan dalam Canny Edge Detector adalah sebagai berikut:

1. Membuat mask Gaussian.
2. Konvolusi gambar dengan mask Gaussian.
3. Hitung gradien setiap pixel pada gambar.
4. Aplikasikan Non-Maximum Suppression (NMS).

Fungsi Gaussian yang digunakan pada Canny Edge Detector mengalami penurunan lebih lanjut. Pada penggunaannya, mask Gaussian berfungsi sebagai operator untuk menghaluskan gambar.

$$h(x, y) = e^{-\frac{(x^2+y^2)}{2\sigma^2}} \quad (1)$$

Setelah konvolusi antara mask Gaussian dengan gambar, maka akan diperoleh gambar yang lebih halus. Gambar yang masing-masing pixel nya memiliki nilai intensitas kemudian dicari gradiennya. Gradien ini merupakan perubahan intensitas yang terjadi terhadap sumbu x maupun sumbu y. Magnitude dari gradien tersebut dapat menunjukkan letak dari tepi pada gambar.

Seringkali tepi yang dihasilkan memiliki ketebalan lebih dari satu pixel dengan magnitudo yang bervariasi. Dengan menggunakan NMS, pixel dengan magnitudo tidak paling besar disepanjang arah gradiennya akan diubah menjadi 0, menyisakan satu pixel saja di sepanjang arah gradien tersebut. Hasilnya, tepian yang memiliki ketebalan lebih dari satu pixel akan menjadi satu pixel tebalnya.

Algoritma Backpropagation

Langkah algoritma Backpropagation dapat dilihat pada pseudocode berikut ini (Kusuma Dewi S, 2003)

```

Input      :      ProblemSize,      InputPattren,
Learningrate, IterationMax
Output : Network
Network <- ConstructNetworkLayer
NetworkWeight <- InitializeWeight (Network,
ProblemSize)
For i=1 to IterationMax
  Pattren[i] <- SelectInputPattren(InputPattren)
  Output[i] <- Forwardpropagate(Pattren[i], Network)
  BackworkPropagateError(Pattren[i], Output[i], Network)
  UpdateWeight(Pattren[i], OutPut[i], Network, LerningRate)
End
Return(Network)
    
```

1. Inialisasi bobot dengan nilai random yang cukup kecil
2. Melakukan Looping :
 - a. 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).
 - b. Tiap-tiap unit pada suatu lapisan tersembunyi ($Z_j, j=1,2,3,\dots,p$) menjumlahkan sinyal-sinyal *input* terbobot :

$$z_in_j = b1_j + \sum_{i=1}^n x_i v_{ij} \quad (2)$$

gunakan fungsi aktivasi untuk menghitung sinyal *output* :

$$z_j = f(z_in_j)$$

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

Tiap-tiap unit *output* ($Y_k, k=1,2,3,\dots,m$) menjumlahkan sinyal-sinyal *input* terbobot.

$$y_in_k = b2_k + \sum_{i=1}^n z_i w_{jk} \quad (3)$$

- c. gunakan fungsi aktivasi untuk menghitung sinyal *output* :
$$y_k = f(y_in_k)$$

dan kirimkan sinyal tersebut ke semua unit di lapisan atasnya (unit-unit *output*).
- d. Tiap-tiap unit *output* ($Y_k, k=1,2,3,\dots,m$) menerima target pola yang berhubungan dengan pola *input* pembelajaran, hitung informasi errornya :

$$\delta 2_k = (t_k - y_k) f'(y_{in_k}) \tag{4}$$

$$\varphi 2_{jk} = \delta_k z_j \tag{5}$$

$$\beta 2_k = \delta_k \tag{6}$$

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

$$\Delta w_{jk} = \alpha \varphi 2_{jk} \tag{7}$$

hitung juga koreksi bias (yang nantinya akan digunakan untuk memperbaiki nilai b_{2k}) :

$$\Delta b_{2k} = \alpha \beta 2_k \tag{8}$$

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

$$\delta_{in_j} = \sum_{k=1}^m \delta 2_k w_{jk} \tag{9}$$

kalikan nilai ini dengan turunan dari fungsi aktivasinya untuk menghitung informasi *error* :

$$\delta 1_j = \delta_{in_j} f'(z_{in_j}) \tag{10}$$

$$\varphi 1_{ij} = \delta 1_j x_j \tag{11}$$

$$\beta 1_j = \delta 1_j \tag{12}$$

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

$$\Delta v_{ij} = \alpha \varphi 1_{ij} \tag{13}$$

Hitung juga koreksi bias (yang nantinya akan digunakan untuk memperbaiki nilai b_{1j}) :

