

Sistem Deteksi Pelat Nomor Kendaraan Roda Dua

Berdasarkan Variabel Kecepatan



TUGAS AKHIR

Disusun dalam rangka memenuhi salah satu persyaratan Untuk menyelesaikan program Strata-1 Prodi Informatika Departemen Informatika Universitas Hasanuddin Makassar

Disusun Oleh:

ANDI MARIMAR MUCHTAMAR

D42116303

**PROGRAM STUDI TEKNIK INFORMATIKA
DEPARTEMEN TEKNIK INFORMATIKA FAKULTAS TEKNIK
UNIVERSITAS HASANUDDIN**

2021

SKRIPSI

Sistem Deteksi Pelat Nomor Kendaraan Roda Dua

Berdasarkan Variabel Kecepatan

Disusun dan diajukan oleh

ANDI MARIMAR MUCHTAMAR

D42116303



**PROGRAM STUDI TEKNIK INFORMATIKA
DEPARTEMEN TEKNIK INFORMATIKA FAKULTAS TEKNIK
UNIVERSITAS HASANUDDIN**

2021

LEMBAR PENGESAHAN SKRIPSI

SISTEM DETEKSI PELAT NOMOR KENDARAAN RODA DUA BERDASARKAN
VARIABEL KECEPATAN

Disusun dan diajukan oleh

ANDI MARIMAR MUCHTAMAR

D421 16 303

Telah dipertahankan di hadapan Panitia Ujian yang dibentuk dalam rangka penyelesaian studi Program Sarjana Program Studi Teknik Informatika Fakultas Teknik Universitas Hasanuddin

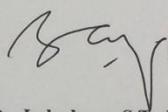
Pada tanggal 22 Oktober 2021

dan dinyatakan telah memenuhi syarat kelulusan

Menyetujui

Pembimbing Utama,

Pembimbing Pendamping,


Dr. Indrabayu, S.T., M.T., M.Bus.Svs.
NIP. 19750716 200212 1 004


Dr. Eng. Intan Sari Areni, S.T., M.T.
NIP. 19750203 200012 2 002

Ketua Program Studi,


Dr. Amil Ahmad Ilham, S.T., M.IT.
NIP. 19731010 199802 1 001

PERNYATAAN KEASLIAN

Yang bertanda tangan di bawah ini :

Name : Andi Marimar Muchtamar
NIM : D42116303
Program Studi : Teknik Informatika
Jenjang : S1

Menyatakan dengan ini bahwa hasil karya tulisan saya berjudul :

Sistem Deteksi Pelat Nomor Kendaraan Roda Dua Berdasarkan Variabel Kecepatan

Adalah karya tulisan saya sendiri. Bukan merupakan pengambilan alihan tulisan orang lain dan bahwa skripsi ini yang saya tulis ini benar benar merupakan hasil karya saya sendiri.

Apabila dikemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan isi skripsi ini hasil karya orang lain. Maka saya bersedia menerima sanksi atas perbuatan tersebut.

Makassar, 1 November 2021

Yang menyatakan,



ANDI MARIMAR MUCHTAMAR

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh.

Puji dan syukur senantiasa penulis panjatkan atas kehadiran Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan laporan Tugas Akhir ini yang berjudul "**Sistem Deteksi Pelat Nomor Kendaraan Roda Dua Berdasarkan Variabel Kecepatan**" sebagai salah satu syarat untuk memperoleh gelar Sarjana Strata Satu (S1) pada Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin.

Dalam proses pembuatan laporan akhir ini tidak lepas dari bantuan, bimbingan serta dukungan dari berbagai pihak, dari masa perkuliahan sampai penyusunan Tugas Akhir. Oleh karena itu dengan segala kerendahan hati, penulis mengucapkan rasa terima kasih sebesar-besarnya kepada:

1. Kedua Orang tua penulis, Bapak Muchtamar Pandaka dan Ibu Andi Amira S.Pd yang selalu memberikan motivasi, dukungan, doa, semangat dan kasih sayang serta selalu sabar dalam mendidik penulis sejak kecil.
2. Bapak Dr. Indrabayu, S.T., M.T., M.Bus.Sys., selaku dosen pembimbing I dan Ibu Dr.Eng. Intan Sari Areni, S.T. M.T., selaku dosen pembimbing II yang selalu menyediakan waktu, tenaga dan pikirannya yang luar biasa untuk mengarahkan penulis dalam penyusunan Tugas Akhir ini
3. Bapak Dr.Amil Ahmad Ilham, S.T., M.I.T., selaku Ketua Departemen Teknik Informatika Fakultas Teknik Univeristas Hasanuddin atas bimbingannya selama masa perkuliahan penulis.

4. Segenap Dosen dan Staff Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin yang telah banyak membantu penulis selama masa perkuliahan.
5. Teman-teman dan kakak-kakak *AIMP Research Group* FT-UH yang telah memberikan begitu banyak bantuan selama penelitian, pengambilan data, dan diskusi terkait progres penyusunan Tugas Akhir.
6. Sahabat-sahabat AIROKU, Cici Purnamasari, Putri Angriani, Sitti Nur Fadillah, Dhinda Fitri Wiludjeng dan Ghina Syukriyah yang selalu menjadi tempat berbagi cerita dan memberikan perhatian dan semangat kepada penulis.
7. Teman-teman Igniter16 atas dukungan dan semangat yang telah diberikan
8. Orang-orang berpengaruh lainnya yang tidak sempat penulis sebutkan satu persatu yang telah memberikan dukungan, semangat dan doanya selama penyusunan Tugas Akhir ini.

Akhirnya dengan segala kerendahan hati, penulis menyadari masih terdapat kekurangan dalam penyusunan laporan ini baik isi maupun cara penyajian. Oleh karena itu penyusun mengharapkan adanya saran dan kritik yang bersifat membangun demi kesempurnaan laporan ini.

Makassar, Juni 2020

Penulis

ABSTRAK

Kepadatan kendaraan di dominasi oleh kendaraan roda dua. Dari data Badan Pusat Statistik tahun 2018 yang bersumber dari Kantor Kepolisian Republik Indonesia menuliskan bahwa jumlah sepeda motor untuk wilayah Indonesia sebanyak 120.101 Juta dari jumlah kendaraan sebanyak 146.858 Juta. Jumlah kendaraan yang banyak mengakibatkan aparat kepolisian sulit untuk menindak lanjuti pengendara jika terjadi pelanggaran. Peristiwa ini menjadi masalah karena kurangnya bukti dalam pelanggaran sehingga jika terjadi pelanggaran tidak dapat ditindak lanjuti. Nomor polisi yang tercantum pada pelat kendaraan bisa digunakan oleh pihak berwajib sebagai barang bukti pengendara yang melakukan pelanggaran lalu lintas. Penelitian-penelitian sebelumnya yang telah dilakukan membuat sistem deteksi pelat dengan satu kendaraan saja. Maka pada penelitian ini mengembangkan sistem yang dapat mendeteksi dan mengenali teks pelat lebih dari satu kendaraan atau *multi detection*. Metode yang digunakan adalah *Local Binary Pattern* (LBP) untuk mendeteksi pelat dan *Optical Character Recognition* (OCR) untuk mengenali teks pada pelat kendaraan roda dua. Penelitian ini menggunakan tiga skenario yang berbeda-beda. Pada skenario pertama dengan 1 kendaraan, skenario kedua dengan 2 kendaraan dan skenario ketiga dengan 3 kendaraan dengan masing-masing kecepatan 40 km/jam, 50 km/jam dan 60 km/jam, diambil menggunakan kamera static Vivotek ip 9165-hp. Hasil penelitian menunjukkan akurasi untuk deteksi pelat menghasilkan akurasi 100%. Tetapi untuk mengenali teks karakter mendapatkan akurasi tertinggi 80% pada *single detection* dan untuk *multi detection* menghasilkan akurasi tertinggi 90%. Data uji digunakan sebanyak 16 file data dengan berdurasi 30 detik dengan masing-masing video terdapat 9 sampai 10 kendaraan roda dua.

