

OPTIMASI KEMAMPUAN SEGMENTASI OTSU PADA IDENTIFIKASI PLAT NOMOR KENDARAAN INDONESIA MENGUNAKAN METODE GAUSSIAN

Erwin Dwika Putra¹, Stefanus Santosa²

¹Pasca Sarjana Teknik Informatika Universitas Dian Nuswantoro

²Politeknik Negeri Semarang

¹edphamka@gmail.com

²stefanus.st@gmail.com

Abstrak: Sistem identifikasi plat nomor kendaraan adalah salah satu penelitian yang paling penting dalam bidang perkembangan teknologi *Intelligent Transportation System (ITS)*. Secara umum, terdapat tiga tahap yaitu identifikasi yang disebut juga dengan deteksi, segmentasi, dan pengenalan. Beberapa metode seperti Otsu dan *connected component label* diterapkan dengan akurasi plat nomor kendaraan menurut karakter segmentasi. Penelitian ini berfokus pada peningkatan *pre-processing* menggunakan Otsu thresholding dengan metode Gaussian dan juga terhubung melalui komponen label yang diadopsi untuk membuktikan peningkatan keberhasilan metode *pre-processing*, sehingga dapat meningkatkan hasil segmentasi karakter. Peningkatan dapat dilihat dengan rata-rata perbedaan penurunan nilai MSE menuju $1,75E+07$ pixel dan PSNR yang lebih tinggi yaitu 4db/pixel. Dengan hasil ini, Otsu dengan metode *pre-processing* gaussian lebih baik daripada metode Otsu yang asli.

Kata Kunci: Otsu, Gaussian, *pre-processing*, segmentasi, identifikasi plat nomor kendaraan Indonesia

Abstract: Vehicle plate identification system is one of the most important researches in Intelligent Transportation System (ITS) technology development. Generally, there are 3 stages consisted in this identification which are detection, segmentation, and recognition. Some method like otsu and connected component label are applied to deal with accuracy of vehicle registration plate character segmentation. This research is focused on pre-processing increase using otsu's thresholding method with gaussian function and also connected component label method adoption to prove pre-processing method increase success, therefore it can increase character segmentation result. The increase can be seen with average difference of MSE decreasing value to $1,75E+07$ pixels and PSNR is higher which is 4db/pixels. With this result it means that otsu's with gaussian pre-processing method is better than original otsu's method.

Keywords: otsu, gaussian, pre-processing, segmentation, Indonesian vehicle plate identification

I. PENDAHULUAN

A. Latar Belakang

Alat transportasi yang saat ini mempunyai peranan sangat besar bagi kebutuhan manusia adalah kendaraan bermotor. Hal tersebut menyebabkan meningkatnya penggunaan transportasi kendaraan bermotor. Dengan peningkatan populasi pengguna kendaraan bermotor akan mengakibatkan meningkatnya masalah yang terjadi pada pelanggaran lalu lintas dan juga sistem keamanan. Karena masalah tersebut [1] identifikasi plat nomor kendaraan dapat diimplementasikan untuk mengatasi masalah pengendalian kendaraan yang efektif.

Kendaraan bermotor di Indonesia mempunyai ciri-ciri seperti yang disebutkan pada Peraturan Pemerintah No. 44 tahun 1993 tentang kendaraan bermotor. Ciri tersebut yaitu warna dasar pada plat

nomor kendaraan berwarna hitam dan tulisan berwarna putih, terdapat nomor pajak pada bagian bawah plat nomor kendaraan. Pada identifikasi plat nomor kendaraan tidak dibutuhkan ciri nomor pajak karena identifikasi plat nomor kendaraan akan diterapkan pada sebuah alat yang berfungsi untuk mengenali nomor plat sebuah kendaraan.

Penggunaan alat-alat yang menerapkan identifikasi plat nomor kendaraan sudah banyak digunakan seperti manajemen parkir, pencarian tindak pencurian kendaraan [2], pencarian kendaraan yang diduga sebagai alat perampokan dan sistem pengawasan lalu lintas [3].

Sistem identifikasi plat nomor kendaraan merupakan bagian dari penelitian yang sangat penting dalam pengembangan teknologi *Inteleigent Transportation System* (ITS) [4][5][2]. Masalah yang kerap kali timbul pada identifikasi plat nomor kendaraan diantaranya [6] seperti kondisi pencahayaan plat, bayangan terhadap plat, kotoran-kotoran gambar plat, resolusi gambar plat, keakuratan pemotongan karakter, dan kecepatan dalam mengenali karakter.

Secara umum terdapat 3 tahapan identifikasi plat nomor kendaraan [6][7][8], yaitu deteksi plat nomor, segmentasi karakter, dan pengenalan karakter. Dari ketiga tahapan tersebut bagian sulit dan penting yang dapat mempengaruhi langsung hasil akurasi pengenalan karakter terletak pada bagian segmentasi karakter [9][7].

Faktor yang menentukan tingkat ke sulitannya tahapan segmentasi karakter adalah tahapan sebelum dilakukannya segmentasi karakter yaitu *pre-processing* segmentasi karakter [2][4], beberapa faktor yang menyebabkan sulitnya proses *pre-processing* [4], yaitu:

1. Kondisi pencahayaan yang tidak merata pada plat nomor kendaraan

2. Bingkai dari plat nomor kendaraan
3. Gangguan diakibatkan dari kotoran pada gambar plat nomor
4. Gangguan dari bayangan pada plat nomor kendaraan

Beberapa metode yang telah diusulkan dalam *pre-processing* adalah metode *otsu*[9][4], *radon transform* [7], *edge detection*[7][10][2], *hough transform*[2][6], *bernsen*[4][5], *Histogram*[6], dan sebagainya. Pada segmentasi karakter beberapa metode yang diusulkan adalah metode *varians projection*[9][7][8][5][1][11][4], *Connected Component Label* [12][13], *Hibrid Binarization* [6], dan sebagainya. Sedangkan tahapan pengenalan karakter plat nomor kendaraan beberapa metode yang telah diusulkan adalah *template mathcing*[11][2] dan *support vektor machine* (SVM) [5], dan sebagainya.

Setiap penelitian identifikasi plat nomor kendaraan mempunyai ruang lingkup yang berbeda-beda, ada yang mengerjakan keseluruhan tahapan dan ada yang mengerjakan beberapa tahapan saja, dan juga ada yang mengerjakan 2 tahapan dari identifikasi plat nomor kendaraan.

Penelitian yang mengerjakan keseluruhan dari identifikasi plat nomor kendaraan, misalnya, penelitian dari Ozbay et al [1]. *Mathematical morphology* digunakan pada tahap deteksi plat nomor dengan tingkat akurasi mencapai 97,6%, *varians projection* pada segmentasi tingkat akurasi 96%, dan *template matching* pada pengenalan karakter dengan tingkat akurasi 98,8%.