$$\Delta b_{1j} = \alpha \beta 1_j \tag{14}$$

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

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta w_{jk} \tag{15}$$

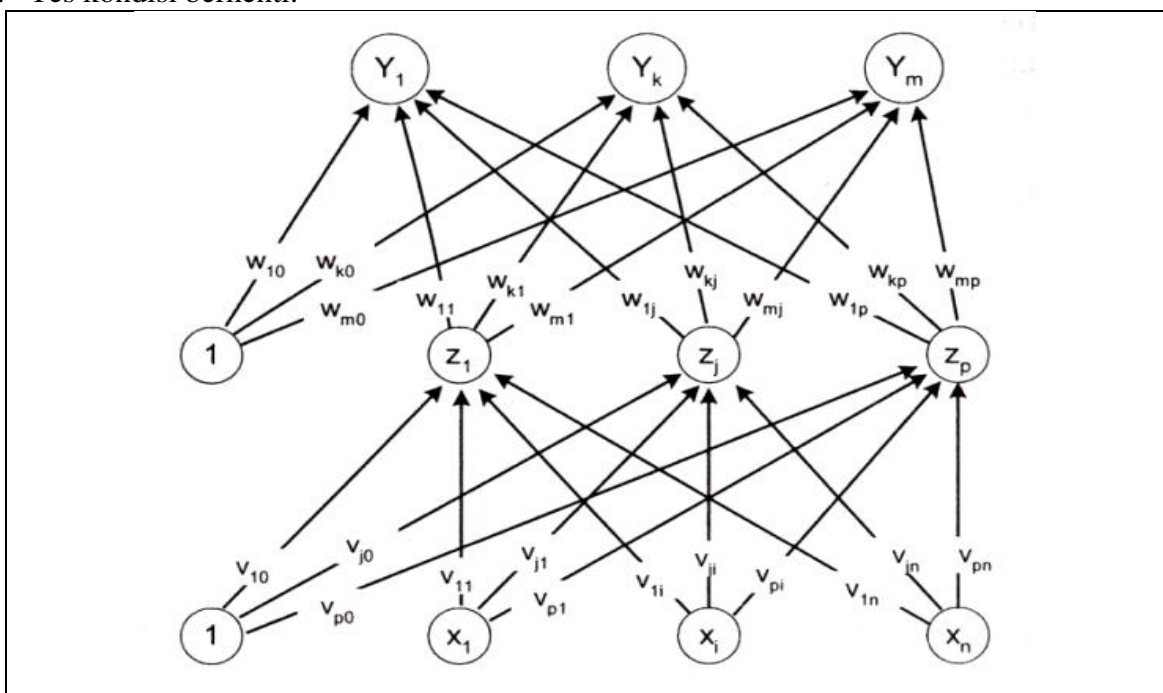
$$b_{2k}(\text{baru}) = b_{2k}(\text{lama}) + \Delta b_{2k} \tag{16}$$

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

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij} \tag{17}$$

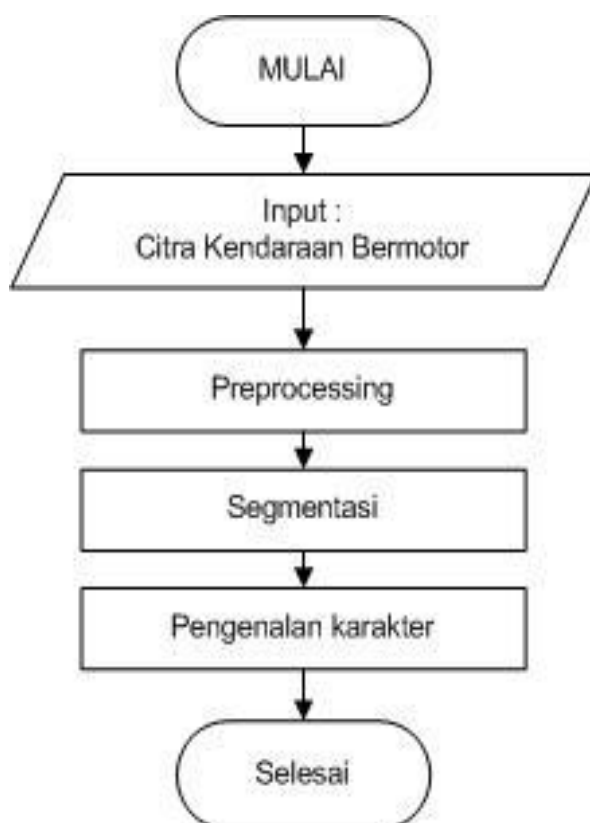
$$b_{1j}(\text{baru}) = b_{1j}(\text{lama}) + \Delta b_{1j} \tag{18}$$

- g. Tes kondisi berhenti.



