TUGAS AKHIR – EE 184801

DETEKSI PLAT NOMOR KENDARAAN BERMOTOR MENGGUNAKAN *SLIDING CONCENTRIC WINDOWS* (SCW) UNTUK APLIKASI SISTEM TRANSPORTASI CERDAS

Rangga Imantaka Firswandy
NRP. 07111240000188

Dosen Pembimbing
Sri Rahayu, S.T., M.Kom.
Dr. Ir. Achmad Affandi, DEA.

DEPARTEMEN TEKNIK ELEKTRO
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember
Surabaya 2019

FINAL PROJECT – EE 184801

MOTOR VEHICLE NUMBER PLATE DETECTION USING SLIDING CONCENTRIC WINDOWS (SCW) METHODS FOR INTELLIGENT TRANSPORTATION SYSTEM APPLICATION

Rangga Imantaka Firswandy
NRP. 07111240000188

Supervisors
Sri Rahayu, S.T., M.Kom.
Dr. Ir. Achmad Affandi, DEA.

DEPARTMENT OF ELECTRICAL ENGINEERING
Faculty of Electrical Technology
Sepuluh Nopember Institute of Technology
Surabaya 2019

PERNYATAAN KEASLIAN
TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan "Deteksi Plat Nomor Kendaraan Bermotor Menggunakan *Sliding Concentric Windows* (SCW) untuk Aplikasi Sistem Transportasi Cerdas" adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, Januari 2019

Rangga Imantaka Firswandy
NRP. 07111240000188

Halaman ini sengaja dikosongkan

DETEKSI PLAT NOMOR KENDARAAN BERMOTOR MENGGUNAKAN METODE *SLIDING CONCENTRIC WINDOWS* (SCW) UNTUK APLIKASI SISTEM TRANSPORTASI CERDAS

TUGAS AKHIR

Diajukan untuk Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik
Pada
Bidang Studi Telekomunikasi Multimedia
Departemen Teknik Elektro
Institut Teknologi Sepuluh Nopember

Menyetujui:

Dosen Pembimbing 1	Dosen Pembimbing 2
Sri Rahayu, S.T., M.Kom.	Dr. Ir. Achmad Affandi, DEA.
NIP. 196802281997022001	NIP. 196510141990021001

KEMENTERIAN RISET, TEKNOLOGI, DAN PENDIDIKAN TINGGI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
FAKULTAS TEKNOLOGI ELEKTRO
DEPARTEMEN TEKNIK ELEKTRO

SURABAYA
JANUARI, 2019

Halaman ini sengaja dikosongkan

DETEKSI PLAT NOMOR KENDARAAN BERMOTOR MENGGUNAKAN METODE *SLIDING CONCENTRIC WINDOWS* (SCW) UNTUK APLIKASI SISTEM TRANSPORTASI CERDAS

Nama : **Rangga Imantaka Firswandy**
Pembimbing : **1. Sri Rahayu, S.T., M.Kom.**
2. Dr. Ir. Achmad Affandi, DEA.

ABSTRAK

Sistem transportasi cerdas semakin menjadi kebutuhan saat ini, sehingga berbagai teknik dan algoritma *License Plate-Recognition* (LPR) terus dikembangkan untuk aplikasi pengenalan plat nomor kendaraan bermotor. Namun, berbagai kendala masih banyak ditemui dikarenakan kondisi plat nomor yang tidak standar (warna, bentuk, ukuran dan pola), faktor cuaca, *level* pencahayaan, dan resolusi kamera pindai turut menentukan pengaruh kualitas citra yang akan dideteksi.

Fokus dari tugas akhir ini adalah bagaimana mengintegrasikan teknik segmentasi SCW dalam sistem LPR dapat dioperasikan dalam kondisi *outdoor* jika parameter-parameter uji yang digunakan diatur secara baik, sehingga dapat menjawab permasalahan yang menjadi kendala dalam LPR. Metode yang digunakan dalam tugas akhir ini merupakan integrasi dari berbagai tahapan proses yang terdiri dari *pra-processing* (*grayscale*), metode segmentasi (SCW, *image masking* (Sauvola), dan *Connected Component Analysis* (*aspect ratio*, orientasi, dan bilangan Euler). Untuk merealisasikan integrasinya dinyatakan dalam susunan algoritma deteksi plat nomor kendaraan, yang diawali dengan pencarian *Region of Interest* (RoI).

Setelah dilakukan pengujian pada *dataset* citra uji yaitu yang terdiri dari 20 citra plat nomor, maka dapat diambil kesimpulan bahwa berhasil terdeteksi sebanyak 14 buah (70%), terdeteksi sebagian sebesar 1 buah (5%), dan tidak terdeteksi sama sekali sebanyak 5 buah (25%).

Kata kunci : citra, deteksi, plat nomor kendaraan, SCW

Halaman ini sengaja dikosongkan

MOTOR VEHICLE NUMBER PLATE DETECTION USING SLIDING CONCENTRIC WINDOWS (SCW) METHODS FOR INTELLIGENT TRANSPORTATION SYSTEM APPLICATION

Name : **Rangga Imantaka Firswandy**
Advisor : **1. Sri Rahayu, S.T., M.Kom.**
2. Dr. Ir. Achmad Affandi, DEA.

ABSTRACT

Intelligent transportation systems are increasingly becoming a necessity nowadays, so that License Plate-Recognition (LPR) techniques and algorithms various continue to be developed for applications to recognize motorized license plates. However, many obstacles are still encountered due to nonstandard number plate conditions (color, shape, size and pattern), weather factors, lighting levels, and resolution of the scan camera also determine the influence of the quality of the image to be detected.

The focus of this final project is how to integrate the SCW's segmentation technique in the LPR's system can be operated in outdoor conditions if the test parameters used are set properly, so that it can answer the problems that are constraints in LPR. The method used in this thesis is an integration of various stages of the process consisting of pre-processing (grayscale), segmentation methods (SCW, image masking (Sauvola), and Connected Component Analysis (aspect ratio, orientation, and Euler numbers)). For realizing its integration is expressed in the arrangement of vehicle number plate detection algorithms, which begins with the search for Region of Interest (RoI).

After testing on the test image dataset, which consists of 20 number plate images, it can be concluded that 14 detected (70%) were detected, semi-detected by 1 pieces (5%), and 5 were not detected at all (25%).

***Keywords**:* Vehicle Number Plate, Image, SCW, Detection

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh.

Alhamdulillah segala puji syukur kami panjatkan kehadirat Allah SWT, karena atas limpahan rahmat dan hidayah-Nya penulis dapat menyelesaikan Tugas Akhir yang berjudul:

"Deteksi Plat Nomor Kendaraan Bermotor Menggunakan Metode *Sliding Concentric Windows* (SCW) untuk Aplikasi Sistem Transportasi Cerdas"

Tugas akhir ini merupakan salah satu syarat yang harus dipenuhi dalam menyelesaikan program studi Sarjana pada Departemen Teknik Elektro, Fakultas Teknologi Elektro, Institut Teknologi Sepuluh Nopember. Dalam proses pengerjaan penelitian ini, penulis banyak mendapat saran, dorongan, bimbingan serta keterangan-keterangan dari berbagai pihak. Oleh karena itu, dengan segala hormat dan kerendahan hati penulis mengucapkan terima kasih kepada:

1. Pihak keluarga yang selalu mendoakan, mengingatkan, dan memberikan dukungan kepada penulis.
2. Sri Rahayu S.T., M.Kom. selaku Dosen Pembimbing I dan Dr. Ir. Achmad Affandi, DEA. selaku Dosen Pembimbing II yang telah memberikan ilmu, pengarahan, dan bantuan secara teknis ketika penelitian berlangsung.
3. Seluruh dosen dan karyawan Teknik Elektro FTE-ITS yang telah membantu dan membimbing penulis selama menjalani proses perkuliahan.
4. Seluruh mahasiswa bidang studi Telekomunikasi Multimedia Teknik Elektro FTE-ITS yang telah memberi dukungan kepada penulis.
5. Seluruh mahasiswa Teknik Elektro FTE-ITS angkatan 2012 yang memberikan motivasi kepada penulis agar dapat menyelesaikan proses perkuliahan dengan sukses dan lancar.
6. Semua pihak yang telah memberikan dukungannya, yang tidak dapat disebutkan satu persatu dalam laporan ini.

Dalam proses perbaikan dan pengembangan kritik dan saran terhadap penelitian ini sangat dibutuhkan untuk mengetahui kekurangan dan kebutuhannya. Akhir kata penulis ucapkan terima kasih.

Wassalamu'alaikum Warahmatullahi Wabarakatuh.

Surabaya, Januari 2019

Penulis

DAFTAR ISI

LEMBAR PERSETUJUAN i
ABSTRAK iii
ABSTRACT v
KATA PENGANTAR vii
DAFTAR ISI ix
DAFTAR GAMBAR xi
DAFTAR TABEL xiii
BAB 1 PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Permasalahan dan Batasannya 1
1.3 Tujuan 2
1.4 Metodologi Penelitian 2
1.5 Sistematika Penulisan 4
1.6 Relevansi 5
BAB 2 TINJAUAN PUSTAKA 7
2.1 Plat Nomor Kendaraan Bermotor 7
2.1.1 Manfaat Tanda Nomor Kendaraan Bermotor 8
2.1.2 Tanda Nomor Kendaraan Bermotor di Indonesia 8
2.2 Citra Digital 9
2.2.1 Konsep Piksel 9
2.2.2 Parameter Citra Digital 10
2.3 Citra RGB 11
2.4 Citra Keabuan (*Grayscale*) 12
2.5 Segmentasi Citra 13
2.6 *Thresholding* 14
2.7 *Image Masking* 14
2.8 Deteksi Tepi (Metode *Canny*) 15
2.9 Kontur 17
2.10 Metode Analisis Citra Digital 17
2.10.1 *Aspect Ratio (AR)* 18
2.10.2 *Orientation* 18
2.10.3 Bilangan Euler (*Euler Number*) 20
2.11 Matlab 21
BAB 3 METODE PENELITIAN 23
3.1 Arsitektur Umum 23
3.2 Rancangan Desain Sistem 24
3.2.1 Algoritma Sistem Deteksi Plat Nomor 26

3.2.2 Metode *Sliding Concentric Windows* (SCW) 27
3.2.3 Operator & *Thresholding Image Masking* 28
3.2.4 Metode *Connected Component Analysis* (CCA) 29
3.3 Skenario Pengujian 30
3.3.1 Langkah Percobaan..... 31
3.3.2 Kriteria Keterdeteksian..... 32
BAB 4 HASIL PENGUKURAN DAN ANALISIS 35
4.1 Kriteria Kedeteksian Dalam Pengujian 35
4.2 Hasil Pengujian 36
4.2.1 Hasil Pengujian Visual 36
4.2.2 Hasil Pengukuran Parameter Uji..... 37
4.3 Analisis dan Pembahasan..... 38
4.4 Perbandingan Hasil Pengujian 41
BAB 5 PENUTUP..... 43
5.1 Kesimpulan 43
5.2 Saran 43
DAFTAR PUSTAKA..... 45
LAMPIRAN A : *Listing* Program..... 47
LAMPIRAN B : Hasil Deteksi Plat Nomor Kendaraan..... 63
RIWAYAT HIDUP..... 67

DAFTAR GAMBAR

Gambar 1.1 Rancang Sistem Deteksi Plat Nomor Kendaraan 3
Gambar 2.1 Plat Nomor Kendaraan Bermotor di Indonesia 8
Gambar 2.2 Sistem Koordinat Citra Digital 9
Gambar 2.3 Contoh Citra RGB 11
Gambar 2.4 Hasil Citra Keabuan (*Grayscale*) 12
Gambar 2.5 Contoh Hasil Segmentasi Citra 13
Gambar 2.6 Hasil Binerisasi Citra Digital 14
Gambar 2.7 Contoh Hasil *Masking* Citra 15
Gambar 2.8 Contoh Proses Deteksi Tepi (Metode *Canny*) 16
Gambar 2.9 Kontur tertutup 17
Gambar 2.10 Kontur Terbuka 17
Gambar 2.11 Sumbu momen kedua terkecil 20
Gambar 2.12 Bilangan Euler 20
Gambar 2.13 Program MATLAB versi 2014a 22
Gambar 3.1 Diagram Blok Metode Penelitian 23
Gambar 3.2 Diagram Alur Pencarian Lokasi Plat Nomor 25
Gambar 3.3 Tahapan Proses Deteksi Plat Nomor 26
Gambar 3.4 (a) Jendela Konsentris dan (b) Pemindaian Citra 27
Gambar 3.5 Tahapan Proses Pengujian yang Berhasil 32
Gambar 3.6 Tahapan Proses Pengujian yang Berhasil Sebagian 33
Gambar 3.7 Tahapan Proses Pengujian yang Berhasil 33
Gambar 4.1 Tingkat Kedeteksian Plat Nomor 41
Gambar 4.2 Persentase Keberhasilan Pengujian Plat Nomor 42

Halaman ini sengaja dikosongkan

DAFTAR TABEL

Tabel 3.1 *Licensed Plate Detection Algorithm* 26
Tabel 3.2 Algoritma SCW 27
Tabel 3.3 *Dataset* Citra Uji 30
Tabel 4.1 Kriteria Kedeteksian secara Visual..... 35
Tabel 4.2 Hasil Pengujian 36
Tabel 4.3 Hasil Pengukuran Parameter Utama/Uji..... 37
Tabel 4.4 Pengaruh Nilai CCA terhadap Keberhasilan Deteksi 38

Halaman ini sengaja dikosongkan

BAB 1
PENDAHULUAN

1.1 Latar Belakang

Sistem transportasi cerdas semakin menjadi kebutuhan saat ini, sehingga berbagai teknik dan algoritma *License Plate-Recognition* (LPR) terus dikembangkan untuk aplikasi pengenalan plat nomor kendaraan bermotor. Namun, berbagai kendala masih banyak ditemui dikarenakan kondisi plat nomor yang tidak standar (warna, bentuk, ukuran dan pola), faktor cuaca, level pencahayaan, dan resolusi kamera pindah turut menentukan pengaruh kualitas citra yang akan dideteksi.

Umumnya, ada tiga tahapan utama untuk pemrosesan pengenalan objek dalam sebuah citra (plat nomor kendaraan) yaitu proses pendeteksian (*image detection*), normalisasi citra (*cropping*, terdapat *grayscale*, binerisasi, dan sebagainya) dan pengklasifikasi/pengenalan citra. Sedangkan dalam kasus pendeteksian plat nomor kendaraan, proses yang dilakukan sedikitnya meliputi deteksi lokasi plat nomor *(license plate detection)*, segmentasi karakter (*character segmentation*) dalam plat, dan pengenalan/klasifikasi karakter (*character recognition).*

Tugas akhir ini akan dibahas salah satu sistem pengenalan plat nomor kendaraan bermotor (LPR) menggunakan metode *sliding concentric windows* (SCW). Beberapa tahapan yang dilakukan meliputi *pra-processing* (pengambilan citra, normalisasi citra, mengubah menjadi citra keabuan/*grayscale*), segmentasi citra, dan pendeteksian citra plat nomor. Metode SCW digunakan untuk mensegmentasi dalam citra plat nomor.

Selanjutnya, akan dilakukan pengujian terhadap sistem deteksi plat nomor kendaraan bermotor yang dibuat pada plat nomor kendaraan roda empat di Indonesia dan akan dievaluasi tingkat keberhasilannya berdasarkan persentase deteksi terhadap karakter dari plat nomor tersebut.

1.2 Permasalahan dan Batasannya

Beberapa permasalahan yang dibahas dalam tugas akhir tentang deteksi plat nomor kendaraan bermotor adalah sebagai berikut:

1. Bagaimana membuat desain dalam sistem deteksi plat nomor kendaraan bermotor.
2. Bagaimana memahami cara kerja sistem deteksi citra plat nomor.