Penelitian yang ruang lingkungannya hanya satu tahapan, misalnya, penelitian Ma et al [9]. Metode *otsu pre-processing*, karena sangat baik dalam menentukan nilai *threshold* pada binerisasi, *varians projection* segmentasi karakter yang digabungkan dengan *mean filter* untuk

menghilangkan *border* pada plat. Kemudian penelitian Qiao et al [4]. Metode *otsu* dibandingkan dengan metode *bernsen pre-processing* untuk menghilangkan efek kemiringan gambar guna menentukan nilai *threshold* yang tepat dan *varians projection* segmentasi karakter.

Penelitian yang ruang lingkupnya 2 tahapan, misalnya, penelitian Guo et al [6]. Metode berbasis fitur warna pada deteksi plat nomor, metode *histogram* pada *pre-prosesing*, dan *hibrid binarization* pada segmentasi karakter. Wen et al [5]. Metode *otsu* dan *bernsen* dibandingkan untuk *pre-processing* dalam menghilangkan efek bayangan untuk menentukan nilai *threshold* yang tepat. Metode SVM terintegrasi digunakan untuk mengenali tiap karakter hasil segmentasi karakter plat nomor kendaraan dengan tingkat akurasi, angka 99,5%, kana 98,6%, alamat 97,8%.

Dari penelitian di atas *pre-processing*, metode *bernsen* dapat digunakan karena kemampuannya dapat memperbaiki gambar dari segi kemiringan [5][4]. Tetapi metode *bernsen* sangatlah rentan terhadap gangguan yang tidak diperlukan seperti pencahayaan yang berlebihan, bingkai, baut pemasangan plat, dan gangguan akibat dari bayangan [4][5]. Begitu juga dengan kinerja metode *edge detection* juga dapat digunakan untuk *pre-processing* plat nomor kendaraan [2][7] dan akurat dalam mengambil setiap tepi karakter, hanya saja metode ini rentan terhadap gangguan seperti *border*, dan kemiringan gambar [2][10]. Sehingga metode *bernsen* dan *edge detection* kurang akurat untuk *pre-processing* karena akan menyulitkan pada saat segmentasi karakter dan pengenalan karakter.

Sedangkan metode *otsu* sangat baik kinerjanya dalam mengatasi gangguan-gangguan yang tidak diperlukan [9], karena kinerja dari metode *otsu* ini

mencari nilai optimal ambang batas dalam *binarization* dari hasil normalisasi histogram pada proses *pre-processing* [5][10][12], maka akan menemukan nilai ambang batas yang signifikan untuk memisahkan latar belakang plat dan karakter pada plat nomor kendaraan [10]. Dengan demikian metode ini sangat tepat digunakan untuk proses *pre-processing* segmentasi karakter pada plat nomor kendaraan, hanya saja jika nilai ambang batas atau nilai maksimum *varians* antar kelas jauh dari nilai yang diinginkan, segmentasi karakter plat seringkali gagal [10] karena adanya bayangan pada gambar dan gangguan kotoran pada gambar, yang mengakibatkan nilai normalisasi histogram tidak signifikan. Oleh karena itu pendekatan ini memerlukan perbaikan untuk meningkatkan hasil dari proses *pre-processing* segmentasi karakter pada plat nomor kendaraan.

Kekurangan yang terdapat pada metode *otsu* dalam menentukan nilai optimal ambang batas, disebabkan nilai normalisasi histogram yang didapat tidak signifikan untuk memisahkan latar belakang dan karakter. Banyak cara yang dapat menyelesaikan masalah ini, seperti fungsi *gaussian* dan teknik-teknik *filtering*[6], yaitu *mean filter*, *uniform filter*, *median filter*. Penelitian ini menggunakan fungsi *gaussian* untuk mendapatkan rerata dari normalisasi histogram pada metode *otsu*, karena fungsi *gaussian* pada penelitian *image proprocessing* digunakan untuk memperbaiki citra [14] seperti menghilangkan detil, dan menghilangkan gangguan dari citra. Dengan menggabungkan fungsi *gaussian* pada metode *otsu*, penelitian dapat menurunkan nilai ambang batas yang dihasilkan metode *otsu* untuk mengurangi tingkat gangguan yang terdapat pada gambar plat nomor kendaraan.

Setelah *pre-processing* selesai maka dilanjutkan dengan segmentasi tiap karakter pada plat nomor kendaraan, metode segmentasi karakter menggunakan *varians projection* kemampuannya baik dalam segmentasi tiap karakter pada plat nomor kendaraan [5][9], karena dapat mensegmentasi karakter pada sisi vertikal dan horizontal tiap karakter. Hanya saja metode ini sangat rentan terhadap gangguan yang terdapat diantara karakter yang akan disegmentasi dan border pada plat nomor kendaraan [9]. Segmentasi karakter pada plat nomor kendaraan juga dapat menggunakan metode *connected component label* [12] dengan memberikan label pada karakter dan juga latar belakang dengan berdasarkan perhitungan *heuristik* tertentu.

Pada metode *connected component label* memberikan informasi dari setiap segmentasi karakter dalam format yang terdiri dari label terstruktur, dimulai dari baris dan kolom pertama hingga akhir [12][13] dan proses ini sangatlah cepat dan akurat apabila gangguan pada plat telah hilang pada saat *pre-processing*. Tetapi terdapat kelemahan pada metode ini, yang dapat diselesaikan menggunakan metode *thining*[2] yaitu ketebalan yang tidak merata pada segmentasi karakter, yang menyebabkan lambatnya penyimpanan hasil segmentasi karakter. Penelitian ini dimaksudkan untuk memperbaiki hasil metode *connected component label*.

Setelah tiap karakter pada plat nomor kendaraan telah diproses dan didapatkan hasil dari *connected component label* akan digunakan metode *thining* untuk mengatasi ketebalan yang tidak merata pada hasil segmentasi karakter tersebut, karena hasil metode *thining* hanya mengambil 1 piksel dari ketebalan karakter yang telah tersegmentasi [2][14]. Hal ini bertujuan agar

dapat meningkatkan kecepatan dalam proses penyimpanan hasil segmentasi karakter.

Pengukuran hasil segmentasi plat nomor kendaraan, dengan cara mendapatkan nilai *mean square error* (MSE) dan waktu penyimpanan dari hasil segmentasi karakter pada plat nomor kendaraan, akan menggunakan metode *neural network learning vector quantization* (LVQ) untuk mengetahui keberhasilan dari proses *pre-processing* dan hasil segmentasi karakter plat nomor kendaraan.

II. LANDASAN TEORI

A. Metode Otsu

Metode *otsu* biasa disebut juga optimum *thresholdotsu*. Metode ini disebut optimum karena kinerjanya yang memaksimalkan varians antarkelas (*between-class variance*) [16]. Pendekatan varians antarkelas ini digunakan untuk melakukan analisis diskriminan yaitu membedakan antara dua atau lebih kelompok berdasarkan kelas yang sudah ditentukan [14]. Variabel antarkelas akan dapat memisahkan latar belakang (*background*) dan latar depan (*foreground*).