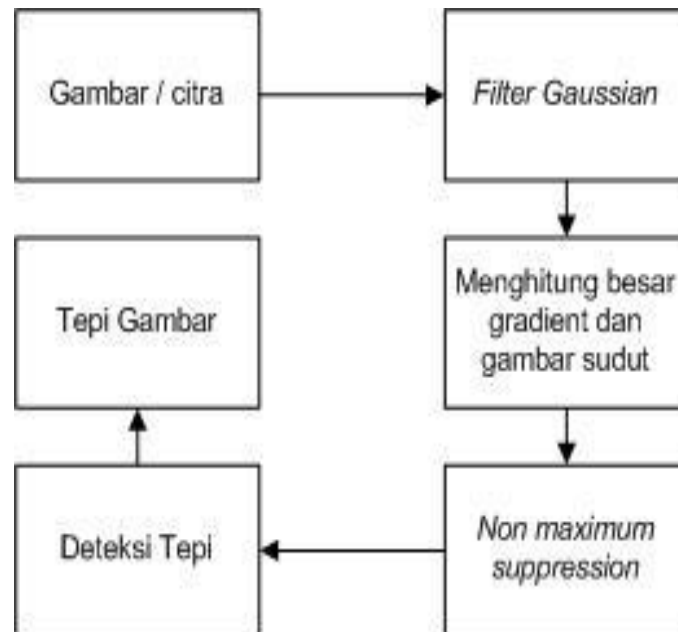
Gambar 1 Arsitektur Multi Layer Perseptron

Secara keseluruhan, sistem kerja pemrosesan citra dibagi menjadi beberapa tahapan yang dapat kita lihat pada gambar 2. Pertama adalah membaca masukan citra yang fungsinya untuk membaca citra masukan RGB yang merupakan citra awal yang akan diproses, serta mengubah citra masukan tersebut menjadi citra greyscale. Langkah selanjutnya adalah Pre-Processing citra. Dalam langkah ini dilakukan perbaikan citra, sehingga lebih mudah dilakukan proses selanjutnya. Setelah proses ini selesai, maka langkah selanjutnya melakukan Segmentasi Karakter. Fungsi dari langkah ini adalah untuk melakukan pemisahan antar objek untuk mendapatkan objek yang diharapkan (karakter plat kendaraan). Pada bagian Segmentasi Karakter ini digunakan metode Connected Component Analysis untuk meng-ekstrak objek yang dianggap sebagai karakter pada plat nomor. Jika telah didapatkan karakter-karakter hasil Segmentasi, maka langkah pengenalan karakter akan diproses. Fungsi dari langkah ini adalah untuk melakukan proses pengenalan karakter plat nomor.



Gambar 2 Flowchart *pre processing* citra Tanda Nomor Kendaraan Bermotor (TNKB)

Proses yang pertama dilakukan adalah mengubah citra masukan berupa kendaraan bermotor menjadi citra greyscale. Selanjutnya dilakukan pemrosesan proses Binarisasi yaitu citra greyscale diubah menjadi citra biner. Selanjutnya dilakukan noise removal dengan metode Canny. Pengambilan citra yang dilakukan pada penelitian ini adalah menggunakan kamera digital sonny cyber-shot. Sehingga dapat dipastikan citra yang dihasilkan menimbulkan derau atau noise. Metode Canny digunakan untuk menghilangkan noise (derau) atau bintik-bintik putih pada citra biner, agar lebih mudah dilakukan proses selanjutnya. Metode ini mempunyai operator khusus seperti metode-metode lain. Metode Canny memiliki langkah tertentu.



Gambar 3 Proses Filter *Canny*

Untuk menentukan rasio filtering karakter pada citra, proses pertama yang dilakukan adalah menentukan letak masing-masing objek pada citra sampel, kemudian menghitung luas, lebar dan tinggi masing-masing objek. Setelah data letak, luas, lebar dan tinggi masing-masing objek diketahui maka dilakukan training pada masing-masing citra. Yaitu dicari objek mana yang merupakan karakter, kemudian nilai dari masing-masing karakter pada citra sampel ditetapkan sebagai rasio filtering.

Nilai rasio yang didapatkan adalah :