3. Bagaimana mengambil data penting dalam citra plat nomor untuk keperluan deteksi lokasi plat nomor.
4. Bagaimana cara mengevaluasi tingkat keberhasilan sistem pendeteksian citra plat nomor.

Terdapat beberapa hal yang menjadi perhatian dan sekaligus menjadi batasan dalam penelitian tugas akhir ini yaitu:

1. Jenis plat nomor kendaraan yang digunakan untuk pengujian adalah plat nomor kendaraan mobil roda empat di Indonesia.
2. Citra plat nomor yang digunakan untuk pengujian deteksi sudah dalam kondisi ternormalisasi sebelumnya.
3. Metode segmentasi untuk mengekstraksi fitur citra plat nomor adalah menggunakan teknik *Sliding Concentric Windows* (SCW).
4. Jumlah plat nomor yang diuji sebanyak 20 buah dengan mengambil foto sendiri.
5. Plat nomor uji yang digunakan hanya plat nomor Indonesia, dimana karakter plat berwarna putih dengan background warna hitam, merah, dan kuning.
6. Bahasa pemrograman yang digunakan untuk merealisasi sistem deteksi plat nomor kendaraan adalah Matlab versi 2014.

1.3 Tujuan

Adapun tujuan dari pembuatan tugas akhir ini adalah untuk menjawab berbagai permasalahan tentang deteksi plat nomor kendaraan bermotor, antara lain:

1. Membuat desain deteksi plat nomor kendaraan bermotor, yang memenuhi kriteria sistem deteksi karakter dalam citra plat nomor kendaraan bermotor.
2. Menerapkan metode SCW untuk keperluan segmentasi citra dalam plat nomor.
3. Melakukan pengujian tingkat keberhasilan terhadap sistem deteksi plat nomor yang dibuat.

1.4 Metodologi Penelitian

Adapun metodologi yang digunakan dalam penyelesaian tugas akhir ini terdiri dari:

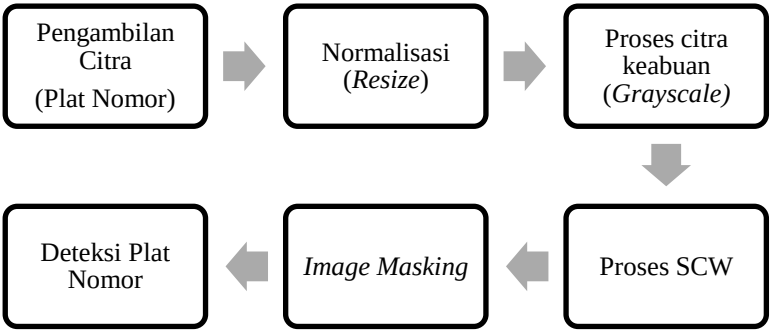
1. Studi Literatur

Studi literatur dilakukan dengan mempelajari beberapa referensi, baik berupa buku, jurnal, maupun sumber penelitian lain yang berhubungan dengan sistem deteksi plat nomor kendaraan bermotor yang menunjang tugas akhir ini. Studi literatur juga mempelajari perangkat lunak (*software*) yang akan digunakan selama penelitian berlangsung. Materi-materi yang akan dibahas dalam penelitian ini, antara lain:

a. Plat nomor kendaraan bermotor
b. Citra digital
c. Teknik *sliding concentric windows* (SCW)
d. Deteksi tepi (dengan menggunakan metode *Canny*)
e. *Connected Component Analysis* (CCA)
f. MATLAB

2. Perancangan Sistem

Membuat rancangan sistem deteksi plat nomor kendaraan bermotor yang akan dibuat dalam tugas akhir ini.



Gambar 1.1 Rancang Sistem Deteksi Plat Nomor Kendaraan

3. Simulasi Sistem

Membuat program sistem deteksi plat nomor kendaraan bermotor yang dilakukan dengan program MATLAB, yang menerapkan algoritma *sliding concentric windows* (SCW) untuk segmentasi karakter dalam plat nomor kendaraan bermotor.

4. **Pengujian Sistem**

Serangkaian pengujian sistem deteksi plat nomor yang dibuat pada jenis plat kendaraan mobil beroda empat dilakukan dengan mengambil plat nomor kendaraan sebanyak 20 citra uji, kemudian akan ditentukan plat nomornya akan terdeteksi secara penuh, terdeteksi sebagian, atau tidak terdeteksi sama sekali.

5. **Pengolahan Data Hasil Pengujian**

Pengolahan data dari hasil pengujian dilakukan berdasarkan data yang telah didapatkan, kemudian ditentukan tingkat keberhasilan deteksi plat nomor kendaraan bermotor dan dianalisis.

6. **Analisis Hasil dan Menarik Kesimpulan**

Melakukan pembahasan atau analisis terhadap hasil pengujian yang telah dilakukan sebelumnya untuk mendapatkan nilai tingkat keberhasilan mendeteksi karakter pada plat nomor kendaraan sepeda motor yang diuji. Selanjutnya dilakukan pengambilan kesimpulan.

7. **Penulisan Laporan**

Melakukan penulisan buku tugas akhir yang memuat semua aspek tugas akhir (mulai dari tinjauan pustaka hingga kesimpulan) sesuai dengan format yang berlaku.

1.5 Sistematika Penulisan

Proses penulisan yang tersusun pada laporan tugas akhir ini dapat dijabarkan sebagai berikut.

1. BAB I

   Bab pertama berisi latar belakang, perumusan masalah, batasan masalah, tujuan penelitian, metodologi penelitian, sistematika penelitian, dan manfaat penelitian dalam pengerjaan tugas akhir.
2. BAB II

   Bab ini menjelaskan dasar teori yang akan menjadi acuan dalam peneltian tugas akhir ini, seperti pemahaman tentang pengolahan citra digital, pengenalan plat nomor kendaraan, dan *sliding concentric windows* (SCW)
3. BAB III

   Pembahasan pada bab ini terkait dengan penjelasan rancangan sistem deteksi plat nomor kendaraan bermotor, metodologi pemecahan masalah, *flowchart* program dan algoritma.

4. BAB IV
   Bab ini membahas hasil dan analisis data dari sistem deteksi plat nomor kendaraan bermotor dengan metode *sliding concentric windows* (SCW) dengan menggunakan *software* Matlab.
5. BAB V
   Bab ini berisi kesimpulan dan saran yang diambil berdasarkan analisis dari hasil pengujian plat nomor kendaraan bermotor.

1.6 Relevansi

Dengan sistem deteksi yang dihasilkan dalam tugas akhir ini diharapkan dapat dimanfaatkan untuk proses pengenalan karakter (*character recognition*) plat nomor kendaraan lebih lanjut untuk mendukung implementasi pengawasan pelanggaran lalu lintas di jalan raya. Di samping itu, kombinasi sistem deteksi dan sistem pengenalan karakter plat nomor bisa diimplementasikan dalam pengelolaan sistem parkir secara digital dalam rangka mewujudkan manajemen sistem transportasi cerdas (*intelligent transportation system)* yang sedang digaungkan oleh Revolusi Industri 4.0.

Halaman ini sengaja dikosongkan

BAB 2
TINJAUAN PUSTAKA

Bab ini akan menjelaskan teori-teori yang berkaitan dengan sistem deteksi plat nomor kendaraan bermotor. Penjelasan ini bertujuan untuk memberikan gambaran umum tentang algoritma yang digunakan dalam pengembangan aplikasi sistem transportasi cerdas.

Secara umum, algoritma sistem deteksi plat kendaraan (*license plate detection*) terdiri dari beberapa tahap, yakni mengekstraksi *region* plat kendaraan, melakukan segmentasi karakter dari plat, dan pendeteksian akhir dari citra/objek plat nomor. Aplikasi ini melakukan praproses terlebih dahulu sebelum melakukan menentukan lokasi plat untuk mendapatkan *region of interest* (ROI), kemudian melakukan segmentasi citra menggunakan metode *sliding concentric window* (SCW), ekstraksi fitur dengan menggunakan parameter *connected component analysis* (CCA), dan pendeteksian citra/objek dari plat kendaraan.

2.1 Plat Nomor Kendaraan Bermotor

Agar kendaraan bermotor dapat dikenali identitasnya, ia diberi tanda khusus berupa plat nomor kendaraan. Plat nomor tersebut merupakan salah satu jenis identitas penting yang mudah dikenali. Bentuknya berupa potongan plat logam atau plastik yang dipasang pada kendaraan bermotor sebagai identifikasi resmi. Biasanya plat nomor jumlahnya sepasang yang ditaruh di bagian depan dan belakang kendaraan.

Keberadaan plat nomor kendaraan atau disebut juga Tanda Nomor Kendaraan Bermotor (TNKB) merupakan tanda regident kendaraan bermotor (ranmor) yang berfungsi sebagai bukti legitimasi pengoperasian ranmor berupa pelat atau berbahan lain dengan spesifikasi tertentu yang diterbitkan oleh kepolisian dan berisikan kode wilayah, nomor registrasi, serta masa berlaku dan dipasang pada kendaraan bermotor [1]. Namun ada aturan hukum tertentu atau jenis kendaraan tertentu yang hanya membutuhkan satu pelat nomor, biasanya untuk dipasang di bagian belakang. Pelat nomor memiliki nomor seri yakni susunan huruf dan angka yang dikhususkan bagi kendaraan tersebut. Di Indonesia, nomor kendaraan bermotor ini disebut nomor polisi dan biasa dipadukan dengan informasi lain mengenai kendaraan bersangkutan, seperti warna, merk,

model, tahun pembuatan, nomor identifikasi kendaraan, serta nama dan alamat pemiliknya. Semua data ini juga tertera dalam Surat Tanda Nomor Kendaraan Bermotor (STNK) yang merupakan surat bukti bahwa nomor polisi itu memang ditetapkan bagi kendaraan tersebut.

2.1.1 Manfaat Tanda Nomor Kendaraan Bermotor

Karena wujudnya yang spesifik, pelat nomor juga digunakan sebagai identifikasi kendaraan oleh banyak lembaga, seperti kepolisian, perusahaan asuransi mobil, bengkel, tempat parkir, dan juga armada kendaraan bermotor. Di beberapa wilayah yang memiliki aturan tertentu, pelat nomor juga dipakai sebagai bukti bahwa kendaraan tersebut sudah memiliki izin untuk beroperasi di jalan raya umum atau juga sebagai bukti pembayaran pajak kendaraan bermotor.

2.1.2 Tanda Nomor Kendaraan Bermotor di Indonesia

Indonesia memiliki berbagai macam plat nomor kendaraan yang sederhana. Warna dasar plat nomor di Indonesia digunakan untuk membedakan jenis kendaraan, seperti warna hitam untuk kendaraan pribadi dan sewa, warna kuning untuk kendaraan umum, warna merah yang digunakan pada kendaraan khusus instansi pemerintahan, dan lain-lain. Plat nomor yang berlaku di wilayah Indonesia memerlukan huruf pada bagian depan yang digunakan intuk membedakan area asal kendaraan tersebut, seperti N untuk wilayah di Kabupaten dan Kota Malang, L untuk Kota Surabaya, W untuk Kabupaten Sidoarjo, B untuk wilayah Jabodetabek, dan lain-lain. Secara hukum, pemilik kendaraan di Indonesia tidak dapat memesan plat nomor sesuai dengan keinginannya karena harus mengikuti peraturan penomoran plat yang berlaku dari pihak kepolisian.

Gambar 2.1 menunjukkan salah satu jenis plat nomor kendaraan bermotor yang berlaku di Indonesia.

Gambar 2.1 Plat Nomor Kendaraan Bermotor di Indonesia

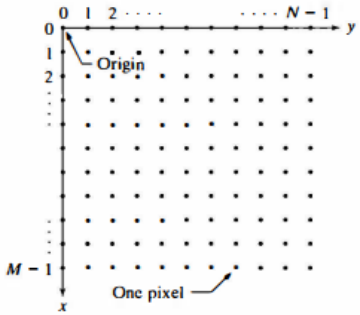
2.2 Citra Digital

Citra adalah suatu gambaran atau kemiripan dari suatu objek dengan menggunakan fungsi dari dua variabel misalnya *a(x,y)*, dimana *a* sebagai amplitudo (misalnya kecerahan) citra pada koordinat (x,y) [2][3]. Citra analog tidak dapat direpresentasikan dalam komputer sehingga tidak dapat diproses oleh komputer secara langsung. Agar bisa diproses di komputer/PC, citra analog harus dikonversi menjadi citra digital. Citra digital adalah citra yang dapat diolah oleh komputer dengan mengubah citra kontinu ke dalam bentuk diskrit, baik koordinat maupun intensitas cahayanya [3].

2.2.1 Konsep Piksel

Citra digital direpresentasikan *array* dua dimensi atau sekumpulan *array* dua dimensi dimana setiap *array* merepresentasikan satu kanal warna. Nilai kecerahan yang didigitalkan disebut sebagai nilai tingkat keabuan. Setiap elemen *array* tersebut dinamakan piksel yang diambil dari istilah '*picture element*' [4][5]. Makin banyak jumlah piksel dalam sebuah citra, makin besar resolusi spasial citra tersebut sehingga citra terlihat makin tajam. Setiap piksel mewakili tidak hanya satu titik dalam sebuah citra, melainkan sebuah bagian berupa kotak yang merupakan bagian terkecil.

Dimensi citra biasanya ditulis dengan format panjang x tinggi (misalnya 640 x 480 piksel). Namun perlu diperhatikan bahwa secara matematis maupun di dalam MATLAB, sebuah citra digital didefinisikan sebagai ukursn tinggi *M* (misalnya 480) dan panjang *N* (misalnya 640). Secara umum, sistem koordinat yang dipergunakan untuk mewakili citra dalam teori pengolahan citra dapat dilihat pada Gambar 2.2.



Gambar 2.2 Sistem Koordinat Citra Digital [6]

Sebuah citra digital diwakili oleh matriks yang terdiri dari *M* baris dan *N* kolom, dimana terdapat piksel yang merupakan perpotongan antara baris dan kolom. Piksel memiliki dua parameter, yaitu koordinat dan intensitas atau warna. Nilai yang terdapat pada koordinat *(x,y)* adalah *f(x,y)*, yaitu besarnya intensitas atau warna dari piksel di titik tersebut [5][6][7]. Definisi citra secara matematis terlihat seperti di bawah ini dimana *x* menunjukkan baris dan *y* menunjukkan kolom: [5][6][7]

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & \cdots & f(0,N-1) \\ f(1,0) & f(1,1) & \cdots & f(1,N-1) \\ \vdots & \vdots & \vdots & \vdots \\ f(M-1,0) & f(M-1,1) & \cdots & f(M-1,N-1) \end{bmatrix} \quad (2.1)$$

Seperti pada layar monitor, koordinat citra dimulai dari pojok kiri atas. Secara matematis dimulai dari *(0,0)* dan berakhir di *(M-1,N-1)* [7][8]. Namun secara implementasi di MATLAB, koordinat citra dimulai dari *(1,1)* dan berkhir di *(M,N)*. Untuk mengakses piksel citra, penulisan indeks secara matematis pada citra bersesuaian juga dengan penulisan indeks pada MATLAB sehingga *f(0,0)* pada matematis = *f(1,1)* di MATLAB.

2.2.2 Parameter Citra Digital

Citra digital memiliki beberapa parameter identitas, antara lain: [8]

1. **Kecerahan *(brightness)***
   Kecerahan *(brightness)* merupakan intensitas cahaya yang di pancarkan piksel dari citra yang dapat ditangkap oleh sistem penglihatan. Kecerahan pada sebuah titik (piksel) di dalam citra merupakan intensitas rata-rata dari suatu area yang melingkupinya.
2. **Kontras *(contrast)***
   Kontras *(contrast)* menyatakan sebaran terang gelap dalam sebuah citra. Pada citra yang baik, komposisi gelap dan terang tersebar secara merata.
3. **Kontur *(contour)***
   Kontur *(contour)* adalah keadaan yang ditimbulkan oleh perubahan intensitas pada piksel-piksel yang bertetangga.