Menurut Gonzales and Woods [16], metode *otsu* didasarkan pada nilai histogramnya. Nilai intensitas dari setiap piksel akan ditunjukkan sembarang oleh histogram pada citra 1 dimensi.

Dengan menggunakan histogram kita dapat melakukan pengelompokan terhadap piksel-piksel dalam citra Sumbu x merupakan level intensitas yang berbeda dan sumbu y menyatakan jumlah piksel yang memiliki nilai intensitas tersebut.

Pengelompokan level intensitas ini didasarkan pada nilai ambang atau *threshold*. Nilai *threshold* ini akan menjadi nilai objektif dan merupakan tujuan dari metode *otsu*, karena dasar dari metode *otsu* adalah perbedaan dari intensitas dari piksel-

piksel yang dipisahkan dalam kelas-kelas tertentu. *Threshold* yang didapat akan memisahkan intensitas pada level-level tertentu hingga didapat nilai *threshold* yang optimal.

Dengan melihat nilai histogram yang berbeda bisa didapatkan dua informasi yaitu, jumlah intensitas yang berbeda (L) dan jumlah piksel-piksel dari setiap level intensitas tersebut (k, dengan k=0...255). Langkah-langkah dari metode *otsu* adalah:

- a. Menghitung nilai histogram yang telah ternormalisasi pada citra (p_i). i merupakan tiap pixel.

$$p_i = \frac{n_i}{N}$$

Dengan n_i adalah jumlah pixel dengan tingkat keabuan i dan N menyatakan banyaknya pixel pada citra

- b. Menghitung jumlah kumulatif ke nol, momen kumulatif ke satu, dan nilai rata-rata dapat dinyatakan sebagai berikut:

$$\omega(k) = \sum_{i=1}^k p_i \quad (1)$$

$$\mu(k) = \sum_{i=1}^k i \cdot p_i \quad (2)$$

$$\mu_t = \sum_{i=1}^L i \cdot p_i \quad (3)$$

- c. Hitung nilai varians antarkelas, untuk mendapatkan nilai maksimum *threshold* (k), dengan persamaan:

$$\sigma_B^2(k) = \frac{[\mu_t \omega(k) - \mu(k)]^2}{\omega(k)[1 - \omega(k)]} \quad (4)$$

- d. Setelah mendapatkan nilai *threshold* yang merupakan index varian kelas maksimum, maka bandingkan jika terdapat dua atau

lebih nilai *threshold* (k^*), reratakan untuk mendapatkan nilai *threshold* tersebut.

Setelah mendapatkan nilai ambang batas atau *threshold* gunakan untuk mendapatkan segmentasi citra tersebut. Nilai optimum yang didapatkan oleh metode *otsu* adalah nilai *threshold* yang diperlukan untuk segmentasi citra, dengan persamaan:

$$g(x, y) = \begin{cases} 1 & \text{if } f(x, y) > k^* \\ 0 & \text{if } f(x, y) \leq k^* \end{cases}$$

B. Fungsi Gaussian

Fungsi *gaussian* sering disebut dengan karakteristik simetrik, fungsi *gaussian* banyak digunakan di berbagai bidang seperti [17], statistik, *signal processing*, *image processing*. Fungsi *gaussian* secara awal mempunyai persamaan:

$$f(x) = a \cdot \exp\left(-\frac{(x-m)^2}{2\sigma^2}\right)$$

Pada *image processing* fungsi *gaussian* mempunyai dua persamaan yaitu, persamaan 1 dimensi (1-D) dapat dinyatakan sebagai berikut:

$$G(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-m)^2}{2\sigma^2}\right) \quad (5)$$

Dengan σ menyatakan standar deviasi dari distribusi, dan m adalah nilai rerata. Sedangkan yang kedua adalah persamaan 2 dimensi (2-D) dapat dinyatakan sebagai berikut:

$$G(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(x-m)^2 + (y-m)^2}{2\sigma^2}\right) \quad (6)$$

C. Metode Connected Component Label

Menurut Putra, Darma [14], *connected component label* adalah pengelompokan piksel-piksel suatu citra yang saling berhubungan dan memeriksa keterhubungan antara piksel-piksel suatu citra. Setiap piksel yang berhubungan akan diberikan tanda atau label yang berbeda. Hal ini bertujuan untuk memisahkan antara *background*

dan *foreground* pada citra, ini merupakan metode yang digunakan pada beberapa aplikasi citra secara otomatis.

Terdapat 2 aturan dalam metode *connected component label*, yaitu *4-connectivity*, dan *8-connectivity*. Langkah-langkah penggunaan masing-masing aturan pada metode ini [14][16], yaitu:

a) *4-connectivity*

- 1) Dilakukan pencarian dari setiap piksel sebuah gambar, dimulai dari baris matriks hingga kolom sampai menemukan titik piksel yang berbeda (p)
- 2) Pada aturan ini setelah ditemukan piksel yang berbeda maka akan diperiksa setiap ketetanggaan dari piksel p, ketetanggaan sebelah kiri dan atas
- 3) Kedua piksel dari ketetanggaan p bernilai 0 maka diberi tanda (label) baru
- 4) Apabila kedua piksel dari ketetanggaan p bernilai 1 maka berilah tanda dari salah satu piksel tetangga tersebut pada p dan buat catatan bahwa kedua tanda yang berbeda tersebut adalah ekuivalen

Pada akhir proses semua piksel yang mempunyai nilai 1 (pada citra biner), telah diberi tanda (label) dan mungkin masih terdapat banyak nilai ekuivalen. Oleh karena itu nilai ekuivalen tersebut diurutkan secara berpasangan ke dalam kelas-kelas ekuivalen kemudian memberi tanda yang berbeda pada setiap kelas ekuivalen.