Kata Kunci : Pelat Nomor Kendaraan, *Local Binary Pattern* (LBP), *Optical Character Recognition* (OCR).

DAFTAR ISI

KATA PENGANTAR	i
ABSTRAK.....	iii
DAFTAR ISI	iv
DAFTAR GAMBAR	vii
DAFTAR TABEL	x
BAB I.....	1
PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	3
1.3 Tujuan Penelitian.....	3
1.4 Batasan Masalah	3
1.5 Manfaat Penelitian.....	4
1.6 Sistematika Penulisan.....	4
BAB II.....	6
TINJAUAN PUSTAKA	6
2.1 Tanda Nomor Kendaraan Bermotor	6
2.2 Pengolahan Citra Digital	7
2.3 Jenis Jenis Citra	10

2.4 Konversi Citra.....	13
2.5 <i>Thresholding</i>	15
2.6 Morfologi Citra.....	16
2.7 Analisis Blob.....	20
2.8 <i>Tracking</i>	21
2.9 <i>Kalman Filter</i>	21
2.10 <i>Gaussians Mixture Model (GMM)</i>	23
2.11 <i>Local Binary Pattern (LBP)</i>	24
2.12 <i>Optical Character Recognition (OCR)</i>	25
2.13 <i>License Plate Recognition</i>	27
2.14 <i>Region Plate Extraction</i>	27
2.15 Segmentasi Pelat.....	28
2.16 <i>Character Recognition</i>	28
2.17 Penelitian Terkait.....	29
BAB III	33
METODOLOGI PENELITIAN	33
3.1 Tahap Penelitian	33
3.2. Waktu dan Lokasi Penelitian	34
3.3. Instrumen Penelitian.....	35
3.4. Teknik Pengambilan Data.....	36

3.5 Perancangan Sistem.....	38
3.5.7 Membagi 3 Bagian Pelat Kendaraan.....	61
3.6 Analisis Kinerja Sistem.....	68
BAB IV	69
HASIL DAN PEMBAHASAN	69
4.1 Hasil Penelitian.....	69
4.2 Pembahasan	113
BAB V.....	123
PENUTUP.....	123
5.1 Kesimpulan	123
5.2 Saran.....	123
DAFTAR PUSTAKA	125
LAMPIRAN	128

DAFTAR GAMBAR

Gambar 2.1 Pengolahan Citra (Syuhada, 2015)	10
Gambar 2.2 Citra biner dan representasinya dalam data digital (a) citra biner (b) nilai penyimpanan di memori (SAHTONI, 2012).	11
Gambar 2.3 Citra Biner (SAHTONI, 2012)	11
Gambar 2.4 Citra <i>Grayscale</i> (SAHTONI, 2012)	12
Gambar 2.5 Citra warna (<i>true color</i>) (SAHTONI, 2012)	13
Gambar 2.6 Proses <i>Threshold</i> beberapa tingkat (Tahir, 2017).....	15
Gambar 2.7 Proses dilasi dengan <i>SE</i> berukuran 3×3 dengan semua elemen <i>SE</i> bernilai 1 (Putra, 2010)	18
Gambar 2.8 Proses erosi dengan <i>SE</i> berukuran 3×3 dengan semua elemen <i>SE</i> bernilai 1. (Putra, 2010)	19
Gambar 2.9 Analisis Luasan Blob Sebuah Objek Kendaraan (Basri, 2015).	21
Gambar 2.10 Contoh Implementasi <i>LBP</i> (a) Nilai Pixel pada Citra, (b) Nilai <i>LBP</i> dari Citra (a) (Afrizal Firdaus, 2016).	25
Gambar 2.11 Blok Diagram Kerja OCR (Trilaksono & Rizal, 2008).	26
Gambar 2.12 Proses segmentasi karakter secara vertikal (Basri, 2015).	28
Gambar 2.13 (a) Database Karakter. (b) Segmentasi Karakter. (c) Pengenalan Karakter (Basri, 2015).....	29
Gambar 3.1 Tahapan Penelitian	33