Karena adanya perubahan intensitas inilah mata mampu mendeteksi tepi-tepi objek di dalam citra.

4. **Warna**
   Warna sebagai persepsi yang ditangkap sistem visual terhadap Panjang gelombang cahaya yang dipantulkan oleh objek.
5. **Bentuk *(shape)***
   Bentuk *(shape)* adalah properti intrinsik dari objek 3 dimensi, dengan pengertian bahwa bentuk merupakan properti intrinsik utama untuk sistem visual manusia.
6. **Tekstur *(texture)***
   Tekstur *(texture)* dicirikan sebagai distribusi spasial dari derajat keabuan di dalam sekumpulan piksel-piksel yang bertetangga. Teskstur adalah keteraturan pola-pola tertentu yang terbentuk dari susunan piksel-piksel dalam citra digital. Informasi tekstur dapat digunakan untuk membedakan sifat-sifat permukaan suatu benda dalam citra yang berhubungan dengan kasar dan halus, juga sifat-sifat spesifik dari kekasaran dan kehalusan permukaan tadi, yang sama sekali terlepas dari warna permukaan tersebut.

2.3 Citra RGB

Citra berwarna atau RGB merupakan salah satu citra yang menyajikan warna dalam bentuk komponen R (merah/*red*), G (hijau/*green*), dan B (biru/*blue*) [4][9]. Warna setiap piksel ditentukan oleh kombinasi dari intensitas warna merah, hijau, dan biru yang disimpan pada bidang warna di lokasi piksel. Format *file* grafis menyimpan citra warna sebagai citra 24 bit yang berasal dari komponen merah, hijau, dan biru yang masing-masing terdiri dari 8 bit [3]. Jika setiap komponen warna menggunakan 8 bit (nilainya antara 0 sampai dengan 255), maka terdapat 16.777.216 warna yang bisa disajikan pada citra ini [4].

Gambar 2.3 Contoh Citra RGB

2.4 Citra Keabuan (*Grayscale*)

Setiap piksel yaitu bayangan abu-abu yang memiliki intensitas 0 hingga 255, rentang intensitas ini menandakan bahwa tiap piksel dapat direpresentasikan oleh delapan bit [4]. Nilai 0 merepresentasikan warna hitam dan nilai 255 merepresentasikan warna putih, sedangkan nilai-nilai di antaranya merepresentasikan warna keabuan yang bervariasi dari hitam hingga cerah menuju putih [9]. Citra keabuan dapat diperoleh dari citra berwarna melalui transformasi dari ruang warna RGB ke ruang warna lain (HSV, HSL, Lab, YcbCr, atau HCL). Citra keabuan biasanya digunakan untuk representasi citra medis (sinar-X), tulisan/buku, dan lain-lain.

Gambar 2.4 Hasil Citra Keabuan (*Grayscale*)

Perubahan citra berwarna menjadi citra *grayscale* dapat dilakukan dengan menggunakan rumus berikut : [4]

$$I(i,j) = 0,299\ (R) + 0,587\ (G) + 0,114\ (B) \quad (2.2)$$

atau

$$I(i,j) = 0,3\ (R) + 0,6\ (G) + 0,1\ (B) \quad (2.3)$$

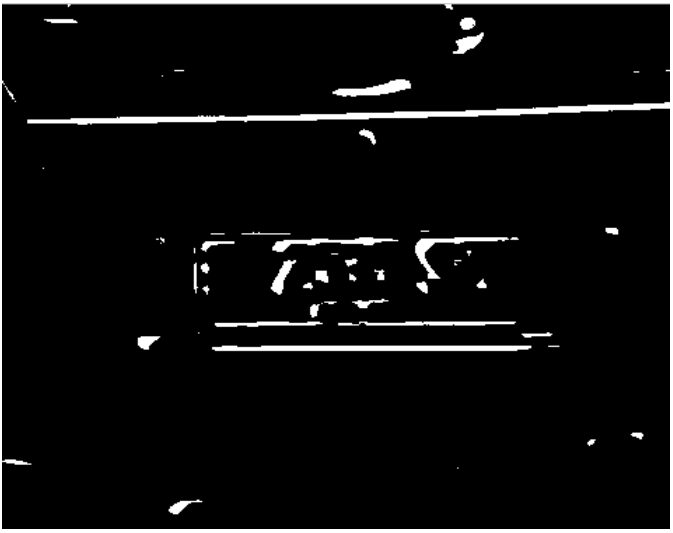
Keterangan :

I(i,j) = Nilai intensitas citra *grayscale*
R(i,j) = Nilai intensitas citra warna merah dari citra asal
G(i,j) = Nilai intensitas citra warna hijau dari citra asal
B(i,j) = Nilai intensitas citra warna biru dari citra asal

2.5 Segmentasi Citra

Segmentasi citra biasanya menjadi bagian dari proses prapengolahan citra (*image preprocessing*). Proses segmentasi dimulai dengan memproses pembagian citra digital menjadi beberapa daerah atau kelompok, dimana masing-masing daerah (wilayah/*region*) yang merepresentasikan objek terdiri dari sekumpulan piksel [8]. Segmentasi merupakan suatu bagian yang sangat penting dalam analisis citra secara otomatis, sebab pada prosedur ini objek yang diinginkan akan disadap untuk proses selanjutnya, misalnya: pada pengenalan pola.

Secara umum, algoritma segmentasi citra didasarkan pada satu dari dua properti nilai intensitas yaitu mendeteksi diskontinuitas atau mendeteksi kesamaan. Diskontinuitas memiliki pendekatan memecah atau memilih citra berdasarkan perubahan intensitas yang tiba-tiba atau cukup besar. Proses segmentasi berdasarkan mendeteksi diskontinuitas terdiri dari deteksi titik, deteksi garis, dan deteksi tepi. Sedangkan, berdasarkan pada memecah citra ke dalam wilayah yang sama menurut beberapa kriteria yang telah ditentukan, diantaranya adalah proses *thresholding*, *region growing*, dan *region splitting* dan *merging*.



Gambar 2.5 Contoh Hasil Segmentasi Citra

2.6 *Thresholding*

Thresholding ialah sebuah metode yang dapat digunakan untuk mengubah sebuah citra keabuan (*grayscale*) menjadi citra biner, yaitu citra yang hanya memiliki nilai 0 (hitam) atau 1 (putih) untuk setiap pikselnya [3]. Pada proses *thresholding*, sebuah piksel dinyatakan sebagai "objek" atau *foreground* (bernilai 1) jika memiliki nilai derajat keabuan lebih besar dari ambang batas (*threshold*) dan dinyatakan sebagai *background* (bernilai 0) jika nilai derajat keabuannya lebih kecil dari *threshold* [4]. Pada *threshold* yang tinggi, tidak dapat perbedaan karena keterbatasan mata manusia.

Gambar 2.6 Hasil Binerisasi Citra Digital

2.7 *Image Masking*

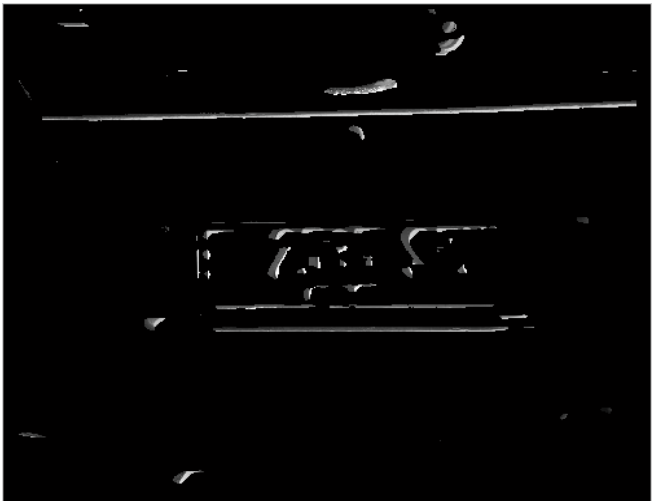
Fungsi *masking* digunakan untuk membuat *mask-image*. Citra *mask* adalah citra biner nilai piksel-pikselnya 0 dan 1. Ketika citra *mask* digunakan dalam proses, maka piksel-piksel pada citra *mask* dengan nilai 1 diproses dan 0 diabaikan. Atau dengan kata lain penggunaan fungsi *masking* dimaksudkan untuk memotong citra.

Jika fungsi *masking* diterapkan pada citra skala keabuan dengan ukuran yang sama, maka semua piksel yang 0 dalam *mask* diatur menjadi 0 pada gambar *output*. Sementara, piksel yang lain dianggap tetap atau tidak berubah. *Masking* sendiri dapat diimplementasikan, baik

menggunakan “perkalian” piksel atau logika “AND” [10]. Cara kedua (AND) umumnya proses menjadi lebih cepat. *Masking* sering digunakan untuk membatasi titik atau operator aritmatika ke area yang ditentukan oleh *mask*.

Di sisi lain, *image masking* juga dapat digambarkan sebagai proses menutupi area yang diinginkan dalam gambar *input*. Hal tersebut dapat diselesaikan dengan memprosesnya menggunakan sebuah operator, kemudian menutupi gambar *input* asli pada area yang belum diproses dengan *masking* terbalik dan akhirnya menggabungkan kembali kedua gambar parsial menggunakan proses penambahan gambar.

Contoh proses *masking* adalah penggunaan operator logika AND. Dalam beberapa paket pemrosesan gambar, *masker* dapat secara langsung didefinisikan sebagai *input* opsional untuk operator titik, sehingga secara otomatis operator hanya diterapkan pada piksel yang ditentukan oleh *masker*.



Gambar 2.7 Contoh Hasil *Masking* Citra

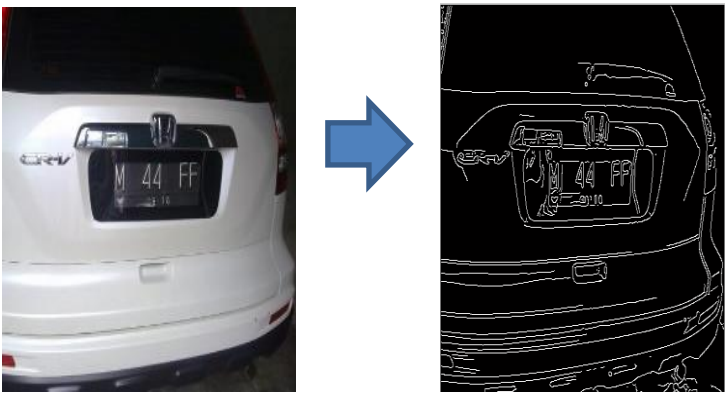
2.8 Deteksi Tepi (Metode *Canny*)

Deteksi tepi merupakan suatu teknik yang bertujuan untuk menemukan garis tepi dari suatu objek pada citra dengan cara mendeteksi perubahan tingkat kecerahan (*brightness level*) yang signifikan atau memiliki diskontinuitas. Terdapatnya diskontinuitas lokal dalam nilai

piksel yang melebihi ambang batas (*threshold*) yang diberikan disebut sebagai “tepi” (*edge*) [5][5].

Istilah “tepi” juga didefinisikan sebagai perbedaan intensitas atau warna antara satu piksel dengan piksel tetangga terdekatnya [11]. Semakin tinggi perbedaannya, maka semakin tampak jelas tepi tersebut. Pendeteksian tepi dapat dilakukan dengan menggunakan *highpass filter* karena tepi termasuk ke dalam bagian citra berfrekuensi tinggi. Pendeteksian tepi bisa digunakan untuk segmentasi citra dan ekstraksi data untuk kebutuhan pengolahan citra, *computer vision*, dan *machine vision* [4].

Salah satu algoritma deteksi tepi modern adalah deteksi tepi dengan menggunakan metode *Canny*. Metode ini ditemukan oleh Marr dan Hildreth yang meneliti pemodelan persepsi visual manusia pada tahun 1986. Pendekatan ini dapat menemukan tepi dengan ketebalan 1 piksel [12]. Langkah pertama adalah menghaluskan citra menggunakan *filter* Gaussian yang dilanjutkan dengan menerapkan *kernel Sobel*. Dari *kernel Sobel*, arah gradien di setiap titik dapat diestimasi. Selanjutnya, prinsip *non-maximum suppression* dapat diterapkan untuk menghilangkan kemungkinan gradien suatu piksel dari *edge* jika piksel tersebut bukan nilai maksimum di arah *edge* tersebut sehingga diperoleh garis tepi yang lebih baik. Namun, jika sebuah tepi memiliki besaran gradien yang terlalu kecil terhubung pada sebuah piksel dengan besaran gradien di atas ambang atas, maka tepi tersebut tidak dipangkas [4].

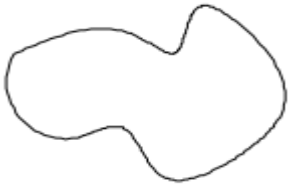


Gambar 2.8 Contoh Proses Deteksi Tepi (Metode *Canny*)

2.9 Kontur

Kontur (*contour*) didefinisikan sebagai rangkaian piksel-piksel tepi yang membentuk batas daerah (*region boundary*). Kontur (*contour*) dapat diartikan sebagai suatu keadaan yang terjadi karena adanya perubahan intensitas pada piksel-piksel yang bertetangga. Dengan adanya perubahan intensitas ini, maka titik-titik tepi (*edge*) pada citra dapat terdeteksi. Kontur juga dapat didefinisikan sebagai urutan titik yang dapat menguraikan bentuk atau *region*. [8]

Kontur dapat bersifat terbuka atau tertutup. Kontur tertutup berkoresponden dengan batas yang mengelilingi suatu daerah seperti pada Gambar 2.9. *Pixel-pixel* di dalam daerah dapat ditemukan dengan algoritma pengisian (*filling algorithm*). Batas daerah berguna untuk mendeskripsikan bentuk objek dalam tahap analisis citra (misalnya untuk mengenali objek). [8]



Gambar 2.9 Kontur tertutup

Sedangkan, kontur terbuka dapat berupa fragmen garis atau bagian dari batas daerah yang tidak membentuk sirkuit (Gambar 2.10).

Gambar 2.10 Kontur Terbuka

2.10 Metode Analisis Citra Digital

Salah satu metode yang digunakan untuk menganalisis karakteristik citra digital adalah *Connected Component Analysis* (CCA).

CCA merupakan salah satu teknik pemrosesan citra yang cukup dikenal, dimana yang dapat digunakan untuk memeriksa dan membuat piksel dari citra yang dipecah menjadi komponen berdasarkan konektivitas piksel. Maksudnya semua piksel yang berada dalam komponen yang terhubung akan berbagi nilai intensitas piksel yang sama dan dalam beberapa hal akan terhubung satu sama lain, baik secara kuartet maupun oktet. Setelah semua kelompok ditentukan, setiap piksel akan diberi label dengan nilai sesuai dengan komponen yang ditugaskan.

CCA dapat bekerja pada citra biner maupun citra keabuan dan ukuran konektivitas yang berbeda dimungkinkan untuk digunakan. Untuk tugas akhir ini, diterapkan metode CCA untuk citra biner dengan pengelompokkan tiap 8-konektivitas. Untuk mengekstraksi dan memberi label pada berbagai komponen yang terpisah dan terhubung dalam suatu citra yang merupakan pusat dari berbagai aplikasi analisis citra otomatis disebabkan banyak ukuran dan fitur yang bisa membantu dalam objek biner yang dapat diekstraksi, seperti area, orientasi, momen, rasio aspek perimeter, dan sebagainya.

2.10.1 *Aspect Ratio (AR)*

Rasio aspek (juga disebut elongasi atau eksentrisitas) dapat ditemukan dengan memindai citra serta mencari nilai minimum dan maksimum pada baris dan kolom dimana objek berada. Rasio ini didefinisikan sebagai rumus: [13][14]

$$AR = \frac{c_{MAX} - c_{MIN} + 1}{r_{MAX} - r_{MIN} + 1} \tag{2.4}$$

Dimana c menunjukkan kolom dan r menunjukkan baris.