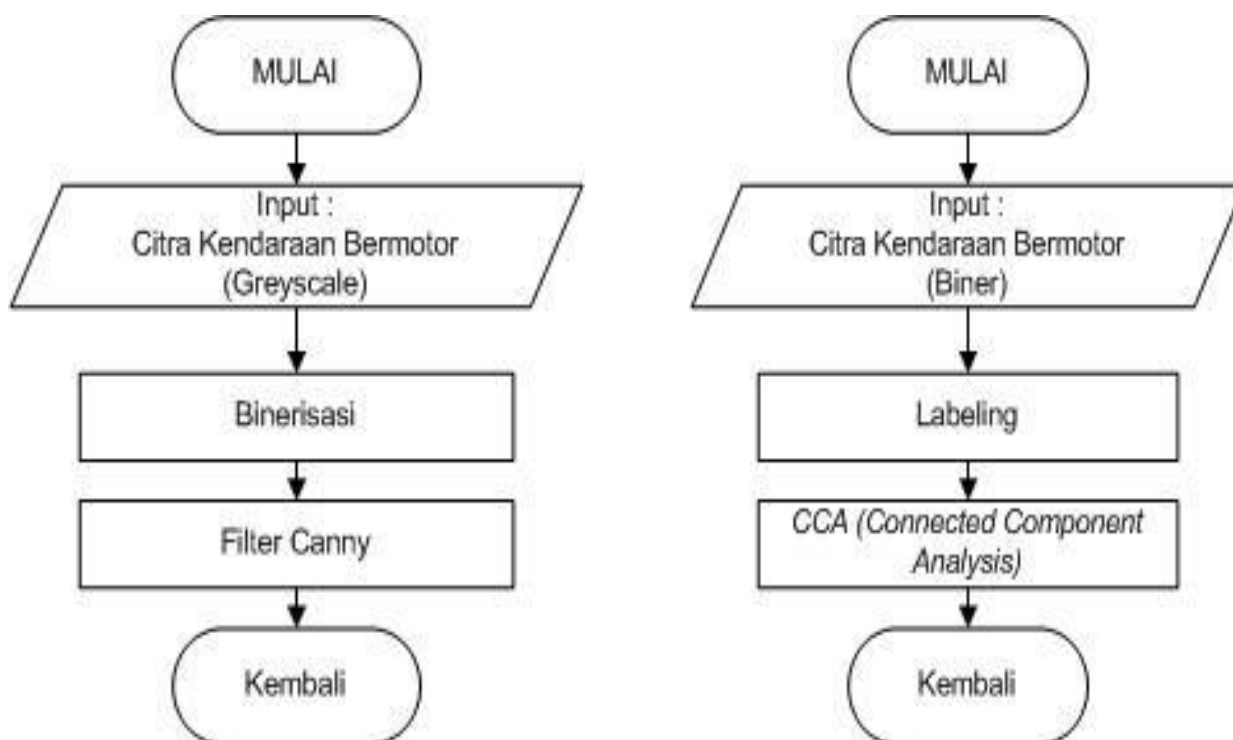
$$\begin{aligned}
 0.0020 &\leq \text{rasio lebar} < 0.090 \\
 0.025 &\leq \text{rasio tinggi} < 0.085 \\
 0.00008 &\leq \text{rasio luas} < 0.0035 \\
 100 &< \text{rasio posisi atas objek} < 400
 \end{aligned}$$

dimana,

$$\text{rasio luas} = \frac{\text{luas objek}}{\text{luas citra}} \tag{19}$$

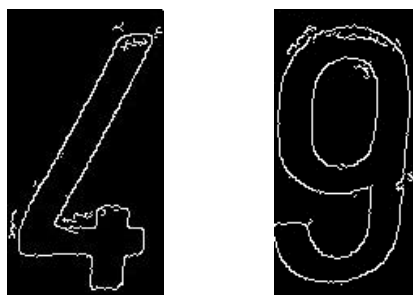
$$\text{rasio lebar} = \frac{\text{posisi kanan objek} - \text{posisi kiri objek}}{\text{lebar citra}} \tag{20}$$

$$\text{rasio tinggi} = \frac{\text{posisi bawah objek} - \text{posisi atas objek}}{\text{tinggi citra}} \tag{21}$$



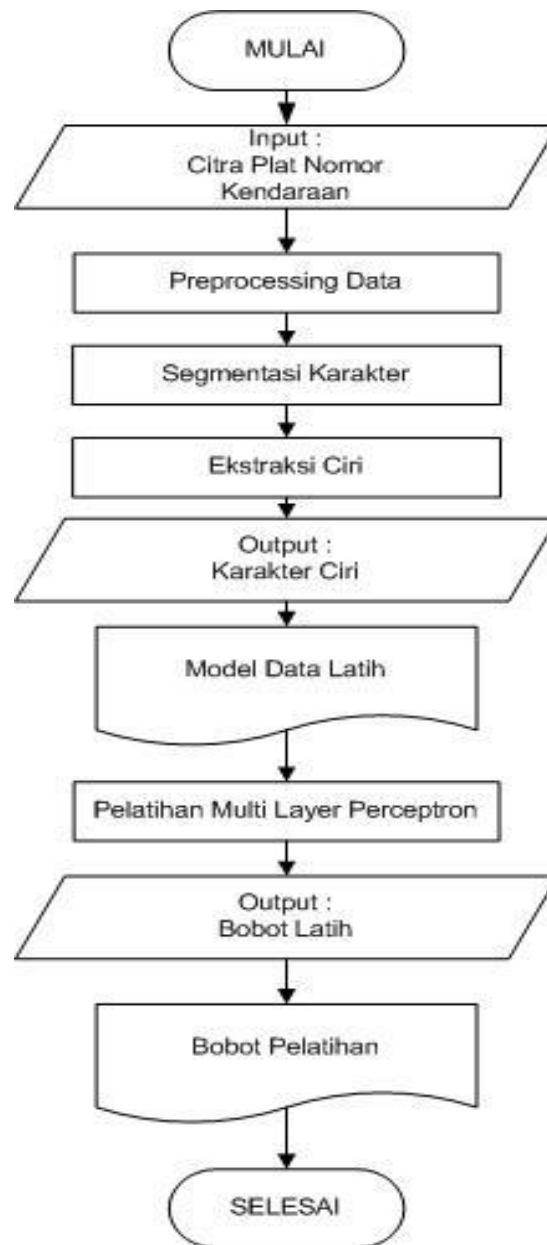
Gambar 4 Flowchart preprocessing dan segmentasi untuk mendapatkan karakter