Gambar 3.2 Lokasi Pengambilan Data	35
Gambar 3.3 Deskripsi Posisi Kamera dan Kendaraan yang dideteksi	37
Gambar 3.4 Ilustrasi Pengambilan Data	38
Gambar 3.5 <i>Flowchart</i> Sistem Deteksi Kendaraan	39
Gambar 3.6 Tampilan File video.....	41
Gambar 3.7 <i>Frame RGB</i>	41
Gambar 3.8 Contoh Citra Hasil Konversi ke <i>Grayscale</i>	42
Gambar 3.9 Contoh Hasil inialisasi <i>Background</i> dengan <i>Foreground</i>	44
Gambar 3.10 Citra Hasil Deteksi Kendaraan Roda Dua yang Terdeteksi dengan <i>Bounding Box</i>	46
Gambar 3.11 <i>Flowchart</i> Deteksi Pelat Kendaraan.....	49
Gambar 3.12 Alur Proses Pelatihan Deteksi Pelat	50
Gambar 3.13. Contoh Frame Positif	50
Gambar 3.14 Contoh ROI Data Positif	51
Gambar 3.15 Contoh Data Positif dan Negatif.....	52
Gambar 3.16 Proses Pelatihan dengan 17 Tahap	54
Gambar 3.17 Ilustrasi proses sliding window	55
Gambar 3.18. Contoh Hasil Deteksi Pelat	55
Gambar 3.19. <i>Flowchart</i> Pengenalan Teks Pelat.....	56
Gambar 3.20 Citra Kandidat Pelat	57

Gambar 3.21. Citra Hasil Konversi RGB ke <i>Grayscale</i>	57
Gambar 3.22 Citra Hasil <i>resize</i>	58
Gambar 3.23 (a) Hasil Konversi <i>Grayscale</i> ke Biner (b) Cita Hasil Konversi <i>Grayscale</i> ke Biner menggunakan Top Hat dengan ‘disk’ 15.....	58
Gambar 3.24 Citra Hasil olah ROI.....	60
Gambar 3.25 Citra Pelat Hasil <i>Preprocessing</i>	61
Gambar 3.26 Bagian Pertama Pelat.....	62
Gambar 3.27 Bagian Kedua Pelat	62
Gambar 3.28 Bagian Ketiga Pelat	63
Gambar 3.29 Blok Diagram Tahap Pengenalan Pelat Kendaraan	66
Gambar 3.30 Huruf Pelat Standar Indonesia	66
Gambar 3.31 Tampilan Fitur <i>OcrTrainer</i>	67
Gambar 3.32 Hasil Visualisasi Sistem berupa Output Deteksi Pelat.....	68
Gambar 4.3 Pelat Kendaraan yang Gagal Dikenali dengan Benar	120
Gambar 4.4 Pelat Kendaraan yang Tidak Akurat dalam Mengenali Teks	121
Gambar 4.5 Data Pelat yang Paling Banyak Terjadi Kesalahan	121
Gambar 4.6 Hasil tidak Menggunakan <i>contrast enhancement</i>	122
dan Menggunakan <i>contrast enhancement</i>	122

DAFTAR TABEL

Tabel 3.11 Fungsi Variabel Penentuan Blob Analisis	45
Tabel 4.1 Hasil Deteksi Kendaraan Roda Dua Pada Kecepatan 40 km/jam, 50 km/jam dan 60 km/jam.	70
Tabel 4.2 Hasil Uji Deteksi dan Pengenalan Pelat Kendaraan Roda Dua dengan Kecepatan 40 km/jam.....	71
Tabel 4.2 (Lanjutan).....	72
Tabel 4.3 Hasil Uji Deteksi dan Pengenalan Pelat Kendaraan Roda Dua dengan Kecepatan 50 km/jam.....	74
Tabel 4.3 (Lanjutan).....	75
Tabel 4.4 Hasil Uji Deteksi dan Pengenalan Pelat Kendaraan Roda Dua dengan Kecepatan 60 km/jam.....	76
Tabel 4.4 (Lanjutan).....	77
Tabel 4.5 Hasil Uji Deteksi dan Pengenalan Pelat Kendaraan Roda Dua dengan Kecepatan 40 km/jam.....	79
Tabel 4.5 (Lanjutan).....	80
Tabel 4.6 Hasil Uji Deteksi dan Pengenalan Pelat Kendaraan Roda Dua dengan Kecepatan 50 km/jam.....	81
Tabel 4.6 (Lanjutan).....	82