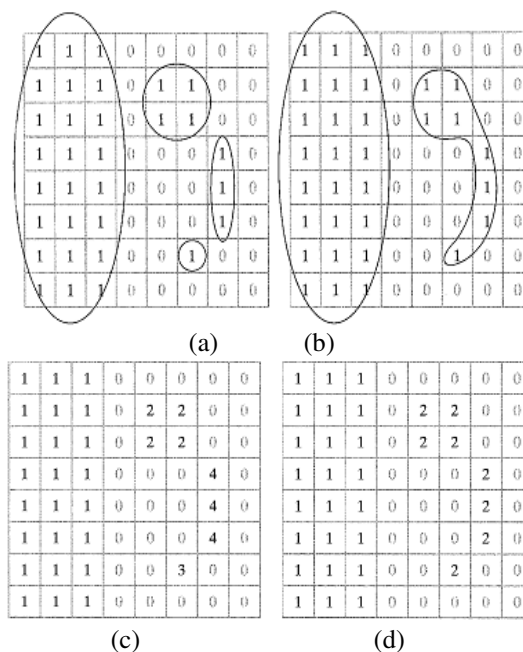
b) *8-connectivity*

Pada prinsipnya langkah-langkah pada *8-connectivity* sama seperti *4-connectivity* hanya saja terdapat sedikit perbedaan yaitu, pada saat pencarian setiap baris apabila *4-connectivity* nilai piksel p telah ditemukan maka yang

dihubungkan yaitu atas dan kiri. Tetapi pada *8-connectivity* apabila nilai p telah ditemukan maka akan menghubungkan setiap piksel dengan memeriksa dari sisi atas, kiri, diagonal atas kiri, dan diagonal atas kanan. Dengan langkah-langkah berikut:

- Bila keempat piksel tetangga bernilai 0 maka berilah tanda baru pada p
- Bila hanya salah satu dari piksel tetangga bernilai 1 maka berilah tanda dari piksel tetangga tersebut pada p
- Bila dua atau lebih piksel tetangga yang bernilai 1 maka berilah salah satu tanda pada p, kemudian semua tanda dari tetangga yang bernilai 1 tersebut adalah ekuivalen.

Proses terakhir dari *4-connectivity* ataupun *8-connectivity* adalah melakukan pemeriksaan atau *scanning* kembali pada citra dan ganti setiap tanda dengan tanda dari kelas ekuivalen. Pada Gambar 1 [16] tampak kinerja dari masing-masing aturan yang ada pada *connected component label*.



Gambar 1. Connected Component Label (a) 4-connectivity, (b) 8-connectivity, (c) Hasil dari 4-connectivity, (d) hasil dari 8-connectivity

D. Metode Thining

Metode *thining* merupakan salah satu teknik *morphologi*, metode ini mendekati metode erosi dan *opening* pada *morphologi*. Pada konsepnya metode *thining* mengubah gambar asli biner menjadi gambar yang hanya menampilkan batas-batas objek/foreground [14], dari gambar biner yang mempunyai lebar beberapa piksel hanya diambil 1 piksel yang merupakan bentuk menyerupai asli dari gambar biner.

Proses yang dilakukan pada metode *thining* hampir menyerupai metode deteksi tepi dalam segi hasil/output. Namun tetap saja ada perbedaan antara metode *thining* dan deteksi tepi, yaitu [14]:

- Deteksi tepi: merubah *gray level* atau intensitas citra menjadi citra yang menampilkan batas-batas/*boundaries* objek berdasarkan kekontrasan warna antarpiksel
- *Thining*: mereduksi piksel pada objek biner menjadi piksel yang bernilai sama dengan nilai piksel pada *background*. Keluaran berupa citra biner dengan informasi berupa batas-batas objek berdasarkan piksel dengan ketebalan satu piksel

Tujuan dari metode *thining* adalah untuk mengurangi piksel objek/foreground pada citra biner, supaya dalam proses penyimpanan suatu objek akan lebih cepat dan efisien.

Operasi *thining* ini [14] dapat dilakukan dengan menggunakan operator *hit and miss*, seperti persamaan berikut:

$$Thin(i, j) = I - HitMiss(i, j) \quad (7)$$

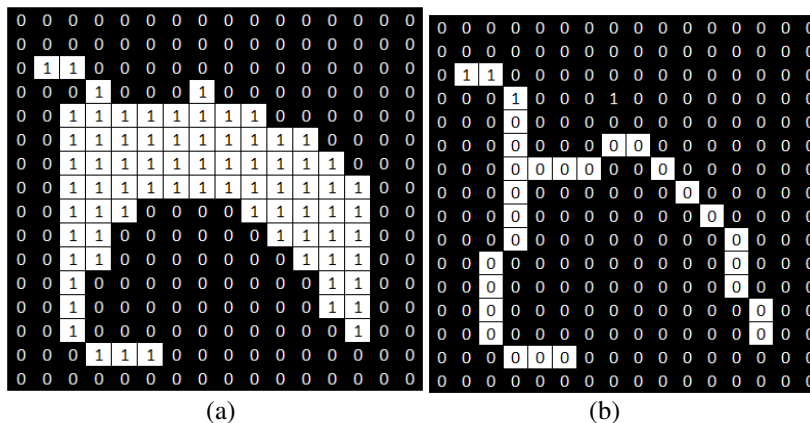
Dengan menggunakan operasi pengurangan di atas dapat dinyatakan pengurangan logis (*logical subtraction*) yang didefinisikan sebagai berikut:

$$x - y = x \cap not y \quad (8)$$

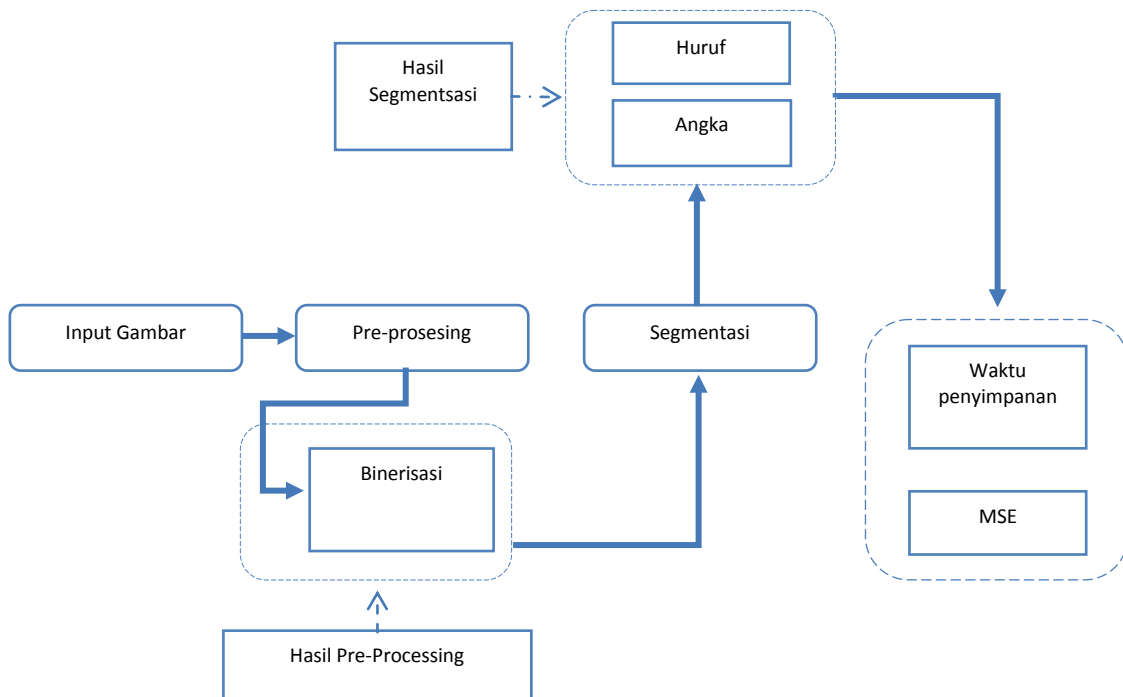
Gambar 2 [14], adalah contoh dari proses metode *thining*. Dari contoh metode *thining* ini, seperti tampak pada citra biner (a) di-*thining* mendapatkan hasil citra dengan tidak merubah dari citra asli (b).