Aspect ratio pada suatu objek tidak *invariant* (variabel yang memiliki tipe data) secara bergiliran sehingga untuk menjadi berguna sebagai ukuran komparatif objek harus diputar ke beberapa orientasi standar; seperti mengarahkan sumbu momen kedua terkecil dalam arah horisontal.

2.10.2 *Orientation*

Untuk mendapatkan persamaan umum pada area, pusat area, dan sumbu momen kedua terkecil (*axis of least second moment*), fungsi $I_i(r,c)$ perlu dirumuskan sebagai berikut [14]:

$$I_i(r,c) = \begin{cases} 1, & \text{jika } I(r,c) = \text{bilangan objek i} \\ 0, & \text{yang lainnya} \end{cases} \tag{2.5}$$

Area pada objek *I* dapat didefinisikan sebagai:

$$A_i = \sum_{r=0}^{N-1}\sum_{c=0}^{N-1} I_i(r,c) \tag{2.6}$$

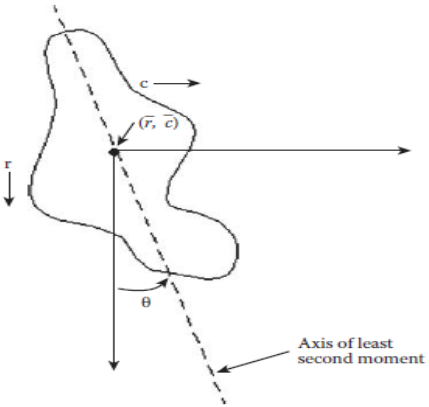
Area, A_i, diukur dalam piksel dan mengindikasikan ukuran relatif objek. Selanjutnya, suatu objek dapat mendefinisikan pusat area (*centroid* dalam kasus umum) yang menemukan titik tengah sepanjang setiap baris dan sumbu kolom yang cocok dengan 'tengah (*middle*)' berdasarkan distribusi spasial piksel dalam objek. Pusat area dapat dirumuskan oleh pasangan (r_i, c_i) sebagai berikut.

$$\bar{r}_i = \frac{1}{A_i}\sum_{r=0}^{N-1}\sum_{c=0}^{N-1} rI_i(r,c) \tag{2.7}$$

$$\bar{c}_i = \frac{1}{A_i}\sum_{r=0}^{N-1}\sum_{c=0}^{N-1} cI_i(r,c) \tag{2.8}$$

Hal tersebut sesuai dengan koordinat baris pusat area untuk objek ke-*i*, $\bar{r}_i$, dan koordinat kolom dari pusat area untuk objek ke-*i*, $\bar{c}_i$. Fitur ini akan membantu untuk menemukan objek pada bidang citra dua dimensi. Fitur selanjutnya yang perlu dipertimbangkan ialah sumbu momen kedua terkecil (*axis of least second moment*) yang memberikan informasi tentang orientasi objek. Sumbu ini sesuai dengan garis yang membutuhkan energi paling sedikit untuk memutar objek yang bentuknya serupa atau sumbu inersia terkecil. Jika objeknya memindahkan objek asal ke pusat area, (*r*,*c*), rumus dari *axis of least second moment* dapat didefinisikan sebagai berikut [13][14] :

$$\tan(2\theta_i) = 2\frac{\sum_{r=0}^{N-1}\sum_{c=0}^{N-1} rcI_i(r,c)}{\sum_{r=0}^{N-1}\sum_{c=0}^{N-1} r^2 I_i(r,c) - \sum_{r=0}^{N-1}\sum_{c=0}^{N-1} c^2 I_i(r,c)} \tag{2.9}$$



Gambar 2.11 Sumbu momen kedua terkecil [14]

Gambar 2.11 menunjukkan sumbu momen kedua terkecil, dimana sudut didefinisikan melalui *centroid* dan berlawanan arah jarum jam dengan sumbu vertikal (sumbu r).

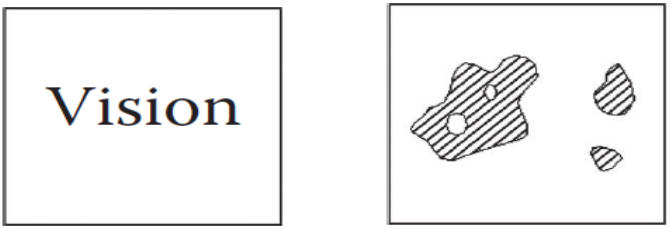
2.10.3 Bilangan Euler (*Euler Number*)

Bilangan Euler dari suatu gambar didefinisikan sebagai jumlah objek dikurangi jumlah lubang [13].

$$\boldsymbol{E = N - H} \tag{2.10}$$

Dimana ***E*** merupakan bilangan Euler, ***N*** adalah jumlah objek, dan ***H*** ialah jumlah lubang pada suatu objek/citra.

Untuk objek tunggal, akan tersedia sejumlah kurva tertutup yang berisi objek. Dalam citra biner, karakter plat menunjukkan "lubang (*hole*)" pada objek (plat nomor kendaraan).



(a) (b)

Gambar 2.12 Bilangan Euler [14]

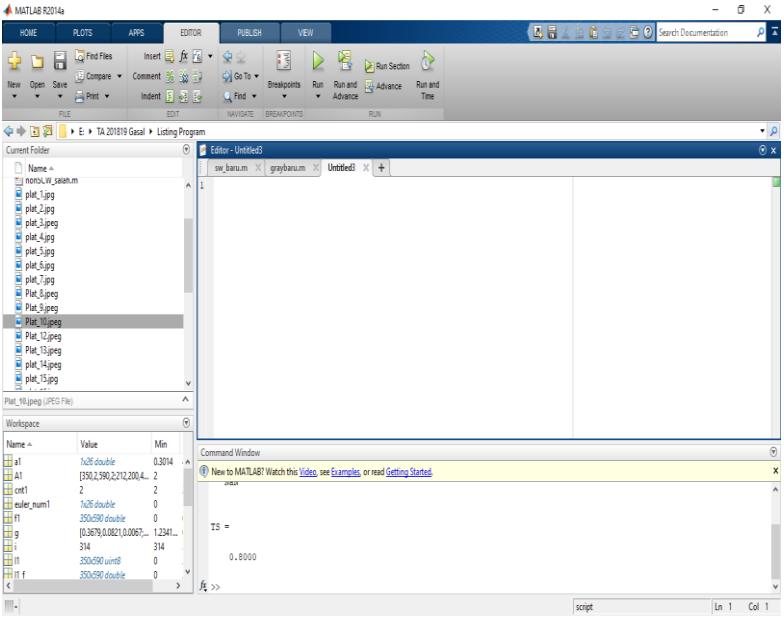
Gambar 2.12(a) menunjukkan adanya citra yang memiliki delapan objek (V = 1, i = 2, s = 1, o = 1, dan n = 1) dan satu lubang. Sedangkan, Gambar 2.12(b) menunjukkan citra yang memiliki tiga objek dan dua lubang. Suatu objek dapat ditemukan bilangan Euler untuk seluruh citra atau untuk satu objek di dalam citra. Misalnya, huruf "i" memiliki bilangan Euler bernilai 2 dan huruf "o" memiliki bilangan Euler senilai 0.

Dengan menggunakan pengukuran biner ini, jika objek memasukkan lebih dari tiga lubang di dalamnya, agar tetap terpelihara, maka bilangan Euler harus di-*set* >3. Hal tersebut dapat diterapkan pada algoritma deteksi agar memungkinkan bisa mendeteksi plat nomor yang berisi tidak kurang dari 4 karakter. [13]

2.11 Matlab

Untuk menyelesaikan tugas akhir tentang sistem deteksi plat nomor kendaraan, digunakan bahasa pemrograman MATLAB. MATLAB adalah sebuah bahasa dengan kinerja tinggi untuk komputasi masalah teknik. Matlab mengintegrasikan komputasi, visualisasi, dan pemrograman dalam suatu model yang sangat mudah untuk pakai dimana masalah-masalah dan penyelesaiannya diekspresikan dalam notasi matematika yang telah dipelajari. Penggunaan Matlab meliputi bidang–bidang, seperti:

- Matematika dan komputasi
- Pembentukan algoritma
- Akusisi data
- Pemodelan, simulasi, dan pembuatan prototipe
- Analisis data, eksplorasi, dan visualisasi
- Grafik keilmuan dan bidang rekayasa

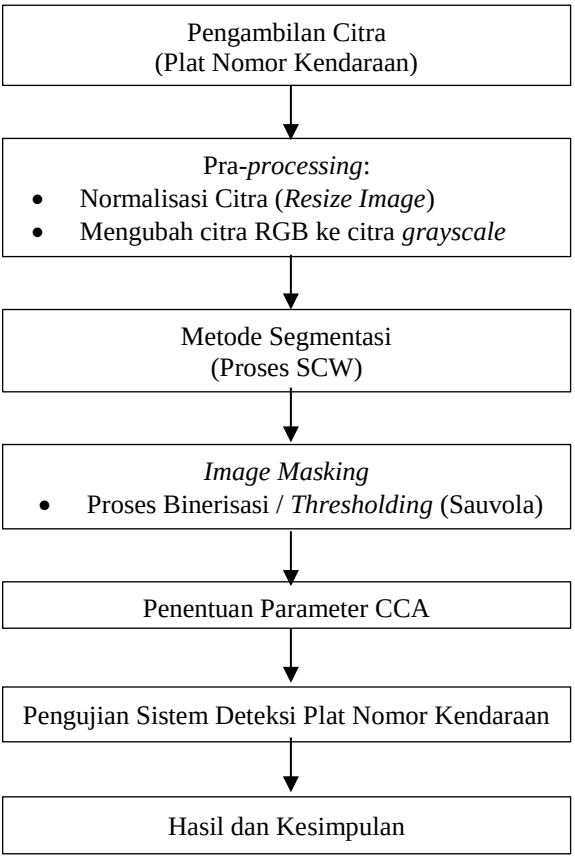


Gambar 2.13 Program MATLAB versi 2014a

BAB 3
METODE PENELITIAN

3.1 Arsitektur Umum

Untuk menyelesaikan pekerjaan deteksi plat nomor kendaraan bermotor dalam tugas akhir ini, perlu dilakukan tahapan-tahapan proses sebagai berikut:



Gambar 3.1 Diagram Blok Metode Penelitian

Pertama-tama hal yang dilakukan pengumpulan sejumlah citra digital yang berisi objek kendaraan, yang akan dijadikan obyek penelitian dalam tugas akhir ini. Jumlah citra plat nomor kendaraan yang dipilih untuk target penelitian sebanyak 20 buah. Masing-masing citra plat tersebut dievaluasi terlebih dahulu terkait ukuran piksel maupun perlu tidaknya pemotongan citra (*cropping*). Untuk ukuran piksel citra yang terlalu besar bisa dilakukan *resize* (diperkecil jumlah piksel) terlebih dahulu. Semua proses tersebut disebut sebagai langkah “normalisasi”.

Selanjutnya dilakukan *preprocessing* dengan cara mengubah citra RGB menjadi citra keabuan (*gray level*). Setelah itu dilakukan proses segmentasi menggunakan metode *Sliding Concentric Window* (SCW). SCW bekerja dengan prinsip hipotesis bahwa plat nomor kendaraan dapat dilihat sebagai suatu penyimpangan dalam tekstur gambar dan oleh karena itu perubahan secara tiba-tiba dalam karakteristik lokal suatu citra dianggap sebagai adanya plat nomor kendaraan. Berdasarkan hal di atas, makalah ini mengusulkan teknik segmentasi adaptif baru bernama *Sliding Concentric Windows* (SCW). Metode ini dikembangkan untuk menggambarkan ketidakteraturan "lokal" pada gambar menggunakan metode statistik gambar seperti standar deviasi dan nilai rata-rata.

Gambar yang dihasilkan setelah metode SCW adalah berupa citra biner, yang merupakan citra *masking* yang akan digunakan dalam operasi berikutnya. Selain itu, nilai ambang batas (*threshold*) perlu ditetapkan setelah ujicoba berkali-kali. Nilai spesifik mungkin dapat ditemukan setelah seluruh sampel citra plat nomor kendaraan selesai diidentifikasi.

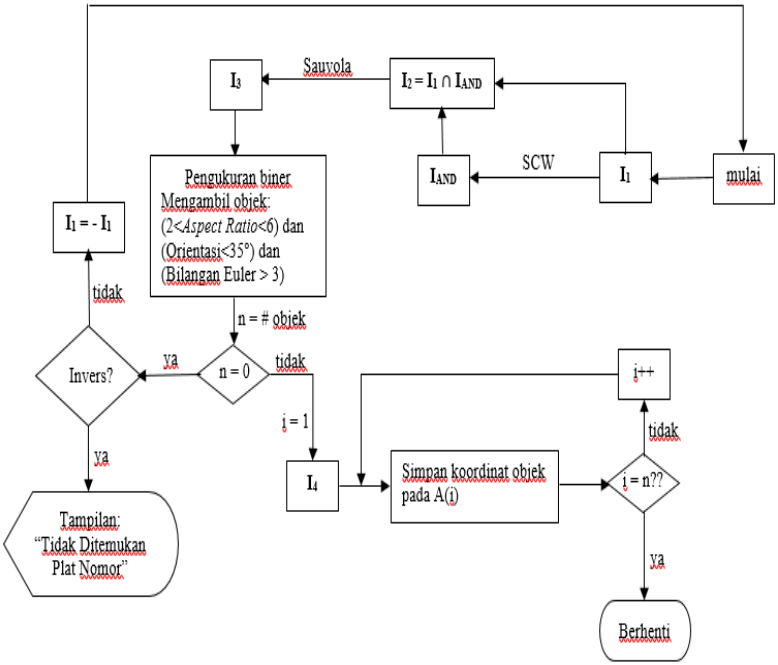
Langkah berikutnya akan dilakukan analisis menggunakan metode tertentu. Dalam tugas akhir ini dipilih metode CCA dengan menggunakan parameter ukur secara statistik berupa rasio aspek, bilangan Euler, dan sudut orientasi. Berdasarkan parameter-parameter itulah keputusan deteksi diambil apakah akan sukses sebagian ataukah sama sekali gagal.

Untuk mengevaluasi kinerja sistem deteksi yang telah dibuat, maka sejumlah rangkaian pengujian akan dilakukan dengan menggunakan *dataset* citra sebanyak 20 buah sebagaimana yang disebutkan di atas.

3.2 Rancangan Desain Sistem

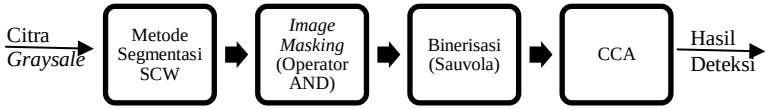
Dalam tugas akhir ini telah dibuat sistem deteksi plat nomor kendaraan dengan rancangan desain dan alur proses sebagaimana yang terlihat pada gambar 3.2. Algoritma adaptif untuk segmentasi citra diimplementasikan pada sistem tersebut, yang disebut metode *Sliding Concentric Windows* (SCW). Metode analisis CCA juga

diimplementasikan dalam sistem deteksi plat nomor ini sebagai dasar pertimbangan untuk menentukan lokasi plat nomor.



Gambar 3.2 Diagram Alur Pencarian Lokasi Plat Nomor

Sebagaimana yang terlihat pada Gambar 3.2, sistem deteksi plat nomor kendaraan pada dasarnya merupakan proses untuk mendapatkan *Region of Interest* (RoI) tertentu berupa plat nomor kendaraan dalam kondisi pencahayaan sekitar yang menjadi kendala. Pada prinsipnya, tahapan deteksi sebagaimana yang ditunjukkan pada Gambar 3.2 terdiri dari empat tahapan pokok, yaitu implementasi metode segmentasi SCW, *image masking*, binarisasi dengan metode Sauvola, dan akhirnya menghubungkan label komponen dan pengukuran biner, yang disusun secara berurutan untuk dijadikan sebagai dasar pengambilan keputusan terdeteksi atau tidaknya plat nomor yang diujicobakan. Adapun keempat tahapan tersebut bisa diilustrasikan sebagaimana pada Gambar 3.3 berikut ini.