Untuk objek dengan nilai yang terletak pada rentang komponen-komponen luas, lebar, tinggi dan letak tersebut diatas, maka objek dianggap sebagai karakter, sedangkan sebaliknya dianggap bukan karakter. Selanjutnya, dilakukan proses filtering jarak antar objek agar hasil yang didapatkan lebih maksimal. Pada proses ini, jarak tiap-tiap objek hasil filtering sebelumnya dihitung, kemudian dibandingkan. Maka objek yang memiliki jumlah jarak terbanyak dengan objek lain dianggap bukan karakter. Sedangkan sisanya merupakan karakter yang dicari.



Gambar 5 Sampel Data Training hasil proses segmentasi dan ekstraksi ciri

Secara global, sistem kerja pengenalan karakter kendaraan bermotor ini dilakukan dengan proses pembelajaran menggunakan sampel data dengan skenario dataset yang dapat dilihat pada gambar 6. Proses pengenalan karakter menggunakan Sistem Jaringan Syaraf Tiruan dengan algoritma Backpropagation arsitektur Multi Layer Perceptron yang digunakan menggunakan 3 Layer dengan jumlah masing masing node 84, 50, 36. Input layer 84 disesuaikan dengan jumlah fitur yang digunakan yaitu 12X7 piksel, hidden layer menggunakan 50 sedangkan output layer yang digunakan adalah 36 node sesuai dengan banyaknya class yang digunakan yaitu huruf kapital A sampai dengan Z serta angka dari 0 sampai dengan 9. Jaringan syaraf tiruan dilatih dengan algoritma backpropagation dengan parameter learning rate 0.3 dan momentum 0.9. Proses training akan dihentikan apabila iterasi mencapai nilai maksimal 10.000 atau MSE (Mean Square Error) 0.0001.

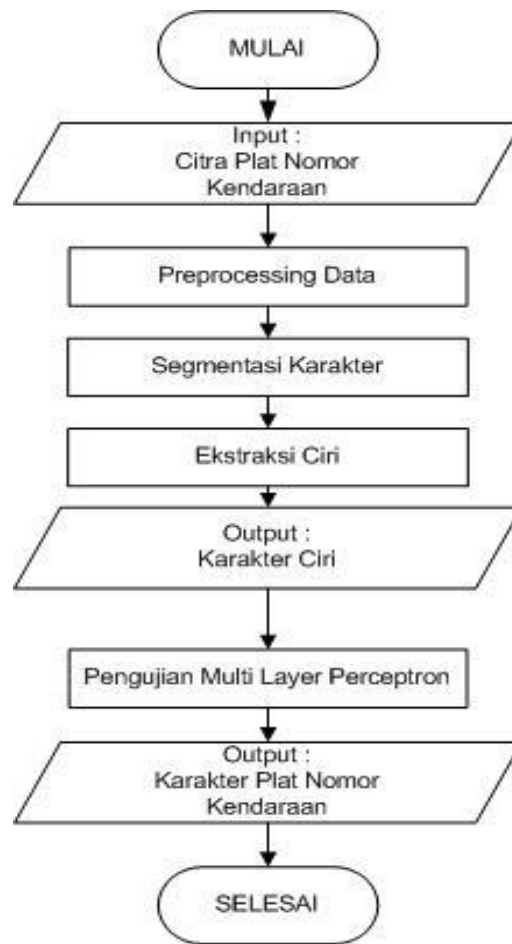


Gambar 6 Proses *Learning* Jaringan syaraf Tiruan

Proses pelatihan yang dilakukan digambarkan pada flowchart gambar 6. Proses pelatihan diawali dengan memasukkan karakter sampel berupa plat nomor kendaraan. Selanjutnya dilakukan proses preprocessing dengan mengkonversi citra sampel kedalam bentuk greyscale. Kemudian dilakukan proses binerisasi yaitu citra greyscale diubah menjadi bentuk biner. Untuk memperbaiki kualitas citra biner maka dilakukan noise removal dengan menggunakan filter canny.

Selanjutnya dilakukan proses segmentasi dengan melakukan pemisahan karakter pada plat nomor kendaraan. Ekstraksi ciri dilakukan untuk mencari karakter pada plat nomor kendaraan dengan melakukan proses labeling pada citra biner yang dijadikan model data latih yang kemudian dilakukan proses pelatihan untuk mendapatkan bobot.