II. OPTIMASI SEGMENTASI OTSU MENGGUNAKAN METODE GAUSSIAN

Dari model di atas, segmentasi karakter dari identifikasi plat nomor kendaraan bermula dari *pre-processing* segmentasi. *Pre-processing* segmentasi merupakan proses awal dan proses paling penting dalam segmentasi karakter plat nomor kendaraan. Pada model segmentasi standard terlihat bahwa proses *pre-processing* hanya dengan menggunakan standard binerisasi untuk memisahkan antara latar belakang plat nomor kendaraan dengan data karakter plat nomor kendaraan seperti yang terlihat pada gambar 3.



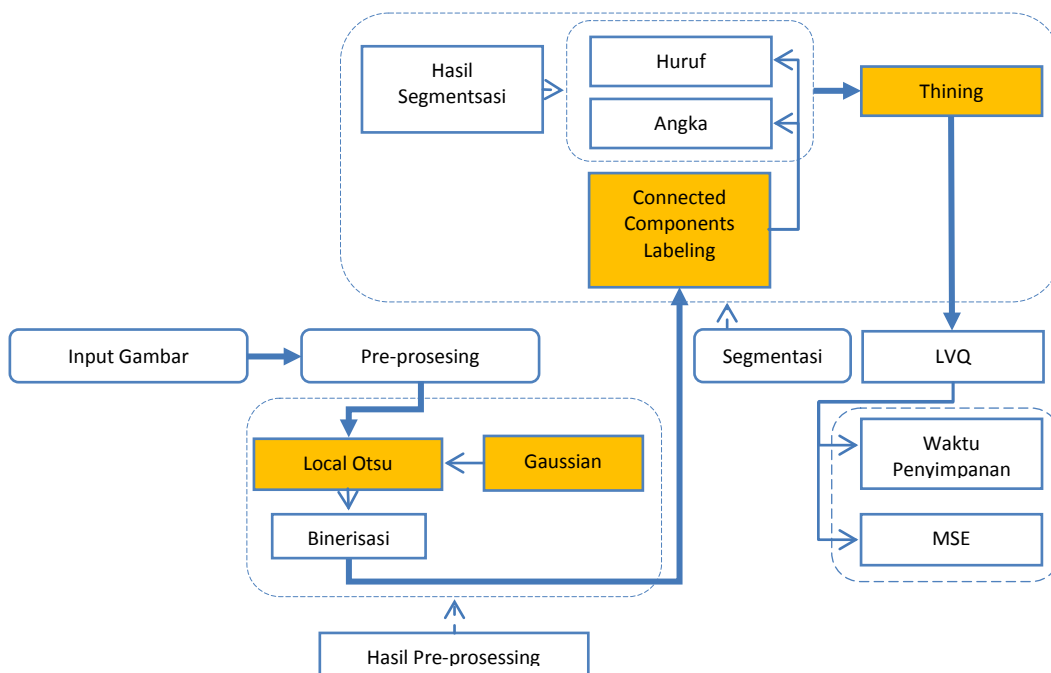
Gambar 2. Contoh Metode Thining



Gambar 3. Model Standart Segmentasi Karakter

Berbeda dengan model standar, model yang diusulkan ini mencari nilai binerisasi atau *threshold* yang didapat dari perhitungan nilai normalisasi histogram menggunakan metode *otsu*. Dari penelitian-penelitian sebelumnya diketahui bahwa kelemahan metode *otsu* adalah dalam menentukan nilai *threshold*, yaitu penentuan nilai

threshold yang terlalu rendah yang disebabkan faktor gangguan pada gambar. Hal ini dapat diperbaiki dengan menggunakan rumus *gaussian* dimensi satu untuk menaikkan nilai *threshold* yang telah ditentukan oleh metode *otsu*. Berikut ini adalah model segmentasi baru yang diusulkan.



Gambar 4. Model Segmentasi yang Diusulkan

Berdasarkan model di atas dapat dilihat bahwa setelah nilai *threshold* didapat melalui metode *gaussian* gambar akan dibinerisasi untuk memisahkan karakter dari latar belakang gambar. Hasil binerisasi selanjutnya akan memotong tiap karakter yang telah terpisah dari latar belakang menggunakan metode *connected component label* (CCL). Metode ini memberikan label pada setiap warna berwarna putih yang dianggap merupakan data karakter. Tetapi hasil dari metode *connected component label* ini menurut *related research* dinyatakan terdapat ketebalan dari setiap data karakter yang ada. Hal ini akan mempengaruhi waktu penyimpanan data untuk hasil segmentasi.

Untuk mengatasi hal tersebut penelitian ini mengusulkan adopsi metode *thinning* untuk mendapatkan satu piksel dari setiap data hasil segmentasi. Hasil dari segmentasi dan waktu penyimpanan diukur menggunakan algoritma *neural network learning vector quatization* (LVQ). Dari hasil pengujian di dapatkan *mean square error* (MSE) dari hasil segmentasi pembaharuan metode *otsu-gaussian* dan waktu penyimpanan penambahan metode *thinning* dari hasil segmentasi karakter metode *connected component label* (CCL).

III. METODE PENELITIAN

A. Pengumpulan Data

Data yang digunakan adalah data yang langsung diambil dari foto-foto kendaraan dengan kondisi *outdoor* (lapangan parkir) dan *indoor* (parkir dalam ruangan), pada waktu pagi, siang dan sore hari. Karakteristik dari plat nomor kendaraan yang diambil adalah plat nomor kendaraan standar Indonesia yang ditetapkan sesuai dengan Peraturan Pemerintah No.44 tahun 1993 dan surat Kapolri kepada Kepolisian tingkat daerah (Polda) No: ST/810/IV/2011 tanggal 25

April 2011, tentang Perubahan Ciri-ciri Tanda Nomor Kendaraan Bermotor (TNKB) dan Tanda Coba Kendaraan Bermotor. Jumlah keseluruhan data yang didapatkan sebanyak 130 gambar plat nomor kendaraan standar Indonesia.

B. Pengolahan Data

Gambar plat nomor kendaraan dipisahkan dari gambar kendaraan dengan memotong bagian plat nomor kendaraan sesuai dengan tujuan penelitian ini yang berfokus pada segmentasi karakter plat nomor kendaraan. Hasil dari pemotongan plat nomor kendaraan dan hasil dari segmentasi karakter plat nomor kendaraan disimpan pada beberapa folder, yaitu folder sampel data, data *training*, dan data *testing*.