Gambar 3.3 Tahapan Proses Deteksi Plat Nomor

3.2.1 Algoritma Sistem Deteksi Plat Nomor

Untuk merealisasikan diagram blok pada Gambar 3.3, maka perlu disusun algoritma (prosedur) pentahapan proses yang terjadi, dan selanjutnya mengkonversi algoritma tersebut ke dalam *command* (fungsi) dalam MATLAB.

Tabel 3.1 *Licensed Plate Detection Algorithm*

Input image I

1 **untuk (*for*)** setiap piksel dalam citra $\mathbf{I_1}$,
jalankan (*run*) metode SCW pada I_1
{parameter SCW: X_1, Y_1, X_2, Y_2, T dan M = nilai rata-rata; }
2 **lakukan (*perform*)** *image masking*
hitung (*calculate*) citra I_2, dimana $I_2 = I_1 \cap I_{AND}$
3 **untuk (*for*)** setiap piksel dalam gambar I_2,
{**binarisasikan $\mathbf{I_2}$** menggunakan metode Sauvola dengan parameter: k = 0,5, R = 128, b = 10}
beri nama gambar yang dihasilkan $\mathbf{I_3}$
4 **lakukan (*perform*)** pengukuran biner pada objek $\mathbf{I_3}$
ambil (*retrieve*) objek yang memenuhi pengukuran berikut:
Aspect Ratio> 2, Orientasi sudut < 35°, dan Bilangan Euler> 3;
5 **hitung (*count*)** n, di mana n adalah jumlah objek yang terdeteksi
jika (*if*) n> 0 di $\mathbf{I_3}$,
{$\mathbf{I_3} = \mathbf{I_4}$ dan lanjutkan ke langkah 6;}
jika (*if*) n = 0 di $\mathbf{I_3}$,
{balikkan $\mathbf{I_1}$ ($\mathbf{I_1} = -\mathbf{I_1}$) dan melakukan langkah 1, 2, 3, 4 dan 5 lagi;
jika (*if*) n = 0 pada gambar terbalik $\mathbf{I_1}$
{hasil : "Tidak ditemukan plat nomor"}
else {$\mathbf{I_3} = \mathbf{I_4}$ dan lanjutkan ke langkah 6;}
6. **untuk (*for*)** i = 1: n pada gambar I_4
{deteksi koordinat X_{min} X_{max} Y_{min} Y_{max} untuk objek i;}
Output: matriks A (i) di mana: A (i) = (X_{min}, X_{max}, Y_{min}, Y_{max});

3.2.2 Metode *Sliding Concentric Windows* (SCW)

Metode segmentasi yang digunakan untuk meng-*capture* objek (plat nomor kendaraan) dalam sebuah citra digital adalah *sliding concentric windows* (SCW) yang memiliki sifat cukup adaptif. Adapun teknik yang terdapat dalam SCW dapat dinyatakan dalam algoritma seperti pada Tabel 3.2. Dalam penelitian ini, hasil dari SCW dinyatakan dalam I_{AND}.

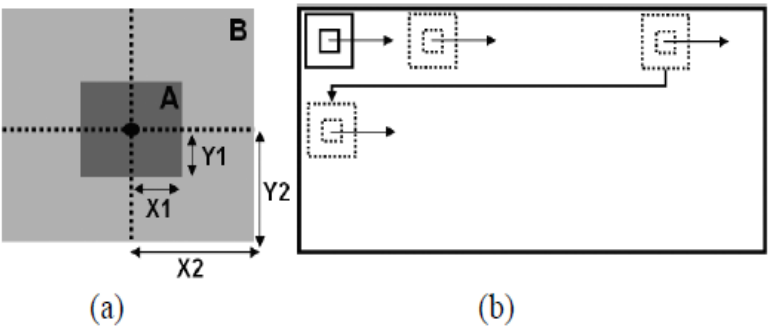
Tabel 3.2 Algoritma SCW

Input : citra keabuan

- Dua jendela konsentris A dan B dengan ukuran piksel $(2X_1)$ x $(2Y_1)$ dan $(2X_2)$ x $(2Y_2)$ masing-masing dibuat untuk piksel pertama dari citra (sudut kiri atas). Jendela tersebut ditunjukkan seperti pada Gambar 3.4 (a).

- Parameter statistik dalam windows A dan B dihitung.

Definisi aturan segmentasi: Jika rasio pengukuran statistik di dua jendela melebihi ambang batas yang ditetapkan oleh pengguna, maka piksel tengah jendela dianggap *Region of Interest* (RoI).



Gambar 3.4 (a) Jendela Konsentris dan (b) Pemindaian Citra

Jika *x* dan *y* merupakan koordinat piksel yang diuji dalam citra yang diamati (*I*), nilai piksel dalam masing-masing koordinat *x* dan *y* dari citra yang dihasilkan (I_{AND}) diset menjadi 0 (jika tidak diperoleh RoI) atau 1

(jika RoI berhasil didapatkan). Hal tersebut bisa dinyatakan dalam rumus sebagai berikut:

$$I_1(x,y) = \begin{cases} I_{AND}(x,y) = 0,\ jika \dfrac{M_B}{M_A} \le T \\ I_{AND}(x,y) = 1,\ jika \dfrac{M_B}{M_A} > T \end{cases} \quad (3.1)$$

dimana *M* adalah pengukuran statistik (nilai *mean* atau standar deviasi). Kedua *window* bergeser sampai semua citra dipindai seperti yang ditunjukkan pada Gambar 3.4 (b). Parameter X_1, X_2, Y_1, Y_2, dan *T* dapat diperbarui sesuai dengan aplikasi tertentu. Namun, setelah bereksperimen dengan metode SCW asli, menarik ditemukan bahwa jika rasio jendela konsentris mendekati rasio objek yang akan disegmentasi, maka hasil segmentasi akan menjadi optimal. Oleh karena itu, parameter X_1, X_2, Y_1, dan Y_2 dipilih berdasarkan pengamatan sebelumnya.

Dengan kata lain, tidak ada bukti yang menyatakan bahwa ada cara terbaik untuk memilih nilai untuk ambang batas tertentu (*T*) dan oleh karenanya nilai *threshold* diputuskan sesuai dengan aplikasi melalui teknik ujicoba.

3.2.3 Operator & *Thresholding Image Masking*

Citra I_2 adalah hasil dari operasi AND antara citra I_{AND} dan citra asli I_1. Untuk binarisasi citra I_2, metode *threshold* adaptif yang dipilih secara lokal akan dipilih. Teknik-teknik *thresholding* adaptif ditandai oleh perhitungan nilai ambang batas pada setiap piksel. Nilai ambang batas bergantung pada beberapa statistik lokal seperti rentang (*range*), varians, dan parameter *fitting* permukaan atau kombinasi dari logika mereka. Menjadi ciri khas pada metode adaptif lokal untuk memiliki beberapa parameter yang dapat disesuaikan. Nilai ambang batas *T(x,y)* akan ditunjukkan sebagai fungsi dari koordinat (*x,y*). Metode ini mengadaptasi ambang sesuai dengan rata-rata lokal dan standar deviasi atas ukuran jendela *b x b*.

Ambang pada piksel (x, y) dihitung sebagai:

$$T(x,y) = m(x,y) + [1 + k.(\frac{\sigma(x,y)}{R} - 1] \quad (3.2)$$

dimana dalam *m (i,j)* dan *σ (i, j)* masing-masing terkait *mean* sampel lokal dan *varians*. Sauvola menyarankan bahwa nilai *k* = 0,5, *R* = 128 dan *b* =

10 yang diadopsi dalam algoritma ini. Kontribusi standar deviasi menjadi lebih adaptif. Misalnya dalam kasus area yang penerangannya buruk, maka nilai ambangnya diturunkan. Oleh karena itu, citra I_2 menggunakan binarisasi dengan rumus berikut ini.

$$I_3(x,y)=\begin{cases}1, & \text{jika } I_2(x,y) \geq T(x,y) \\ 0, & \text{jika } I_2(x,y) < T(x,y)\end{cases} \quad (3.3)$$

3.2.4 Metode *Connected Component Analysis* (CCA)

Untuk menganalisis parameter-parameter statistik dari citra plat nomor kendaraan, dalam tugas akhir ini digunakan metode CCA. Mengekstraksi dan memberi label dari berbagai komponen yang terpisah dan terhubung dalam suatu gambar merupakan tujuan utama dalam berbagai aplikasi analisis gambar otomatis karena tersedia banyak hasil pengukuran dan fitur bermanfaat dalam objek biner hasil ekstraksi, seperti area, orientasi, momen, rasio aspek perimeter, dan lain-lain.

Mengikuti pelabelan komponen terhubung yang berhasil diperoleh dalam citra I_3 sejumlah ukuran-ukuran, seperti orientasi, rasio aspek, dan bilangan Euler untuk setiap objek biner dalam citra yang ditampilkan. Dalam tugas akhir ini, digunakan *threshold* untuk komponen CCA yaitu orientasi $<35^O$, $2<$*aspect ratio*<6 dan bilangan Euler$>$ 3. Jika hasil pengujian citra uji memenuhi kriteria dalam range tersebut, maka dianggap memiliki potensi sebagai kandidat plat nomor kendaraan yang berhasil dideteksi dan koordinatnya disimpan dalam matriks A.

Namun demikian, prosedur di atas tidak menjamin bahwa plat nomor dengan latar belakang gelap dan karakter putih akan berhasil dideteksi. Oleh karena itu, seperti yang ditunjukkan pada Gambar 3.2 atau pada langkah 5 di algoritma deteksi plat nomor (Tabel 3.1), maka jika n = 0, dimana *n* adalah jumlah objek kandidat, I_1 akan diinversi, dan langkah 1,2,3 dan 4 sekali lagi dieksekusi pada citra yang diinversi tersebut. Sedangkan jika lagi n = 0 pada langkah 5, maka tidak ada plat nomor pada citra (atau algoritma gagal menemukannya). Sebaliknya jika n>0 (setidaknya terdapat satu objek), maka $I_3 = I_4$ dan I_4 diteruskan ke langkah 6. Selanjutnya proses akan berulang sampai semua koordinat objek kandidat disimpan dalam matriks A.

3.3 Skenario Pengujian

Untuk menguji rancangan sistem deteksi plat nomor kendaraan yang dibuat, maka dilakukan pengujian untuk membuktikan kemampuan sistem tersebut dalam menjalankan tugasnya sebagaimana tujuan mengapa sistem tersebut dibuat.

Tabel 3.3 *Dataset* Citra Uji

Dataset Citra Uji			
Plat 1 500 x 423	Plat 2 500 x 500	Plat 3 420 x 305	Plat 4 380 x 383
Plat 5 508 x 284	Plat 6 500 x 377	Plat 7 482 x 488	Plat 8 500 x 359
Plat 9 500 x 399	Plat 10 500 x 400	Plat 11 510 x 382	Plat 12 500 x 350

Tabel 3.3 *Dataset* Citra Uji (Lanjutan)

Dataset Citra Uji			
Plat 13 398 x 541	Plat 14 499 x 352	Plat 15 501 x 351	Plat 16 500 x 355
Plat 17 383 x 326	Plat 18 500 x 505	Plat 19 500 x 499	Plat 20 500 x 380

Sementara, *dataset* yang digunakan untuk menguji rancangan sistem deteksi plat nomor kendaraan adalah 20 citra plat nomor dari berbagai negara dan memiliki bentuk dan karakteristik yang bervariasi sehingga bisa mewakili berbagai kriteria citra uji yang mewakili kecerahan (*brightness*), warna (*colour*), bentuk (*shape*), tekstur, kontur, dan kontras yang berbeda-beda.

3.3.1 Langkah Percobaan

Adapun pengujian sistem deteksi plat nomor kendaraan dilakukan dengan mengikuti langkah-langkah sebagai berikut.

1. Memilih plat nomor kendaraan yang dijadikan *dataset* citra uji.
2. Melakukan normalisasi citra plat nomor jika diperlukan (*resize*, *cropping*, dan sebagainya).
3. Memasukkan *dataset* ke dalam *folde*r tersendiri, dimana masing-masing plat nomor dibuat dalam *file* yang terpisah.
4. Membuka layar editor pada program MATLAB dan meng-*entry* salah satu nama *file* plat nomor kendaraan.
5. Menjalankan program (*running*).

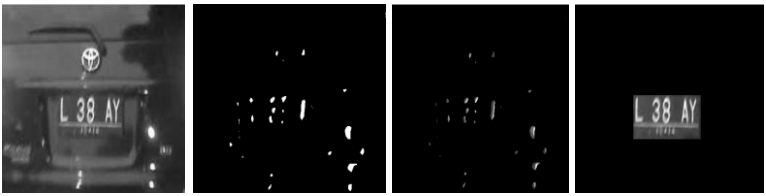
6. Menyimpan citra-citra hasil yang diperoleh tahap demi tahap.
7. Menampilkan data parameter *aspect ratio* dari citra yang diuji.
8. Menampilkan data parameter sudut orientasi dari citra yang diuji.
9. Menampilkan data parameter bilangan Euler dari citra yang diuji.
10. Mengulangi langkah 1 sampai 9 untuk *file* citra yang lain.

3.3.2 Kriteria Keterdeteksian

Untuk memberi gambaran apakah hasil pengujian termasuk dalam kriteria berhasil atau gagal, maka berikut ini akan dicontohkan beberapa kemungkinan yang terjadi terhadap hasil citra dari salah satu citra uji dalam *dataset*.

a. Kriteria Berhasil (Terdeteksi)

Gambar 3.5 menunjukkan hasil pengujian pada tiap tahapan proses deteksi plat nomor kendaraan, dimulai dari citra masukan (*grayscale*), citra hasil SCW, citra hasil *masking*, dan citra hasil deteksi plat nomor yang sukses.



Gambar 3.5 Tahapan Proses Pengujian yang Berhasil

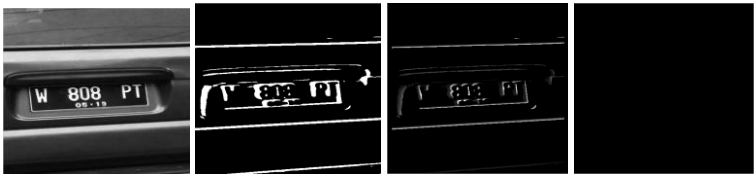
b. Kriteria Kurang Berhasil (Terdeteksi Sebagian)

Berikut ini dicontohkan salah satu proses pengujian citra plat nomor yang berhasil terdeteksi sebagian. Citra keabuan yang tampak tentunya masih kondisi normal, yang kemudian dilakukan proses SCW. Dibandingkan dengan Gambar 3.5, pada tahap *masking* di Gambar 3.6 sudah terlihat ketidakjelasam keberadaan objek. Dan hasilnya menunjukkan hanya mampu dideteksi sebagian saja dari plat nomor.

Gambar 3.6 Tahapan Proses Pengujian yang Berhasil Sebagian

c. Kriteria Tidak Berhasil (Tidak Terdeteksi)

Pada bagian ini akan dicontohkan hasil proses pengujian yang pada akhirnya tidak mampu mendeteksi keberadaan plat nomor kendaraan sama sekali. Hasil deteksi berupa citra berwarna hitam yang identik dengan isi piksel yang bernilai 0 semua.



Gambar 3.7 Tahapan Proses Pengujian yang Berhasil

Halaman ini sengaja dikosongkan

BAB 4
ANALISIS DATA DAN PEMBAHASAN

4.1 Kriteria Kedeteksian Dalam Pengujian

Sebelum membahas hasil pengujian, akan dilakukan pengelompokkan terlebih dahulu terhadap kualitas hasil deteksi plat nomor berdasarkan penglihatan visual mata pada umumnya. Adapun jenis-jenis kualitas deteksi visual plat nomor yang berhasil ditampilkan ada tiga macam, yaitu terdeteksi penuh, terdeteksi sebagian, dan tidak terdeteksi sama sekali.