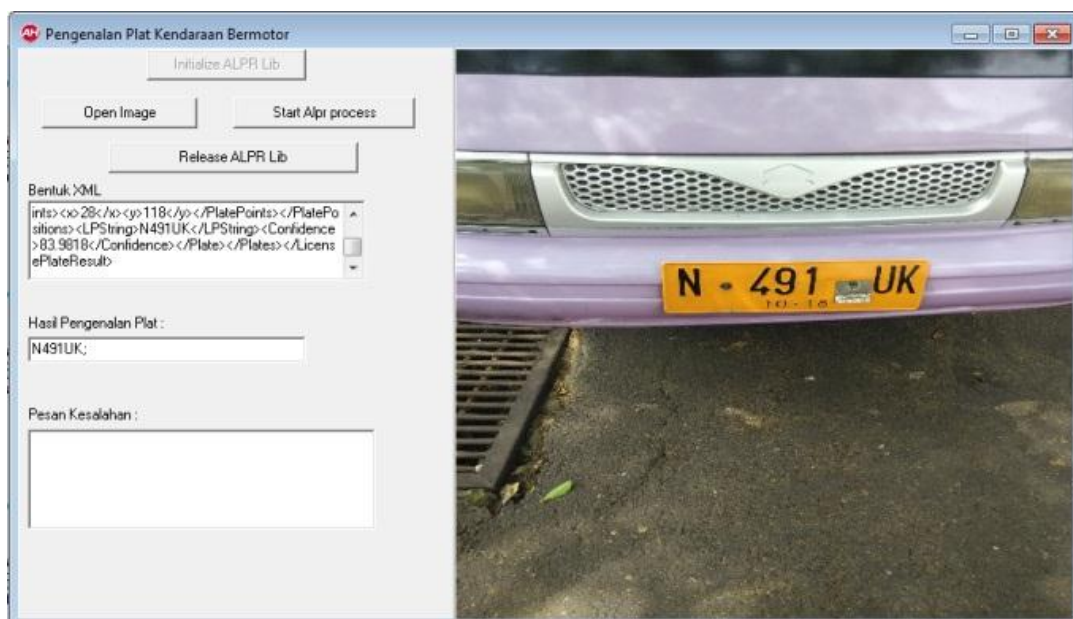
Setelah bobot latih didapat, sistem digunakan untuk mengenali karakter di luar dari sampel yang digunakan. Adapun tahap pengenalan karakter plat nomor kendaraan bermotor dapat dilihat pada flowchart gambar 7. Tahapan proses yang dilakukan hampir sama dengan proses training. Yang menjadi pembeda ada pada output. Pada proses learning hasilnya adalah bobot sedangkan pada proses ini hasilnya merupakan karakter dari plat nomor kendaraan yang telah dikenali.



Gambar 7 Proses Pengujian Jaringan syaraf Tiruan

Hasil peneltian dan pembahasan

Pada uji coba ini digunakan sebanyak 20 data plat nomor kendaraan pengenalan yang dilakukan dengan dua skenario dataset. Dataset1 sistem dilakukan *training* dengan 20 tanda nomor kendaraan bermotor sedangkan dataset2 menggunakan 60 data *training*. Pengambilan data pengujian dilakukan pada siang hari antara pukul 10.00 – 15.00 dengan kondisi cuaca cerah.



Gambar 8 Hasil pengujian sistem pengenalan karakter tanda nomor kendaraan bermotor

Pengujian yang dilakukan adalah dengan mengamati pengenalan karakter tanda nomor kendaraan dengan menghitung presentase keberhasilan dalam mengenali. Pengujian dengan menggunakan skenario dataset1 dapat dilihat pada tabel 1.

Tabel 1 Pengujian dengan data uji dengan menggunakan training dataset1

Plat Nomor	Hasil Pengenalan	Persentase Keberhasilan
S 7234 RA	S 7234 BA	85,71%
S 8062 JA	S 800G2 IA	71,43%
S 8337 JB	S 8337 IB	85,71%
S 7178 UU	S J1JB UU	57,14%
S 8923 JB	S 8928 18	57,14%
D 8596 SR	D 859G SR	71,43%
S 8058 JB	S 80S8 1B	57,14%
S 7110 RA	S 7IIO RA	57,14%
S 7103 UU	S 7IO8 UU	57,14%
S 8195 JB	S 8I9S 18	42,86%
S 7322 UJ	S 7322 U1	85,71%
N 9147 CG	S 9I47 CG	85,71%
S 8032 JB	S 8032 IB	71,43%
S 6274 UU	S G74 UU	85,71%
S 2434 RU	S 2484 RU	85,71%
S 8081 JA	S 8081 1A	71,43%
S 8513 JA	S 8S18 1A	42,86%
S 6133 UU	S G133 UU	85,71%
S 2224 NB	S 2224 NB	100,00%
H 1381 MA	S I88I MA	57,14%

Hasil dataset 1 yang menggunakan 20 data sampel dapat dilihat dalam tabel 1. Hanya 1 plat nomor kendaraan yang berhasil dikenali oleh sistem. Untuk meningkatkan performa maka ditambahkan data trainer yang diskenariokan pada dataset2. Setelah ditambahkan data sampel untuk training ternyata diperoleh hasil peningkatan yang cukup signifikan. Sistem mampu mengenali 11 plat nomor kendaraan secara sempurna seperti yang terlihat dalam tabel 2.