Untuk pemisahan tersebut digunakan *tool Sofonesia Picture Cropper*. Penggunaan *tool* ini dikarenakan setiap hasil pemotongan gambar tidak akan merubah ukuran dari gambar aslinya. Hasil pemotongan tampak pada gambar 5.

Setelah hasil sampel data plat nomor kendaraan didapatkan selanjutnya sampel plat nomor kendaraan dinormlisasi ukuran dimensinya menjadi 237x82. Tentu tidak efisien apabila dimensi semua gambar diubah secara manual. Di sini digunakan software *image processing* yang dapat menangani *multi file* gambar yang bernama *Multiple Image Resizer.Net*.

C. Eksperimen dan Pengujian

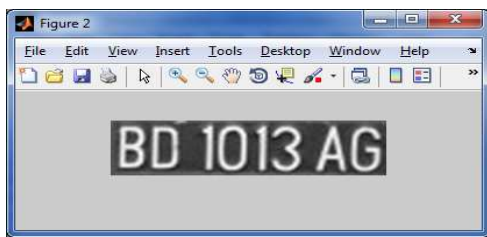
Pada eksperimen dan pengujian metode pada penelitian ini *tool* yang digunakan adalah MATLAB. Penggunaan *tool* ini dimulai dari awal penelitian *pre-processing* sampai dengan didapatkan hasil *mean square error* dan waktu penyimpanan setiap data. Dibawah ini akan dijelaskan tahapan-tahapan penelitian yang dilakukan berdasarkan model yang diusulkan.



Gambar 5. Sampel Data Plat Nomor Kendaraan

• *Pre-Processing*

Pada *pre-processing* tahapan awal setelah pemanggilan setiap gambar dilakukan proses perubahan jenis gambar dari tiga warna red, green, blue (RGB) menjadi abu-abu (*grayscale*) yang bertujuan untuk memudahkan dalam pengambilan bagian karakter pada plat nomor kendaraan yang bukan bagian dari karakter pajak kendaraan. Seperti tampak pada gambar.



Gambar 6. Bagian Karakter Plat Nomor Kendaraan

Setelah didapatkan bagian karakter plat nomor kendaraan dilanjutkan dengan proses berdasarkan metode otsu dengan langkah-langkah sebagai berikut:

1. Menghitung nilai normalisasi histogram gambar bagian karakter plat nomor kendaraan menggunakan persamaan normalisasi histogram jumlah frekuensi.

$$new_{count} = \frac{count}{\sum count} \quad (9)$$

Ket:

- *count* adalah jumlah nilai setiap piksel yang terdapat pada gambar.
 - *sum count* adalah jumlah keseluruhan nilai piksel pada gambar
2. Bilai nilai normalisasi histogram telah didapat maka selanjutnya akan dihitung jumlah kumulatif ke nol dari, $\omega(k)$ untuk $k = 0,1,2,\dots,L-1$ dengan persamaan berikut:

$$\omega(k) = \sum_{i=1}^k p_i \quad (10)$$

3. Menghitung nilai momen kumulatif ke satu dari, $\mu(k)$ untuk $k = 0,1,2,\dots,L-1$ dengan persamaan berikut:

$$\mu(k) = \sum_{i=1}^k i \cdot p_i \quad (11)$$

4. Menghitung nilai rata-rata menggunakan persamaan berikut:

$$\mu_r = \sum_{i=1}^L i \cdot p_i \quad (12)$$

5. Untuk mendapatkan nilai *threshold*, maka dilanjutkan dengan menghitung nilai varians antarkelas menggunakan persamaan berikut:

$$\sigma_B^2(k) = \frac{[\mu_t \omega(k) - \mu(k)]^2}{\omega(k)[1 - \omega(k)]} \quad (13)$$

6. Setelah mendapatkan nilai *threshold* dari metode *otsu*, nilai *threshold* yang didapatkan diperbaiki dengan menggunakan fungsi *gaussian* dengan perbandingan *if T <= 0,5*. Angka perbandingan 0,5 diambil karena nilai rata-rata ambang batas pada binerisasi bernilai antara 0 sampai dengan 1. Hal ini dapat dilakukan dengan persamaan.

$$k^* = \begin{cases} \text{if } \sigma_B^2(k) \leq 0,5 \text{ then } \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x^*)^2}{2\sigma^2}\right) \\ \text{else } \sigma_B^2(k) \end{cases} \quad (14)$$

Ket:

- Nilai *x* dan σ didapat dari nilai *threshold* awal metode *otsu*.

7. Setelah mendapatkan nilai *threshold* dari persamaan di atas maka dilakukan binerisasi menggunakan fungsi *im2bw* dari *tool matlab*.

- Segmentasi

Hasil binerisasi dari pembaharuan metode *otsu*, dilanjutkan dengan memotong setiap karakter pada plat nomor kendaraan dengan menggunakan metode *connected component label*. Cara kerja dari metode *connected component label* dengan langkah-langkah berikut:

1. memberikan penandaan dari setiap piksel yang ditemukan, proses dalam penandaan pada metode ini dengan menggunakan aturan 8-*connectivity*. Dimana penandaan atau pelabelan pada setiap piksel dilakukan dengan pemeriksaan atau scanning kepada citra, apabila nilai piksel pertama (*p*) ditemukan maka akan dikoreksi dari setiap piksel yang

sama dari sisi atas, kiri, diagonal atas kiri, dan diagonal atas kanan.

2. Setelah penandaan selesai dilanjutkan dengan pencarian setiap piksel yang telah diberi label untuk dipindahkan ke dalam sebuah matrix kosong, dengan ukuran matrix yang disesuaikan oleh tinggi dan lebar dari karakter dari gambar original yang telah diberi label.
3. Hasil pemotong dari pelabelan telah selesai menggunakan metode *connected component label* belum akan diproses untuk penyimpanan, karena menurut *related research* bahwa terdapat ketebalan pada hasil metode ini, maka diterapkan metode *thinning* untuk mendapatkan satu piksel dari hasil metode *connected component label*.
4. Operasi *thining* ini [14] dapat dilakukan dengan menggunakan operator *hit and miss*, seperti persamaan berikut:

$$Thin(i, j) = 1 - HitMiss(i, j) \quad (15)$$

5. Dengan menggunakan operasi pengurangan di atas dapat dinyatakan pengurangan logis (*logical subtraction*) yang didefinisikan sebagai berikut:

$$x - y = x \cap \text{not } y \quad (16)$$

IV. HASIL DAN PEMBAHASAN

A. Pre-processing

Berdasarkan penelitian yang dilakukan pembaharuan pada metode *otsu* dengan menggunakan fungsi *gaussian*, nilai awal menggunakan nilai *threshold* yang dihasilkan oleh metode *otsu*, sedangkan nilai *x* pada fungsi *gaussian* menginputkan nilai-nilai yang ditentukan. Penentuan nilai *x* berdasarkan nilai *threshold* yang dihasilkan oleh original *otsu*, seperti dijelaskan pada tabel.