Tabel 4.1 Kriteria Kedeteksian secara Visual

No.	Jenis Kedeteksian	Kriteria	Contoh	
1	Terdeteksi (terdeteksi penuh)	• Seluruh / hampir seluruh bagian plat nomor (objek) terdeteksi • Tidak terdapat karakter yang hilang (tidak terbaca)		
2	Terdeteksi sebagian	• Ada bagian plat nomor (objek) yang terpotong • Ada sebagian karakter yang hilang (tidak terdeteksi)		
3	Tidak terdeteksi	• Tidak ada bagian plat nomor (objek) yang terdeteksi • Tidak ada karakter yang terbaca • Tampilan (*screen*) berwarna hitam		

4.2 Hasil Pengujian

Setelah dilakukan pengujian deteksi plat nomor kendaraan terhadap 20 citra uji (bisa dilihat di Lampiran B), maka didapatkan dua hasil pengujian yaitu tampilan secara visual dan data-data parameter yang didapat.

4.2.1 Hasil Pengujian Visual

Dengan menguji sebanyak 20 kali pada plat nomor uji, masing-masing diproses secara terpisah, baik dalam tahapan proses *scanning* (SCW), binerisasi (Sauvola), deteksi tepi (*Canny*), proses *masking* (CCA), maupun proses deteksi akhir, maka diperoleh hasil pengujian secara visual sebagai berikut:

Tabel 4.2 Hasil Pengujian

No.	Citra Plat	Ukuran Piksel	Hasil Deteksi
1.	Plat 1	500 x 423	Terdeteksi
2.	Plat 2	500 x 500	Tidak terdeteksi
3.	Plat 3	420 x 305	Terdeteksi
4.	Plat 4	380 x 383	Tidak terdeteksi
5.	Plat 5	508 x 284	Terdeteksi
6.	Plat 6	500 x 377	Terdeteksi sebagian
7.	Plat 7	482 x 488	Tidak terdeteksi
8.	Plat 8	500 x 359	Tidak terdeteksi
9.	Plat 9	500 x 399	Terdeteksi
10.	Plat 10	500 x 400	Terdeteksi
11.	Plat 11	510 x 382	Terdeteksi
12.	Plat 12	500 x 350	Terdeteksi
13.	Plat 13	398 x 541	Terdeteksi
14.	Plat 14	499 x 352	Terdeteksi
15.	Plat 15	501 x 351	Terdeteksi
16.	Plat 16	500 x 355	Terdeteksi
17.	Plat 17	383 x 326	Terdeteksi
18.	Plat 18	500 x 505	Terdeteksi
19.	Plat 19	500 x 499	Terdeteksi
20.	Plat 20	500 x 380	Tidak Terdeteksi

4.2.2 Hasil Pengukuran Parameter Uji

Berdasarkan proses pengujian pada tahap pertama (Tabel 4.2) dengan menambahkan beberapa baris *coding*, bisa didapatkan nilai-nilai parameter-parameter pengukuran terkait dengan objek (plat nomor) yaitu *aspect ratio*, aspek orientasi, dan aspek bilangan Euler, dan hasilnya terlihat seperti pada Tabel 4.3 dan ketiga parameter tersebut sering disebut sebagai *connected components analysis* (CCA).

Tabel 4.3 Hasil Pengukuran Parameter Utama/Uji

No.	Citra Plat	*Aspect Ratio*	Orientasi(^{O})	Bilangan Euler
1.	Plat 1	4,9164	10,197	1
2.	Plat 2	5,9863	16,724	0
3.	Plat 3	3,0611	-19,6964	1
4.	Plat 4	11,1241	13,0806	6
5.	Plat 5	3,0024	-18,0916	1
6.	Plat 6	3,492	-17,353	1
7.	Plat 7	5,4848	-7,8233	1
8.	Plat 8	2,5587	-30,4713	1
9.	Plat 9	1,9352	-27,9897	1
10.	Plat 10	1,8874	-30,672	1
11.	Plat 11	7,4182	-23,6686	1
12.	Plat 12	3,1248	-0,4579	3
13.	Plat 13	4,8718	36,0989	0
14.	Plat 14	2,2692	-9,2285	1
15.	Plat 15	3,3884	-6,4302	1
16.	Plat 16	2,345	-31,4951	0
17.	Plat 17	3,661	-12,237	14
18.	Plat 18	1,7253	-29,5132	1
19.	Plat 19	1,8	0,5614	1
20.	Plat 20	3,8424	0,7186	1

Dari Tabel 4.3 terlihat bahwa terdapat tiga parameter utama yang dijadikan dasar pengambilan keputusan sebuah citra kendaraan berhasil terdeteksi plat nomornya atau tidak, Ketiga parameter tersebut meliputi aspek rasio (lebih dari 2 dan kurang dari 6), sudut orientasi (kurang dari 35°), dan bilangan Euler (lebih besar dari 3). Agar sebuah citra kendaraan bisa dipastikan terdeteksi secara visual, maka ketiga syarat tersebut harus

terpenuhi. Sedangkan jika tidak terpenuhi salah satu untuk ketiganya, maka ada potensi plat nomor kendaraan bisa berhasil terdeteksi atau tidak.

4.3 Analisis dan Pembahasan

Berdasarkan data-data hasil pengujian yang sudah ditampilkan sebelumnya, maka bisa dilakukan analisis hubungan antara nilai-nilai parameter CCA (*Connected Component Analysis*) yang terdiri dari aspek rasio, orientasi, dan bilangan Euler terhadap sukses atau tidaknya deteksi plat nomor kendaraan. Untuk itu, perlu dilakukan pengamatan terhadap relasi antara batasan nilai parameter CCA dengan keberhasilan deteksi, dan hasilnya bisa dilihat dalam Tabel 4.4 berikut ini.

Tabel 4.4 Pengaruh Nilai CCA terhadap Keberhasilan Deteksi

No.	Citra Plat	*Aspect Ratio*	Orientasi (^{O})	Bilangan Euler	Hasil Deteksi
1.	Plat 1	4,9164	10,197	1	Terdeteksi
2.	Plat 2	5,9863	16,724	0	Tidak terdeteksi
3.	Plat 3	3,0611	-19,6964	1	Terdeteksi
4.	Plat 4	11,1241	13,0806	6	Tidak terdeteksi
5.	Plat 5	3,0024	-18,0916	1	Terdeteksi
6.	Plat 6	3,492	-17,353	1	Terdeteksi sebagian
7.	Plat 7	5,4848	-7,8233	1	Tidak terdeteksi
8.	Plat 8	2,5587	-30,4713	1	Tidak terdeteksi
9.	Plat 9	1,9352	-27,9897	1	Terdeteksi
10.	Plat 10	1,8874	-30,672	1	Terdeteksi
11.	Plat 11	7,4182	-23,6686	1	Terdeteksi
12.	Plat 12	3,1248	-0,4579	3	Terdeteksi
13.	Plat 13	4,8718	36,0989	0	Terdeteksi
14.	Plat 14	2,2692	-9,2285	1	Terdeteksi
15.	Plat 15	3,3884	-6,4302	1	Terdeteksi
16.	Plat 16	2,345	-31,4951	0	Terdeteksi
17.	Plat 17	3,661	-12,237	14	Terdeteksi
18.	Plat 18	1,7253	-29,5132	1	Terdeteksi
19.	Plat 19	1,8	0,5614	1	Tidak terdeteksi
20.	Plat 20	3,8424	0,7186	1	Terdeteksi

Dari aspek nilai rasio yang mensyaratkan >2 dan <6, maka secara mayoritas hasil pengujian telah memenuhi syarat kedeteksian, kecuali hasil percobaan ke-9 sebesar 1,9352 (<2), ke-10 bernilai 1,8874 (<2), ke-11 dengan nilai 7,4182 (>6), dan ke-18 senilai 1,7253 (<2), yang keempat-empatnya di luar *range* 2-6, namun plat tersebut masih tetap bisa terdeteksi. Hal ini masih bisa ditoleransi karena meskipun keempat-empatnya di luar *range* (2-6), namun masih mendekati nilai limit. Sedangkan kasus yang berbeda justru tampak pada hasil percobaan ke-6 dengan nilai yang sebenarnya normal (3,492), namun hanya terdeteksi sebagian (karakter huruf pinggirnya tidak terdeteksi). Kasus terakhir ini bisa dijelaskan penyebabnya karena warna mobil (*background*)-nya putih yang sama dengan warna karakter pinggir sehingga karakter pinggirnya menarik ke dalam klaster *body* mobil sehingga oleh sistem tidak dianggap bagian dari plat nomor kendaraan. Sementara pada kasus plat nomor ke-4, nilai rasio jauh di atas *range* (2-6) yaitu sebesar 11,1241 dikarenakan warna mobil (*background*) sama dengan warna *background* plat nomor sehingga sistem deteksi menganggapnya *body* mobil termasuk bagian dari plat nomor.

Sementara dari sisi nilai orientasi yang mensyaratkan $<35^{O}$, semua hasil pengukuran telah memenuhi syarat tersebut. Jika ternyata ada plat nomor yang tidak terdeteksi, bukan disebabkan karena aspek orientasi tersebut (plat nomor 4, 7, dan 19). Hanya ada satu hasil pengujian yang tampak aneh yaitu pada hasil percobaan ke-13 dimana dengan nilai orientasi sebesar $36{,}0989^{O}$ ($>35^{O}$), namun hasilnya terdeteksi dikarenakan selisihnya hanya $1{,}0989^{O}$ saja. Sementara untuk plat nomor ke-2, meskipun memiliki orientasi sebesar $16{,}724^{O}$ namun tetap tidak terdeteksi, hal tersebut dimungkinkan karena posisi plat nomor kendaraan yang tidak tegak lurus (miring).

Adapun untuk bilangan Euler yang mengakomodasi hingga >3 (4 atau lebih), maka terbukti dengan nilai bilangan Euler yang sebesar 14 (percobaan ke-17) masih bisa terdeteksi plat nomornya. Angka tersebut bisa terhitung bukan karena jumlah kelompok karakter di dalam plat nomor, melainkan latar belakang (*background*) citra yang mengandung banyak kontras hasil pantulan cahaya ditambah *body* mobil yang berwarna *silver* sehingga terdeteksi banyak *spot* yang warna keputihan. Kasus sebaliknya terjadi pada plat nomor ke-4, dengan nilai bilangan Euler sebesar 6, namun plat nomornya tidak berhasil terdeteksi. Hal ini bisa dijelaskan bahwa bilangan Euler sebesar 6 disebabkan karena banyak *spot-spot* pantulan cahaya yang berwarna keputihan sehingga bisa

terhitung nilai Euler-nya. Namun, tetap saja tidak terdeteksi disebabkan karena *background body* mobil yang berwarna hitam sehingga sama dengan *background* plat nomornya. Akibatnya, lokasi plat nomor tidak terdeteksi karena batas tepinya tidak jelas.

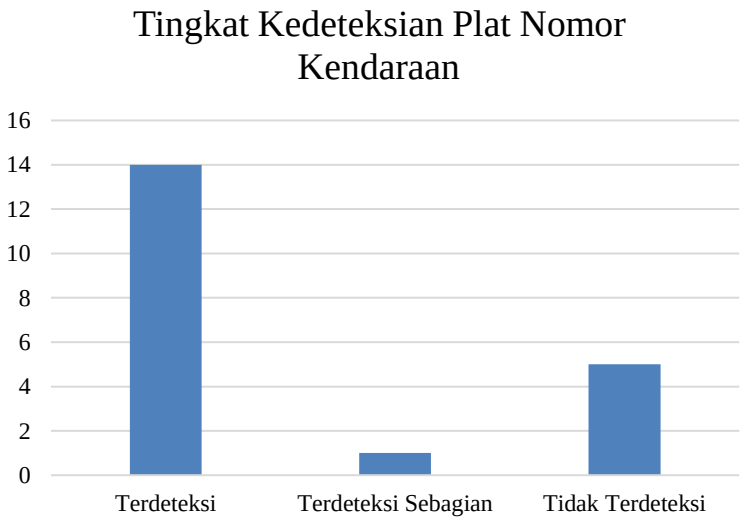
Sementara bila ukuran kelompok karakter di dalam plat nomor memiliki ukuran yang tidak jauh berbeda, maka jumlah bilangan Euler baru dapat dihitung. Sedangkan untuk perbedaan ukuran karakter yang terlalu kontras (terlalu kecil atau terlalu besar), maka seringkali diabaikan sehingga hanya dihitung 1. Sedangkan, nilai bilangan Euler sebesar 0 dapat terjadi karena terlalu banyak ukuran karakter yang mirip sehingga sulit dihitung. Hal lain yang bisa menimbulkan nilai Euler sebesar 0 karena kondisi plat nomor yang terlalu gelap dan susah terbaca jenis karakternya (plat nomor 2, 13, dan 16).

Dari keseluruhan pengujian yang telah dilakukan, hampir semua nilai parameter CCA yang diperoleh bisa digunakan untuk memprediksi kesuksesan deteksi. Namun demikian, tidak selalu menjadi jaminan bahwa bila ketiga parameter terpenuhi, maka otomatis plat nomor kendaraannya bisa berhasil dideteksi. Hal ini bisa ditunjukkan pada hasil percobaan ke-2, ke-7, dan ke-8, dimana nilai *aspect ratio*-nya berturut-turut sebesar 5,9863, 5,4848, dan 2,5587, sudut orientasi masing-masing senilai $16{,}724^{O}$, $-7{,}8233^{O}$, dan $-30{,}4713^{O}$, serta bilangan Euler-nya bernilai 0 (untuk hasil percobaan ke-2) dan bernilai 1 (untuk hasil percobaan ke-7 dan 8), namun tidak berhasil dilakukan deteksi plat nomor. Hal ini sebenarnya bisa dijelaskan bukan karena faktor parameter CCA-nya, melainkan lebih disebabkan karena posisi plat nomor kendaraan yang miring (plat nomor ke-2), sementara plat nomor ke-7 dan ke-8 disebabkan karena warna *body* mobil terlalu keputihan dan pantulan cahaya.

Dengan keberhasilan deteksi sebesar 14 dari 20 citra uji, maka sistem deteksi yang dibuat memiliki kinerja keberhasilan sebesar 70%. Dengan demikian, dapat disimpulkan parameter CCA cukup baik dipakai dalam rancangan sistem deteksi plat nomor, namun tidak cocok untuk diterapkan pada citra plat nomor yang kondisinya miring atau tidak tegak lurus. Kelemahan yang lain, parameter CCA juga kurang bisa diandalkan untuk mendeteksi plat nomor kendaraan dengan warna *body* kendaraan yang sama atau mendekati warna *background* plat nomor dan/atau warna karakter dalam plat nomor.

4.4 Perbandingan Hasil Pengujian

Dari hasil pengujian dan pembahasan yang sudah dilakukan sebelumnya, maka dapat dikelompokkan tingkat (*level)* kedeteksian plat nomor kendaraan yang diuji menjadi tiga macam sebagaimana grafik pada Gambar 4.1 berikut.