Perhitungan keberhasilan pengenalan karakter adalah dengan membagi jumlah karakter yang telah berhasil dikenali sistem dengan benar dengan keseluruhan karakter pada plat nomor kendaraan. Dirumuskan sebagai berikut :

$$\% \text{ Keberhasilan} = \frac{\text{jumlah karakter yang berhasil dikenali}}{\text{keseluruhan karakter}} \times 100 \%$$

Sebagai contoh pada tabel 1 baris nomor 1, jumlah karakter keseluruhan pada plat nomor kendaraan adalah sebanyak 7 buah S 7243 RA sedangkan sistem mengenali dengan S 7243 BA. Ternyata hanya ada 6 karakter yang berhasil dikenali oleh sistem. Perhitungan persentase keberhasilannya sebagai berikut :

$$\text{Persentase Keberhasilan} = \frac{6}{7} \times 100\% = 85,71 \%$$

Tabel 2 Pengujian dengan data uji dengan menggunakan training dataset2

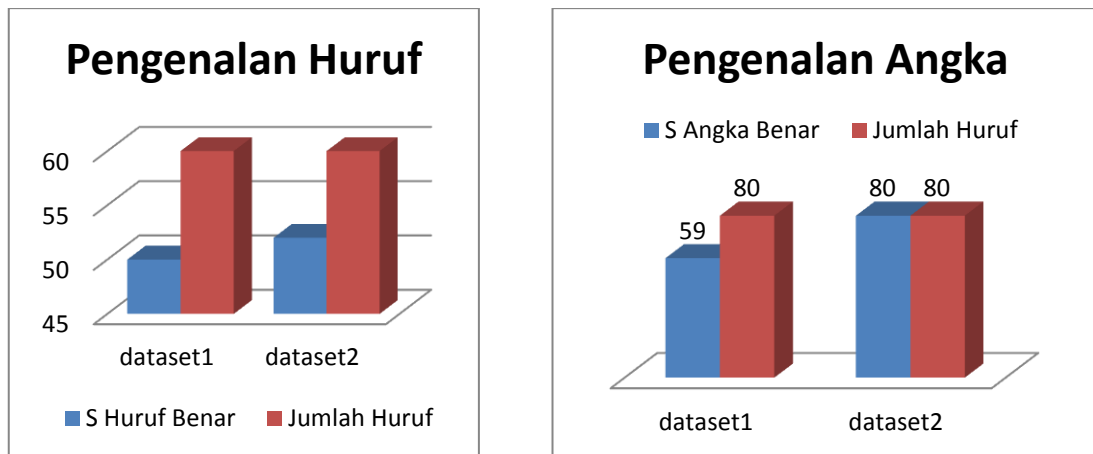
Plat Nomor	Hasil Pengenalan	Persentase Keberhasilan
S 7234 RA	S 7234 RA	100,00%
S 8062 JA	S 8062 JA	85,71%
S 8337 JB	S 8337 IB	85,71%
S 7178 UU	S 7178 UU	100,00%
S 8923 JB	S 8928 1B	100,00%
D 8596 SR	D 8596 SR	100%
S 8058 JB	S 8058 1B	100,00%
S 7110 RA	S 7110 RA	100,00%
S 7103 UU	S 7103 UU	100,00%
S 8195 JB	S 8195 1B	85,71%
S 7322 UJ	S 7322 U1	85,71%
N 9147 CG	N 9147 CG	100,00%
S 8032 JB	S 8032 IB	85,71%
S 6274 UU	S 6274 UU	100,00%
S 2434 RU	S 2434 RU	85,71%
S 8081 JA	S 8081 1A	85,71%
S 8513 JA	S 8518 1A	85,71%
S 6133 UU	S 6133 UU	85,71%
S 2224 NB	S 2224 NB	100,00%
H 1381 MA	H 1381 MA	100,00%

Dari Tabel 1 dan Tabel 2 dapat diperoleh jumlah kesalahan dalam pengenalan baik pada angka maupun huruf. Setiap kesalahan dalam pengenalan karakter dijumlahkan untuk memperoleh persentase keberhasilan dalam pengenalan karakter. Hasil pengolahan data ditunjukkan pada Tabel 3