Tabel 1. Nilai x Fungsi Gaussian

Threshold T_i (Original Otsu)	Nilai x fungsi gaussian
if $T_i \leq 0,6$	- 0,2
if $T_i \leq 0,5$	- 0,4
if $T_i \leq 0,4$	- 0,5
If $T_i \leq 0,3$	- 0,47

Nilai x pada fungsi gaussian ditentukan untuk mendapatkan nilai *threshold* agar dapat meningkat dari nilai *threshold* sebelumnya, dengan tujuan mendapatkan kondisi warna hitam lebih dominan dan warna putih sedikit berkurang dari setiap pikselnya.

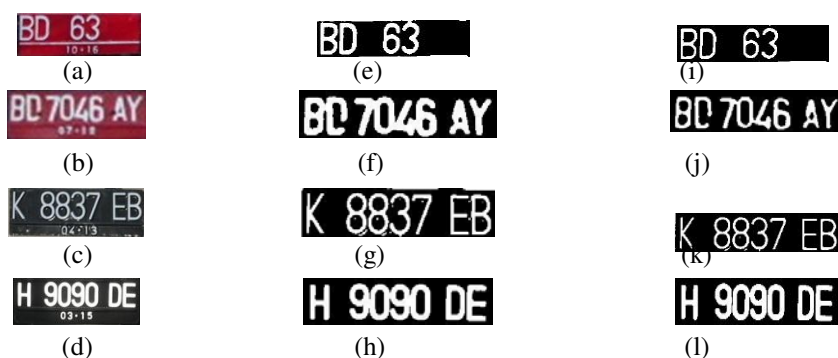
Pada penelitian ini ditemukan pula dalam penentuan nilai x pada fungsi gaussian apabila nilai x pada fungsi gaussian di atas 0,5 maka nilai *threshold* yang didapatkan akan lebih rendah dari *threshold* original metode *otsu*. Hal ini berakibat kondisi warna putih lebih dominan dan warna hitam semakin berkurang sehingga menyebabkan banyaknya gangguan pada gambar.

Tampak dari Gambar 7 (e) hasil binerisasi dari metode *otsu* terdapat *border* bagian atas dan bawah dengan nilai *threshold* yang didapatkan 0,4392, merupakan nilai *threshold* $\leq 0,4$ maka nilai *threshold* akan ditingkatkan oleh metode *otsu-gaussian* dengan hasil 0,6000, dari peningkatan nilai *threshold* tampak pada Gambar 7 (i) bahwa *border* atas hilang dan bawah terlihat menipis. Begitu juga pada Gambar 7 (c) terdapat gangguan titik-titik putih pada gambar hasil

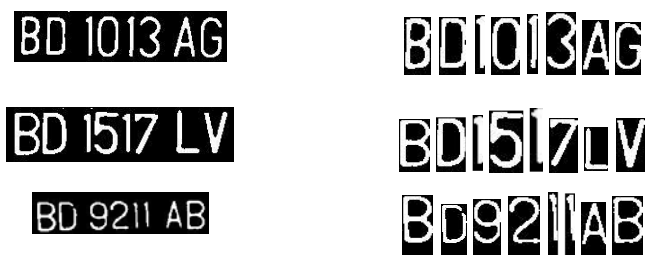
binerisasi metode *otsu*, dengan nilai *threshold* 0,4431, ditingkatkan dengan hasil 0,5990 sedikitnya gangguan kotoran pada gambar plat nomor kendaraan berkurang.

Dengan meningkatkan nilai *threshold* yang didapatkan metode *otsu* dengan fungsi gaussian bertujuan untuk meningkatkan nilai intensitas warna hitam dan mengurangi intensitas warna putih. Maka dapat disimpulkan bahwa metode yang diusulkan lebih baik dari pada metode original *otsu* yaitu dapat mengurangi gangguan seperti *border* dan titik-titik putih yang merupakan gangguan kotoran pada gambar plat nomor kendaraan walaupun untuk gangguan akibat baut dan *border* pada kendaraan tidak dapat hilang sepenuhnya. *Segmentasi Karakter*

Segmentasi merupakan proses pembagian bagian yang diinginkan dari keseluruhan gambar tergantung dari kriteria-kriteria yang diinginkan, seperti berdasarkan warna abu-abu, nilai dari piksel dalam gambar, atau mungkin tergantung pada fitur lain dari gambar. Dalam hal binerisasi, segmentasi ditunjukkan pada perbedaan komponen latar belakang dan latar depan. Pada segmentasi karakter, karakter merupakan latar depan dari sebuah gambar. Hasil segmentasi karakter menggunakan metode CCL dan binerisasi menggunakan metode yang diusulkan *otsu-gaussian* dapat dilihat pada Gambar 8.



Gambar 7. Perbandingan Metode Original Otsu dan Otsu-Gaussian (a)-(d) Original Image; (e)-(h) Original Otsu; (i)-(l) Otsu-Gaussian



Gambar 8. Hasil Segmentasi Karakter Connected Component Label (CCL)

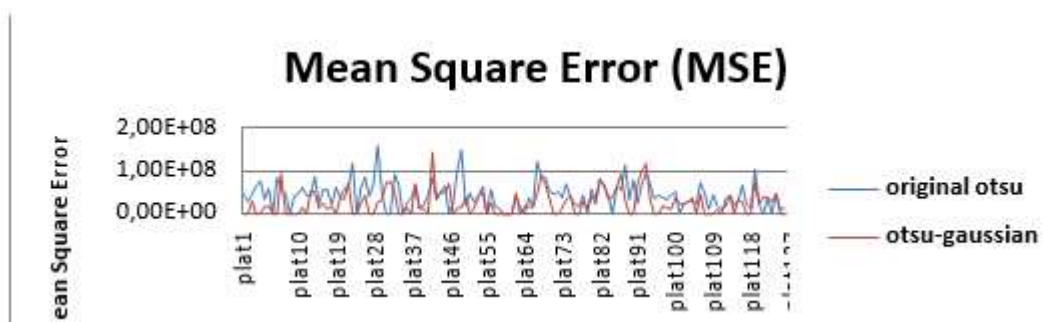
Hasil pengujian dari model ini yaitu nilai *mean square error* (MSE) dan *peak signal to noise ratio* (PSNR). Hasil MSE dan PSNR didapat dari perbandingan data gambar target yang harus dicapai dengan data hasil metode yang diusulkan *otsu-gaussian* dan metode original *otsu*. Hasil MSE dan PSNR dapat dilihat pada grafik Gambar 9 dan Gambar 10.