Tabel 4.7 Hasil Uji Deteksi dan Pengenalan Pelat Kendaraan Roda Dua dengan Kecepatan 60 km/jam.....	83
Tabel 4.7 (Lanjutan).....	84
Tabel 4.8 Hasil Uji Deteksi dan Pengenalan Pelat Kendaraan Roda Dua dengan Kecepatan 40 km dan 50 km	85
Tabel 4.8 (Lanjutan).....	86
Tabel 4.9 Hasil Uji Deteksi dan Pengenalan Pelat Kendaraan Roda Dua dengan Kecepatan 40 dan 60 km/jam.	88
Tabel 4.9 (Lanjutan).....	89
Tabel 4.10 Hasil Uji Deteksi dan Pengenalan Pelat Kendaraan Roda Dua dengan Kecepatan 50 dan 60 km/jam.	90
Tabel 4.10 (Lanjutan).....	91
Tabel 4.11 Hasil Uji Deteksi dan Pengenalan Pelat Kendaraan Roda Dua dengan Kecepatan 40 km/jam.....	93
Tabel 4.11 (Lanjutan).....	94
Tabel 4.12 Hasil Uji Deteksi dan Pengenalan Pelat Kendaraan Roda Dua dengan Kecepatan 40 km/jam.....	95
Tabel 4.12 (Lanjutan).....	96
Tabel 4.13. Hasil Uji Deteksi dan Pengenalan Pelat Kendaraan Roda Dua dengan Kecepatan 50 km/jam.....	97
Tabel 4.13 (Lanjutan).....	98

Tabel 4.14 Hasil Uji Deteksi dan Pengenalan Pelat Kendaraan Roda Dua dengan Kecepatan 50 km/jam.....	99
Tabel 4.14 (Lanjutan).....	100
Tabel 4.15. Hasil Uji Deteksi dan Pengenalan Pelat Kendaraan Roda dua dengan Kecepatan 60 km/jam.....	101
Tabel 4.15 (Lanjutan).....	102
Tabel 4.16. Hasil Uji Deteksi dan Pengenalan Pelat Kendaraan Roda dua dengan Kecepatan 60 km/jam.....	104
Tabel 4.17. Hasil Uji Deteksi Pelat dan Pengenalan pelat kendaraan Roda Dua dengan Kecepatan 40,50 dan 60 km/jam.	106
Tabel 4.17 (Lanjutan).....	107
Tabel 4.18 Akurasi Keseluruhan Pengenalan Pelat pada Kecepatan 40 km, 50 km dan 60 km.....	113

BAB I

PENDAHULUAN

1.1 Latar Belakang

Salah satu karakteristik dari kendaraan yaitu pelat nomor kendaraan. Pelat merupakan identitas utama dari sebuah kendaraan bermotor yang layak jalan / beroperasi di jalan raya, sehingga pelat nomor polisi umumnya diletakkan pada posisi yang mudah dilihat, Identitas inilah yang membedakan antara kendaraan yang satu dengan yang lainnya.

Data dari Badan Pusat Statistik tahun 2018 yang bersumber dari Kantor Kepolisian Republik Indonesia menuliskan bahwa jumlah sepeda motor untuk wilayah Indonesia sebanyak 120.101 Juta dari jumlah kendaraan sebanyak 146.858 Juta. Untuk wilayah Sulawesi Selatan berdasarkan data Badan Pusat Statistik tahun 2015 menuliskan bahwa jumlah sepeda motor wilayah Makassar yaitu 1.062.943 Juta dari jumlah keseluruhan kendaraan yaitu 2.701.739 Juta yang terdapat di wilayah Sulawesi Selatan (Badan Pusat Statistik, 2018).