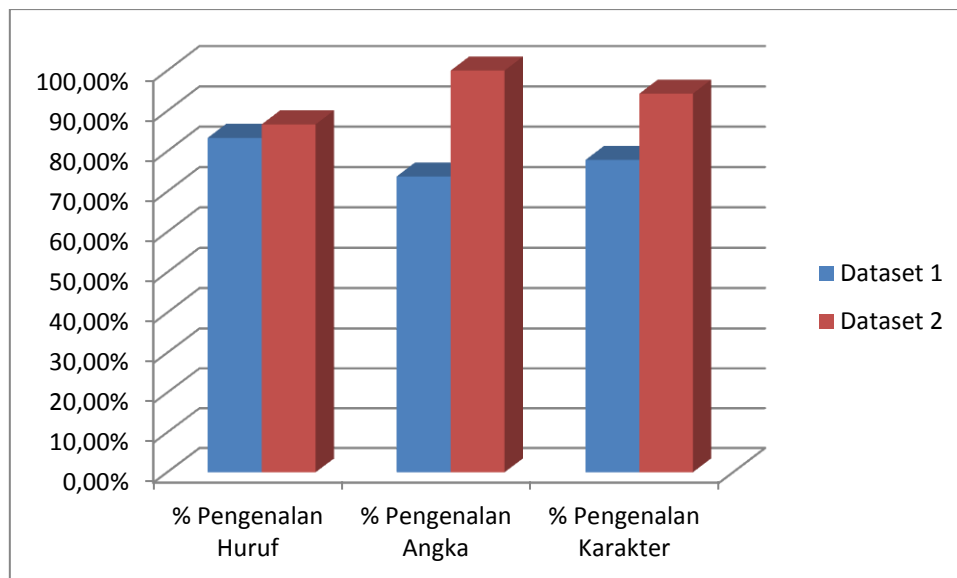
Tabel 3 Pengolahan Hasil Uji Coba Pengenalan Plat Nomor Kendaraan

Data Training	□ Huruf Benar	□ Angka benar	% Pengenalan Huruf	% Pengenalan Angka	% Pengenalan Karakter	% Pengenalan Plat Benar
Dataset1	50	59	83,33%	73,75%	77,86%	5,00%
Dataset2	52	80	86,67%	100,00%	94,29%	55,00%

Dengan menambahkan data training mampu meningkatkan performa pengenalan huruf dari 83,33% menjadi 86,67% dan pengenalan angka dari 73,86% menjadi 100%. Secara keseluruhan sistem mampu mengenali karakter dengan tingkat keberhasilan 94,29%



Gambar 8 Grafik Prosentase Keberhasilan Pengenalan Huruf dan Angka



Gambar 9 Perbandingan Persentase pengenalan huruf dan angka dan keseluruhan karakter

Kesimpulan

Implementasi filter canny dan ekstraksi ciri mampu digunakan untuk mengenali karakter pada Tanda Nomor Kendaraan Bermotor (TNKB). Dengan menggunakan algoritma jaringan syaraf tiruan dengan arsitektur 84 node sebagai input, 50 node untuk hidden layer serta 36 node sebagai output untuk mengenali huruf A sampai dengan Z dan angka mulai 0 sampai dengan 9 dengan data training sejumlah 60 plat kendaraan bermotor mampu mengenali karakter sampai dengan 94,29%.

Daftar Pustaka

- [1]. Baoming shan "License Plate Character Segmentation and Recognition Based on RBF Neural Network " *2010 Second International Workshop on Education Technology and Computer Science*, 2010
- [2]. Choo Kar Soon, Kueh Chiung Lin, Chung Ying Jeng and Shahrel A. Suandi "Malaysian Car Number Plate Detection and Recognition System" *Australian Journal of Basic and Applied Sciences*, 6(3): 49-59, 2012 ISSN 1991-8178, 2012

- [3]. David J. Roberts and Meghann Casanova. 2012 “Automated License Plate Recognition Systems: Policy and Operational Guidance for Law Enforcement ” Federal funds provided by the U.S. Department of Justice and prepared the following final report 2012
- [4]. Kusumadewi, S. . *Artificial Intelligence (Teknik dan Aplikasinya)*. Yogyakarta: Graha Ilmu, 2003
- [5]. Lihong Zheng.; Xiangjian He.; Bijan Samali and Laurence T. Yang. 2010. Accuracy Enhancement for License Plate Recognition 10th *IEEE International Conference on Computer and Information Technology (CIT 2010)*.
- [6]. Manuel D. Rossetti, Jeff Baker “ Application and Evaluation of Automatic License Plate Reading System
- [7]. P.Vijayalakshmi. and M.Sumathi. 2012 “Design of Algorithm for Vehicle Identification by Number Plate Recognition” *IEEE- Fourth International Conference on Advanced Computing, ICoAC 2012 MIT*, Anna University, Chennai. December 13-15, 2012
- [8]. Thomas B Moeslund. . *Canny Edge Detection* .Lecture note of Department of Computer Science and Engineering Indian Institute of Technology Delhi, 2009