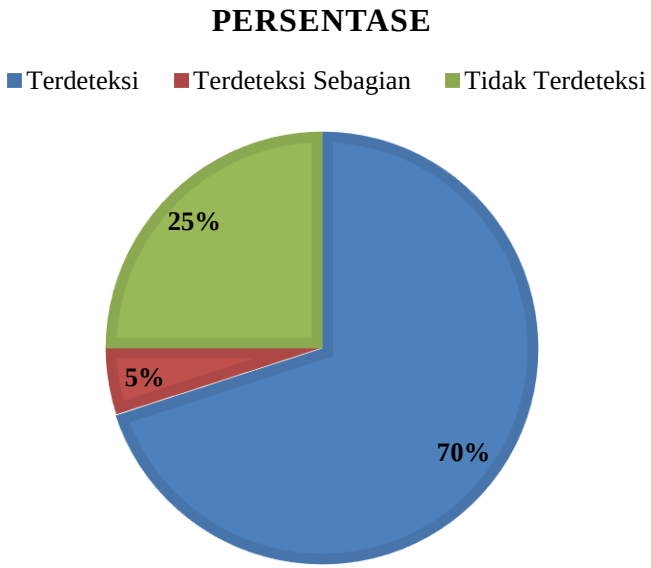


Gambar 4.1 Tingkat Kedeteksian Plat Nomor

Sebagaimana yang ditunjukkan pada Gambar 4.1 di atas, maka dari 20 kali pengujian plat nomor kendaraan yang telah dilakukan, diperoleh jumlah keberhasilan deteksi sebanyak 14 buah. Sisanya terdeteksi sebagian sebesar 1 buah dan tidak terdeteksi sama sekali sebanyak 5 buah.

Jika ditelisik lebih lanjut, penyebab plat nomor hanya terdeteksi sebagian dikarenakan warna mobil (*background*)-nya putih yang sama dengan warna karakter pinggir sehingga karakter pinggirnya menarik ke dalam klaster *body* mobil. Sedangkan untuk kasus tidak terdeteksi sama sekali, objek (plat nomor) lebih disebabkan karena warna mobil (*background*) sama dengan warna *background* plat nomor sehingga sistem deteksi menganggapnya *body* mobil termasuk bagian dari plat nomor serta kondisi plat nomor yang tidak tegak lurus (miring).

Untuk memberi gambaran tingkat keberhasilan deteksi plat nomor kendaraan secara keseluruhan, maka dapat disimpulkan bahwa dengan menggunakan parameter rasio, orientasi, dan bilangan Euler sebagai dasar pengambilan keputusan deteksi (*Connected Component Analysis*), 70% pengujian dikatakan berhasil terdeteksi, 5% terdeteksi sebagian, dan 25% sisanya tidak terdeteksi sama sekali.



Gambar 4.2 Persentase Keberhasilan Pengujian Plat Nomor

BAB 5
PENUTUP

5.1 Kesimpulan

Setelah dilakukan pengkajian, pengujian, dan pembahasan hasil yang diperoleh terkait dengan judul tugas akhir "*Deteksi Plat Nomor Kendaraan Bermotor Menggunakan Metode Sliding Concentric Window (SCW) untuk Aplikasi Sistem Transportasi Cerdas*", maka dapat diambil kesimpulan sebagai berikut.

a. Salah satu metode yang dapat digunakan untuk mendeteksi plat nomor kendaraan adalah menggunakan teknik *Sliding Concentric Window (SCW)* yang men-*scan* seluruh piksel dalam gambar untuk menemukan lokasi plat nomor.
b. Tahapan yang dilakukan untuk proses deteksi plat nomor dalam tugas akhir ini meliputi *pra-processing* (konversi ke citra keabuan), deteksi tepi (metode *Canny*), ekstraksi fitur citra (proses SCW), *image masking* (penghitungan rasio, orientasi, bilangan Euler), dan proses pengambilan keputusan deteksi (menggunakan relasi parameter CCA).
c. Dari 20 citra yang diujikan, berhasil terdeteksi sebanyak 14 buah (70%), terdeteksi sebagian sebesar 1 buah (5%), dan tidak terdeteksi sama sekali sebanyak 5 buah (25%).
d. Faktor utama penyebab tidak berhasilnya proses deteksi adalah kondisi plat nomor yang buram atau tidak jelas, kondisi latar belakang (*background*) plat nomor yang terlalu terang karena faktor cahaya yang terlalu kuat dan warna mobil yang dominan terang (warna putih atau *silver*).
e. Anomali (pengecualian) penggunaan metode SCW terjadi pada kondisi plat nomor kendaraan yang miring (tidak berhasil dideteksi), meskipun nilai CCA-nya telah memenuhi syarat.

5.2 Saran

Penggunaan metode SCW dikombinasi dengan teknik analisis CCA dalam proses deteksi plat nomor kendaraan bermotor ternyata masih terjadi kelemahan sehingga untuk kepeluan penelitian berikutnya perlu dikembangkan beberapa hal, antara lain:

a. Pemilihan nilai *threshold* (rasio orientasi dan bilangan Euler) yang lebih presisi agar tingkat keberhasilan lebih besar lagi.
b. Penggunaan metode lain yang lebih mampu mengatasi orientasi dan kemiringan plat nomor kendaraan.

Halaman ini sengaja dikosongkan

DAFTAR PUSTAKA

[1] Kepolisian Republik Indonesia, "Peraturan Kepala Kepolisian Negara Republik Indonesia Nomor 5 Tahun 2012 Tentang Registrasi dan Identifikasi Kendaraan Bermotor", Jakarta, 2012.

[2] Young, Ian T., Gerbrands, Jan. J., dan Lucas J. van Vliet, "*Fundamentals of Image Processing"*, Delft University of Technology, Belanda, 1998.

[3] Andono, Pulung Nurtantio, T. Sutojo, Muljono, "*Pengolahan Citra Digital*", Penerbit ANDI, Yogyakarta, 2017.

[4] Hidayatullah, Priyanto, "*Pengolahan Citra Digital : Teori dan Aplikasi Nyata*", Penerbit Informatika, Bandung, 2017.

[5] McAndrew, Alasdair, *"An Introduction to Digital Image Processing with Matlab : Notes for SCM2511 Image Processing 1"*, School of Computer Science and Mathematics, Victoria University of Technology, Australia, 2004.

[6] Gonzalez, Rafael C., Woods, Richard E., Eddins, Steven L., "*Digital Image Processing Using Matlab®*", edisi ke-2, Gatesmark Publishing, Amerika Serikat, 2009.

[7] Petrou, Maria, Petrou, Costas, *"Image Processing : The Fundamentals"*, edisi ke-2, Wiley, Inggris, 2010.

[8] Munir, Rinaldi, "*Pengolahan Citra Digital dengan Pendekatan Algoritma"*, Informatika, Bandung, 2004.

[9] Li, Ze-Nian, Drew, Mark S., Liu, Jiangchuan, "*Fundamentals of Multimedia*", edisi ke-2, Springer, 2014.

[10] Marques, Oge, *Practical Image and Video Processing Using Matlab®*, John Wiley & Sons Inc., New-Jersey, 2011.

[11] Madenda, Sarifuddin, "*Pengolahan Citra & Video Digital : Teori, Aplikasi, dan Pemrograman Menggunakan MATLAB*", Penerbit Erlangga, Jakarta, 2015.

[12] T. B. Moeslund, *Introduction to Video and Image Processing*, Springer, London, 2012.

[13] Anagnostopoulos, Christos Nikolaos E., dkk, *"A License Plate-Recognition Algorithm for Intelligent Transportation System Apllication"*, IEEE Transactions on Intelligent Transportation Systems, Vol. 7, No. 3. 2006.

[14] S. E. Umbaugh, *Computer Vision and Image Processing*, Prentice Hall, New Jersey, 1998, pp. 133-138.

[15] Anonim, "License Plate Extraction, Processing, and Character Segmentation" <URL : https://lebgeeks.com/forums/viewtopic.php?id=7786>, November 2010.

Halaman ini sengaja dikosongkan

LAMPIRAN A
LISTING PROGRAM [15]

```
clc
clear all
close all

% Proses konversi dari citra RGB ke citra grayscale
image = rgb2gray(imread('plat_63.jpg'));
figure;
imshow(image)
title('Citra Grayscale')

%%
% Proses Segmentasi SCW (Sliding Concentric Windwow)

%image = [1 2 3 4 5 6;7 8 9 10 11 12];
imageWidth  = size(image, 2)
imageHeight = size(image, 1)

windowWidth = 24;
windowHeight =8;

disp('Processing window1...');

for j = 1:imageHeight - windowHeight + 1
   for i = 1:imageWidth - windowWidth + 1

window1(j,i) = round((mean(mean(image(j:j + windowHeight - 1, i:i +
windowWidth - 1, :)))));
      % lakukan hal-hal dengan subcitra
   end
end

imageWidth1  = size(window1, 2)
imageHeight1 = size(window1, 1)

windowWidth1 = 12;
windowHeight1 =4;

disp('Processing window2..');
```

```
for j = 1:imageHeight - windowHeight1 + 1
   for i = 1:imageWidth - windowWidth1 + 1

window2(j,i) =round(mean(mean(image(j:j + windowHeight1 - 1, i:i +
windowWidth1 - 1, :))));
       % lakukan hal-hal dengan subcitra
   end
end


imageWidth2 = size(window2, 2)
imageHeight2 = size(window2, 1)

newwindow2 = window2(1:imageHeight1,1:imageWidth1);


for j = 1 :imageHeight1
  for i = 1 :imageWidth1

      if((window1(j,i)/newwindow2(j,i))>1.43)%1.25)
         newimage(j,i)=1;
      else
         newimage(j,i)=0;
      end
  end
end
figure;
imshow(newimage)
title('Hasil SCW pada citra (IAND)')

[M1,N1] = size(image);
[M1_new,N1_new] = size(newimage);
% Memperluas matriks newimage pada batas yang hilang; nilai masukan
pada boarder yang ditambahkan
% adalah nol.
diff = size(image) - size(newimage);
newimage = horzcat(zeros(M1_new,(diff(2)-
1)/2+1),newimage,zeros(M1_new,(diff(2)-1)/2));
[M1_new,N1_new] = size(newimage);
```

```
newimage = vertcat(zeros((diff(1)-
1)/2+1,N1_new),newimage,zeros((diff(1)-1)/2,N1_new));

imageAND = zeros(M1,N1); % Inisialisasi Matriks

%%
% Proses Image Masking

% Hasil segmentasi SCW kemudian digunakan untuk mengidentifikasi
daerah mana dari citra masukan yang dapat diterima (menggunakan
logika AND)

for i = 1 :M1
  for j = 1 : N1
     if(newimage(i,j))
        imageAND(i,j)= image(i,j);
     else
        imageAND(i,j)= 0;
     end
  end
end

figure;
imshow(imageAND,[])
title('Image Masking (I2)')

%%
% 3. Binerisasi Sauvola

windowWidth = 10;
windowHeight = 10;
[M1,N1] = size(imageAND);
% Perluas batas yang terjadi secara merata di sepanjang sisi:
% 5 kolom ke kiri dan 5 kolom ke kanan, 5 baris ke atas dan 5 baris ke
% bawah: citra selalu dalam posisi terpusat

imageAND_ext =
horzcat(imageAND(:,1:windowWidth/2),imageAND,imageAND(:,N1-
(windowWidth/2-1):N1));
```

```
imageAND_ext =
vertcat(imageAND_ext(1:windowHeight/2,:),imageAND_ext,imageAN
D_ext(M1-(windowHeight/2-1):M1,:));
[M1_ext,N1_ext] = size(imageAND_ext);
imageAND_ext_t = imageAND_ext';

% Kode diproses lebih dari ukuran asli citra =>
% i dan j terbatas dalam kisaran tertentu
for j = windowHeight/2+1:N1_ext-windowHeight/2+1
   for i = windowWidth/2+1:M1_ext-windowWidth/2+1
      % Nama window dimisalkan b untuk binerisasi Sauvola
      % Window diekstraksi dari hasil masking citra (image masking),
      % tidak yang asli; juga, jendela (window) pemindaian dipusatkan
      % sehingga mencakup seluruh citra
      b = imageAND_ext_t(j-windowHeight/2:j + windowHeight/2 - 1,i-
windowWidth/2:i + windowWidth/2 - 1);
      std2 = std(std(b));
      mA = mean(mean(b));
      K = .5;
      R = 128;
      T = mA+(1+K*((std2/R)-1));
      % Ambang batas dilakukan untuk hasil citra pada proses masking,
      % bukan asli
      if (imageAND_ext_t(j,i) >= T)
         NEW_ext(j,i) = 1;
      else
         NEW_ext(j,i) = 0;
      end
   end
end

% Hapus batas yang diperluas dan putar gambar untuk tampilan yang
benar
NEW_ext2 = NEW_ext';
NEW = NEW_ext2(windowWidth/2+1:M1_ext-
windowWidth/2+1,windowHeight/2+1:N1_ext-windowHeight/2+1);

figure;
imshow(NEW,[])
title('Hasil citra setelah binerisasi Sauvola (I3)')
```

```
%%
% Deteksi Aspect Ratio pada Plat Nomor Kendaraan

% clear all
% clc

ob_cnt = 4;

% Menghasilkan Gaussian mask yang akan digunakan untuk
mengaburkan citra ambang batas (Thresholded image) sebelum
melakukan pelabelan objek.

% Setelah menghasilkan beberapa Gaussian masks, hasil terbuaik
diperoleh untuk sig = 1
sig = 1;
n = 3*sig;
for i=1:n
   for j=1:n
     g(i,j) = exp(-(i^2+j^2)/2*sig^2);
   end
end

figure(ob_cnt+1)
I1 = imread('plat_63.jpg');
I1 = rgb2gray(I1);
imshow(I1,[]);
[M1,N1] = size(I1);

figure(ob_cnt+2)
I_bin1 = edge(I1,'canny');   % Deteksi tepi dengan Metode Canny
imshow(I_bin1,[])

figure(ob_cnt+3)
f1 = conv2(double(I_bin1),double(g));
f1 = f1(1:M1,1:N1);
[W1,num1] = bwlabel(f1,8);
imshow(W1,[])

figure(ob_cnt+4)
% Mengekstrak plat nomor kendaraan dari bagian tengah yang
diharapkan
```

```
% (Menggunakan algoritma pemotongan yang tidak ditampilkan di sini)
imshow(W1(640:480,640:480),[])

figure(ob_cnt+5)

% Label Plat Nomor Kendaraan

L1 = zeros(M1,N1);
[r1,c1,v1] = find(W1==342); % Hasil akhir
label1 = vertcat(r1',c1');
for i = 1:length(label1)
   L1(label1(1,i),label1(2,i)) = W1(label1(1,i),label1(2,i));
end
imshow(L1,[])

r1_max = max(label1(1,:));
r1_min = min(label1(1,:));
c1_max = max(label1(2,:));
c1_min = min(label1(2,:));
a1 = (c1_max - c1_min + 1)/(r1_max - r1_min + 1)

%-----------------------------------------------------------------------------

% Deteksi Sudut Orientasi pada Plat Nomor Kendaraan


% clear all
% clc

ob_cnt = 9;


sig = 1;
n = 3*sig;
for i=1:n
   for j=1:n
     g(i,j) = exp(-(i^2+j^2)/2*sig^2);
   end
end

figure(ob_cnt+1)
```

```
I1 = imread('plat_63.jpg');
I1 = rgb2gray(I1);
imshow(I1,[]);
[M1,N1] = size(I1);

figure(ob_cnt+2)
I_bin1 = edge(I1,'canny');
imshow(I_bin1,[]);

figure(ob_cnt+3)
f1 = conv2(double(I_bin1),double(g));
f1 = f1(1:M1,1:N1);
[W1,num1] = bwlabel(f1,8);
imshow(W1,[])

figure(ob_cnt+4)
L1 = zeros(M1,N1);
[r1,c1,v1] = find(W1==342);
label1 = vertcat(r1',c1');
for i = 1:length(label1)
   L1(label1(1,i),label1(2,i)) = W1(label1(1,i),label1(2,i));
end
imshow(L1,[])

for i = 1:M1
   for j = 1:N1
      rc_component1(i,j) = i*j*L1(i,j);
   end
end
for i = 1:M1
   for j = 1:N1
      r2_component1(i,j) = i^2*L1(i,j);
   end
end
for i = 1:M1
   for j = 1:N1
      c2_component1(i,j) = j^2*L1(i,j);
   end
end
phi1 = atand((2*sum(sum(rc_component1)))/...
         (sum(sum(r2_component1))-sum(sum(c2_component1))))
```

```
%%
%Identifikasi Parameter Fase CCA

figure(ob_cnt+6)
cnt1 = 0;
for k = 1:num1              % Hitung aspect ratio
   L1 = zeros(M1,N1);
   [r1,c1,v1] = find(W1==k);
   label1 = vertcat(r1',c1');
   for i = 1:size(label1,2)
      L1(label1(1,i),label1(2,i)) = W1(label1(1,i),label1(2,i));
   end
   r1_max = max(label1(1,:));
   r1_min = min(label1(1,:));
   c1_max = max(label1(2,:));
   c1_min = min(label1(2,:));
   a1(k) = (c1_max - c1_min + 1)/(r1_max - r1_min + 1);
   clear L1 r1 c1 v1 label1
end
for k = 1:num1              % Menghitung sudut orientasi
   L1 = zeros(M1,N1);
   [r1,c1,v1] = find(W1==k);
   label1 = vertcat(r1',c1');
   for i = 1:size(label1,2)
      L1(label1(1,i),label1(2,i)) = W1(label1(1,i),label1(2,i));
      rc_component1(label1(1,i),label1(2,i)) = label1(1,i)*label1(2,i)...
                                 *L1(label1(1,i),label1(2,i));
      r2_component1(label1(1,i),label1(2,i)) = label1(1,i)^2*...
                                  L1(label1(1,i),label1(2,i));
      c2_component1(label1(1,i),label1(2,i)) = label1(2,i)^2*...
                                  L1(label1(1,i),label1(2,i));
   end
   phi1(k) = atand((2*sum(sum(rc_component1)))/...
              (sum(sum(r2_component1))-sum(sum(c2_component1))));
   clear L1 r1 c1 v1 label1 rc_component1 r2_component1
c2_component1
end
for k = 1:num1              % Menghitung bilangan Euler
   L1 = zeros(M1,N1);
   [r1,c1,v1] = find(W1==k);
```

```
    label1 = vertcat(r1',c1');
    for i = 1:size(label1,2)
        L1(label1(1,i),label1(2,i)) = W1(label1(1,i),label1(2,i));
    end
    L1_ed = edge(L1,'canny');
    euler_num1(k) = bweuler(L1_ed,8);
    clear L1 r1 c1 v1 label1 L1_ed
end

%%
%Ekstraksi Plat Nomor Kendaraan dalam citra

I1_ob = zeros(M1,N1);
I1_f = zeros(M1,N1);
for k = 1:num1              % Pengambilan objek dengan spesifikasi aspect
ratio,
                            % sudut orientasi, and bilangan Euler
    if ((a1(k) > 1.5)&&(a1(k) < 5.5))&&...
       (((phi1(k)+180) > 105)&&((phi1(k)+180) < 140))&&...
       (euler_num1(k) >= 3)
        L1 = zeros(M1,N1);
        [r1,c1,v1] = find(W1==k);
        label1 = vertcat(r1',c1');
        for i = 1:size(label1,2)
            L1(label1(1,i),label1(2,i)) = W1(label1(1,i),label1(2,i));
        end
        for i = 1:length(label1)
            I1_ob(label1(1,i),label1(2,i)) = L1(label1(1,i),label1(2,i));
        end
        r1_max = max(label1(1,:));
        r1_min = min(label1(1,:));
        c1_max = max(label1(2,:));
        c1_min = min(label1(2,:));
        A1(cnt1+1,:) = [r1_max,r1_min,c1_max,c1_min];
        I1_f(r1_min:r1_max,c1_min:c1_max) = ...
                                I1(r1_min:r1_max,c1_min:c1_max);
        cnt1 = cnt1 + 1;
        clear L1 r1 c1 v1 label1 r1_max r1_min c1_max c1_min
    end
end
```

```
%%
%Invers Citra (SCW)

if (~cnt1)                  % Deteksi jika invers citra dibutuhkan
   clear newimage imageAND NEW W1 I1_ob f1 a1 phi1 euler_num1
   figure
   I11 = uint8(255*ones(M1,N1)-double(I1)); % Citra invers
   image = I11;
   clear I11
   imshow(image,[]);

   % Ulangi langkah proses SCW, image masking, SB, and CCA

   %image = [1 2 3 4 5 6;7 8 9 10 11 12];
   imageWidth  = size(image, 2)
   imageHeight = size(image, 1)
   windowWidth = 24;
   windowHeight =8;
   disp('Processing window1...');
   for j = 1:imageHeight - windowHeight + 1
      for i = 1:imageWidth - windowWidth + 1
         window1(j,i) = round((mean(mean(image(j:j + windowHeight -
1, i:i + windowWidth - 1, :)))));
         % lakukan hal-hal dengan subcitra
      end
   end

   imageWidth1  = size(window1, 2)
   imageHeight1 = size(window1, 1)
   windowWidth1 = 12;
   windowHeight1 =4;
   disp('Processing window2..');
   for j = 1:imageHeight - windowHeight1 + 1
      for i = 1:imageWidth - windowWidth1 + 1
         window2(j,i) =round(mean(mean(image(j:j + windowHeight1 -
1, i:i + windowWidth1 - 1, :))));
         % lakukan hal-hal dengan subcitra
      end
   end

   imageWidth2 = size(window2, 2)
```

```
imageHeight2 = size(window2, 1)
newwindow2 = window2(1:imageHeight1,1:imageWidth1);

for j = 1 :imageHeight1
   for i = 1 :imageWidth1
      if((window1(j,i)/newwindow2(j,i))>1.43)%1.25)
         newimage(j,i)=1;
      else
         newimage(j,i)=0;
      end
   end
end
figure;
imshow(newimage);
title('Hasil SCW pada citra (IAND)');

[M1,N1] = size(image);
[M1_new,N1_new] = size(newimage);
diff = size(image) - size(newimage);
newimage = horzcat(zeros(M1_new,(diff(2)-
1)/2+1),newimage,zeros(M1_new,(diff(2)-1)/2));
[M1_new,N1_new] = size(newimage);
newimage = vertcat(zeros((diff(1)-
1)/2+1,N1_new),newimage,zeros((diff(1)-1)/2,N1_new));

imageAND = zeros(M1,N1);
for i = 1 :M1
   for j = 1 : N1
      if(newimage(i,j))
         imageAND(i,j)= image(i,j);
      else
         imageAND(i,j)= 0;
      end
   end
end

figure;
imshow(imageAND,[]);
title('Image Masking (I2)');

%Binerisasi Sauvola
```

```
    windowWidth = 10;
    windowHeight = 10;
    [M1,N1] = size(imageAND);
    imageAND_ext =
horzcat(imageAND(:,1:windowWidth/2),imageAND,imageAND(:,N1-
(windowWidth/2-1):N1));
    imageAND_ext =
vertcat(imageAND_ext(1:windowHeight/2,:),imageAND_ext,imageAN
D_ext(M1-(windowHeight/2-1):M1,:));
    [M1_ext,N1_ext] = size(imageAND_ext);
    imageAND_ext_t = imageAND_ext';

    for j = windowHeight/2+1:N1_ext-windowHeight/2+1
        for i = windowWidth/2+1:M1_ext-windowWidth/2+1
            b = imageAND_ext_t(j-windowHeight/2:j + windowHeight/2 -
1,i-windowWidth/2:i + windowWidth/2 - 1);
            std2 = std(std(b));
            mA = mean(mean(b));
            K = .5;
            R = 128;
            T = mA+(1+K*((std2/R)-1));
            if (imageAND_ext_t(j,i) >= T)
                NEW_ext(j,i) = 1;
            else
                NEW_ext(j,i) = 0;
            end
        end
    end
    NEW_ext2 = NEW_ext';
    NEW = NEW_ext2(windowWidth/2+1:M1_ext-
windowWidth/2+1,windowHeight/2+1:N1_ext-windowHeight/2+1);

    figure;
    imshow(NEW,[]);
    title('Hasil citra setelah binerisasi Sauvola (I3)');
    clear imageAND_ext NEW M1_ext N1_ext

    figure(ob_cnt+5)
    f1 = conv2(double(I_bin1),double(g));
    f1 = f1(1:M1,1:N1);
    [W1,num1] = bwlabel(f1,8);
```