Terlihat dari data hasil rata-rata MSE dan PSNR yang didapat bahwa untuk mengurangi

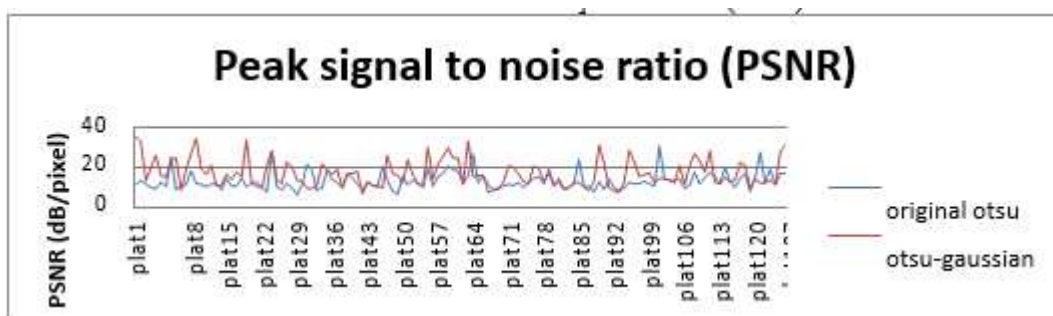
gangguan yang ada pada gambar plat nomor kendaraan model *otsu-gaussian* lebih baik dari pada metode original *otsu*, dengan selisih rata-rata MSE $1,75E+07$ pixel dan rata-rata PSNR yaitu 4 db/pixel, seperti dijelaskan pada tabel 2.

Tabel 2. Tabel Rata-Rata Hasil Pengukuran Plat

Penguk	Original	Otsu-gaussian
<i>Mean square error</i> (MSE)	4,37E+07 piksel	3,E+07 piksel
<i>Peak signal to noise ratio</i> (PSNR)	13 dB/piksel	17 dB/piksel



Gambar 9. Grafik Mean Square Error (MSE)



Gambar 10. Grafik Peak Signal to Noise Ratio (PSNR)

V. PENUTUP

Dari hasil penelitian ini dapat disimpulkan bahwa terdapat perbaikan model original *otsu*. Perbaikan diperoleh melalui peran fungsi *gaussian* pada tahap *pre-processing*. Nilai kesalahan yang diukur dengan *mean square error* (MSE) adalah $3,0 \times 10^7$ piksel dan *peak signal to noise ratio* (PSNR) sebesar 17dB/piksel dari seluruh plat nomor kendaraan. Hasil ini lebih baik dibandingkan dengan menggunakan metode original *otsu*. Nilai kesalahan MSE yang dicapai lebih tinggi, yakni $4,37 \times 10^7$ piksel dan PSNR 13dB/piksel dari seluruh plat nomor kendaraan.

Selama proses penelitian berlangsung juga terdapat temuan bahwa apabila nilai x pada fungsi *gaussian* ditentukan di atas 0,5 maka akan menyebabkan menurunnya nilai *threshold*, yang mengakibatkan intensitas warna putih lebih dominan dari pada warna hitam, dan hal tersebut. Hal ini berefek lebih buruk pada gambar plat karena akan menambah gangguan pada gambar plat nomor kendaraan.

Kelebihan dari model penelitian yang ditawarkan ini (proposed model) adalah dapat mengurangi gangguan-gangguan pada gambar plat dengan menurunkan intensitas warna putih dan meningkatkan intensitas warna hitam pada hasil binerisasi segmentasi plat nomor kendaraan. Sebaliknya masih terdapat juga kelemahan dari model ini yaitu gangguan berupa *border* yang belum dapat hilang sempurna, gangguan akibat terdapat baut pada plat, dan akibat faktor pencahayaan yang kurang merata.

REFERENSI

- [1] S. Ozbay, "Automatic vehicle identification by plate recognition," *World Academy of Science, Engineering and*, pp. 222-225, 2005.
- [2] J. Quan, Q. Shuhai, and S. Ying, "A fast license plate segmentation and recognition method based on the modified template matching," *Image and Signal*, 2009.
- [3] K.-ming Hung and C.-tang Hsieh, "A Real-Time Mobile Vehicle License Plate Detection and Recognition," *Science*, vol. 13, no. 4, pp. 433-442, 2010.
- [4] S. Qiao, Y. Zhu, X. Li, and T. Liu, "Research of improving the accuracy of license plate character segmentation," *Frontier of Computer*, pp. 489-493, 2010.
- [5] Y. Wen, Y. Lu, J. Yan, Z. Zhou, K. M. von Deneen, and P. Shi, "An Algorithm for License Plate Recognition Applied to Intelligent Transportation System," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 12, no. 99, pp. 1-16, 2011.
- [6] J. Guo, "License plate localization and character segmentation with feedback self-learning and hybrid binarization techniques," *Vehicular Technology, IEEE Transactions on*, vol. 57, no. 3, pp. 1417-1424, 2008.
- [7] X. Jia, X. Wang, W. Li, and H. Wang, "A novel algorithm for character segmentation of degraded license plate based on prior knowledge," in *Automation and Logistics, 2007 IEEE International on Conference*, 2007, pp. 249-253.
- [8] H. Xia, "The study of license plate character segmentation algorithm based on vertical projection," *Consumer Electronics, Communications and*, pp. 4583-4586, 2011.
- [9] Y. Ma, J. Chi, R. Hu, and G. Yang, "A new algorithm for characters segmentation of license plate based on variance projection and mean filter," *2011 IEEE 5th International Conference on Cybernetics and Intelligent Systems (CIS)*, pp. 132-135, Sep. 2011.
- [10] J. Park, "An intelligent framework of illumination effects elimination for Car License Plate character segmentation," *Machine Learning and Cybernetics (ICMLC)*, 2010, no. July, pp. 11-14, 2010.
- [11] J. Wang, W. Zhou, and J. Xue, "The research and realization of vehicle license plate character segmentation and recognition technology," *Pattern Recognition*, no. July, pp. 11-14, 2010.
- [12] S. Saha, "Automatic Localization and Recognition of License Plate Characters for Indian Vehicles," *International Journal of Computer Science &*, no. 4, pp. 520-533, 2011.
- [13] F. Benzarti and H. Amiri, "License Plate Detection Using Mathematical Morphology," *Sciences-New York*, pp. 1-5, 2012.
- [14] D. Putra, *Pengolahan Citra Digital*, 1st ed. Yogyakarta: Andi Yogyakarta, 2010.
- [15] "Peraturan Pemerintah No 44," 1993.
- [16] R. C. Gonzalez. and R. E. Woods., *Digital Image Processing Using MATLAB third edition*, 3rd ed. 2007.
- [17] H. Guo, "A Simple Algorithm for Fitting a Gaussian Function [DSP Tips and Tricks]," *IEEE Signal Processing Magazine*, vol. 28, no. 5, pp. 134-137, Sep. 2011.