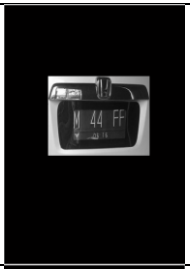
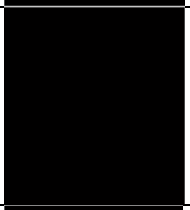
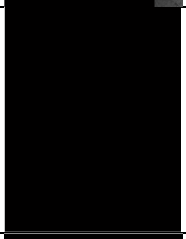
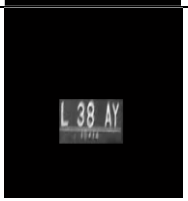
```
imshow(W1,[]);

figure
cnt1 = 0;
for k = 1:num1              % Menghitung aspect ratio
    L1 = zeros(M1,N1);
    [r1,c1,v1] = find(W1==k);
    label1 = vertcat(r1',c1');
    for i = 1:size(label1,2)
        L1(label1(1,i),label1(2,i)) = W1(label1(1,i),label1(2,i));
    end
    r1_max = max(label1(1,:));
    r1_min = min(label1(1,:));
    c1_max = max(label1(2,:));
    c1_min = min(label1(2,:));
    a1(k) = (c1_max - c1_min + 1)/(r1_max - r1_min + 1);
    clear L1 r1 c1 v1 label1
end
for k = 1:num1              % Menghitung sudut orientasi
    L1 = zeros(M1,N1);
    [r1,c1,v1] = find(W1==k);
    label1 = vertcat(r1',c1');
    for i = 1:size(label1,2)
        L1(label1(1,i),label1(2,i)) = W1(label1(1,i),label1(2,i));
        rc_component1(label1(1,i),label1(2,i)) = label1(1,i)*label1(2,i)...
                                *L1(label1(1,i),label1(2,i));
        r2_component1(label1(1,i),label1(2,i)) = label1(1,i)^2*...
                                L1(label1(1,i),label1(2,i));
        c2_component1(label1(1,i),label1(2,i)) = label1(2,i)^2*...
                                L1(label1(1,i),label1(2,i));
    end
    phi1(k) = atand((2*sum(sum(rc_component1)))/...
            (sum(sum(r2_component1))-sum(sum(c2_component1))));
    clear L1 r1 c1 v1 label1 rc_component1 r2_component1
c2_component1
end
for k = 1:num1              % Menghitung bilangan Euler
    L1 = zeros(M1,N1);
    [r1,c1,v1] = find(W1==k);
    label1 = vertcat(r1',c1');
    for i = 1:size(label1,2)
```

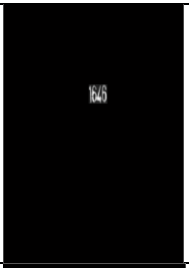
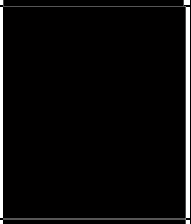
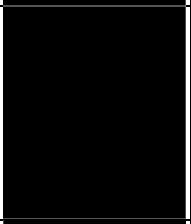
```
        L1(label1(1,i),label1(2,i)) = W1(label1(1,i),label1(2,i));
    end
    L1_ed = edge(L1,'canny');
    euler_num1(k) = bweuler(L1_ed,8);
    clear L1 r1 c1 v1 label1 L1_ed
  end

  I1_ob = zeros(M1,N1);
  I1_f = zeros(M1,N1);
  for k = 1:num1
    if ((a1(k) > 1.5)&&(a1(k) < 5.5))&&...
      (((phi1(k)+180) > 105)&&((phi1(k)+180) < 140))&&...
      (euler_num1(k) >= 3)
      L1 = zeros(M1,N1);
      [r1,c1,v1] = find(W1==k);
      label1 = vertcat(r1',c1');
      for i = 1:size(label1,2)
        L1(label1(1,i),label1(2,i)) = W1(label1(1,i),label1(2,i));
      end
      for i = 1:length(label1)
        I1_ob(label1(1,i),label1(2,i)) = L1(label1(1,i),label1(2,i));
      end
      r1_max = max(label1(1,:));
      r1_min = min(label1(1,:));
      c1_max = max(label1(2,:));
      c1_min = min(label1(2,:));
      A1(cnt1+1,:) = [r1_max,r1_min,c1_max,c1_min];
      I1_f(r1_min:r1_max,c1_min:c1_max) = ...
                              I1(r1_min:r1_max,c1_min:c1_max);
      cnt1 = cnt1 + 1;
      clear L1 r1 c1 v1 label1 r1_max r1_min c1_max c1_min
    end
  end
end

%%
%Jika setelah invers citra dan tidak ada deteksi plat nomor

if (~cnt1)
  disp('Tidak ada Plat Nomor Kendaraan pada citra');
end
```

```
%%
% Hasil Akhir
figure
imshow(I1_ob,[])          % Lokalisasi wilayah Plat Nomor Kendaraan
pada citra

figure
imshow(I1_f,[])          % Ekstraksi Plat Nomor Kendaraan pada citra
```

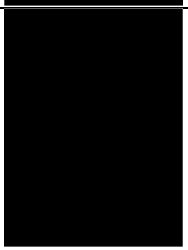
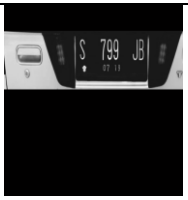
Halaman ini sengaja dikosongkan

LAMPIRAN B
Hasil Deteksi Plat Nomor Kendaraan

No.	Citra Plat Asli	Hasil Graysale	Hasil Deteksi	Ukuran Piksel
1.				500 x 423
2.				500 x 500
3.				420 x 305
4.				380 x 383
5.				508 x 284

No.	Citra Plat Asli	Hasil Graysale	Hasil Deteksi	Ukuran Piksel
6.				500 x 377
7.				482 x 488
8.				500 x 359
9.				500 x 399
10.				500 x 400

No.	Citra Plat Asli	Hasil Graysale	Hasil Deteksi	Ukuran Piksel
11.				510 x 382
12.				500 x 350
13.				398 x 541
14.				499 x 352
15.				501 x 351

No.	Citra Plat Asli	Hasil Graysale	Hasil Deteksi	Ukuran Piksel
16.				500 x 355
17.				383 x 326
18.				500 x 505
19.				500 x 499
20.				500 x 380

RIWAYAT HIDUP

Rangga Imantaka Firswandy dilahirkan di Tulungagung, Jawa Timur pada tanggal 29 Juni 1993. Penulis merupakan anak pertama dari pasangan Eko Subagyo dan Krisna Atmawati. Penulis memulai pendidikan formal di TK Mutiara Sidoarjo (1998-2000), SD Laboratorium UNESA Ketintang pada tahun 2000-2006, SMP Laboratorium YDWP UNESA Ketintang (2006-2009), dan SMA Al Falah Ketintang (2009-2012). Setelah lulus SMA, penulis melanjutkan studi S-1 Teknik Elektro, Fakultas Teknologi Elektro, Institut Teknologi Sepuluh Nopember. Penulis berhasil diterima di Institut Teknologi Sepuluh Nopember melalui Jalur Kemitraan pada Program Kemitraan dan Mandiri (PKM) ITS. Sebelumnya, penulis dapat diterima di Program Studi S-1 Instrumentasi, Jurusan Fisika, Fakultas Matematika dan Ilmu Pengetahuan Alam (FMIPA), Universitas Brawijaya, Kota Malang melalui Jalur SNMPTN Tulis. Namun, penulis yang bersangkutan tidak mengambil program studi tersebut karena adanya pertimbangan orang tua dan penulis. Pada September 2014, penulis memilih bidang studi Telekomunikasi Multimedia. Penulis menulis buku tugas akhir ini sebagai salah satu syarat untuk memperoleh gelar sarjana pada wisuda periode Maret 2019. Penulis dapat dihubungi melalui alamat *e-mail* : rangga.imantaka93@gmail.com.

Halaman ini sengaja dikosongkan