Pada kota-kota besar banyak jalan utama yang padat kendaraan. Dengan kepadatan lalu lintas, sering terjadi pelanggaran lalu lintas di mana pengendara yang ugal-ugalan melewati batas kecepatan yang telah ditentukan. Peristiwa ini menjadi masalah karena kurangnya bukti dalam pelanggaran sehingga jika terjadi pelanggaran tidak dapat ditindak lanjuti.

Berdasarkan peraturan Menteri no.11 tahun 2015 tentang batas kecepatan kendaraan. Kecepatan kendaraan pada daerah pemukiman yang padat, batas

kecepatan paling tinggi adalah 30 km/jam dan batas kecepatan di daerah yang tidak padat, kecepatan paling tinggi adalah 50 km/jam.

Nomor polisi yang tercantum pada pelat kendaraan bisa digunakan oleh pihak berwajib sebagai barang bukti pengendara melakukan pelanggaran lalu lintas. Jumlah kendaraan yang banyak mengakibatkan aparat kepolisian sulit untuk mengenali pelat nomor kendaraan secara manual, Penelitian-penelitian sebelumnya yang telah dilakukan membuat sistem deteksi pelat dengan satu kendaraan saja. Maka pada penelitian ini mengembangkan sistem yang dapat mendeteksi dan mengenali teks pelat lebih dari satu kendaraan atau *multi detection*.

Ada berbagai macam metode untuk mendeteksi posisi pelat dan mengenali teks pelat kendaraan roda dua. Salah satunya yaitu metode *Local Binary Pattern* (LBP) untuk mendeteksi posisi pelat dan *Optical Character Recognition* (OCR) untuk mengenali teks pelat. metode *Local Binary Pattern* (LBP) adalah salah satu metode yang paling bagus dan cocok digunakan untuk mengekstraksi ciri berupa tekstur dari pelat kendaraan yang akan diuji (Syawaluddin, 2016). *Optical Character Recognition* (OCR) adalah algoritma yang berfungsi untuk menscan citra dan dijadikan teks. Teks yang discan dengan OCR dapat dicari kata per kata atau per kalimat. Oleh karena itu pada penelitian ini digunakan metode LBP dan OCR untuk mendeteksi dan mengenali teks pelat kendaraan roda.

1.2 Rumusan Masalah

- a. Bagaimana sistem dapat mendeteksi pelat kendaraan roda dua dengan *multi detection* ?
- b. Bagaimana unjuk kerja sistem dalam mendeteksi pelat dengan *multi detection* dan pengenalan teks pelat kendaraan roda dua menggunakan *Local Binary Pattern* dan *Optical Character Recognition* ?

1.3 Tujuan Penelitian

- a. Membuat sistem yang dapat mendeteksi pelat kendaraan roda dua dengan *multi detection*.
- b. Unjuk kerja sistem dalam mendeteksi pelat dengan *multi detection* dan pengenalan teks pelat kendaraan roda dua menggunakan *Local Binary Pattern* dan *Optical Character Recognition*.

1.4 Batasan Masalah

Yang menjadi batasan masalah dalam penelitian ini adalah :

1. Kendaraan yang dideteksi adalah kendaraan roda dua
2. Data yang diolah berupa data video dengan resolusi 1920x1080
3. Waktu pengambilan data dilakukan disiang hari
4. Data video diambil menggunakan static kamera dengan sudut 60°
5. Pengambilan data dilakukan dari arah depan
6. Pelat yang dikenali bukan pelat modifikasi

1.5 Manfaat Penelitian

Menghasilkan sistem yang dapat mendeteksi pelat lebih dari satu kendaraan roda dua atau *multi detection* dan mengenali teks pada pelat kendaraan roda dua.

1.6 Sistematika Penulisan

Untuk memberikan gambaran singkat mengenai isi tulisan secara keseluruhan, maka akan diuraikan beberapa tahapan dari penulisan secara sistematis, yaitu :

BAB I PENDAHULUAN

Bab ini menguraikan secara umum mengenai hal yang menyangkut latar belakang, perumusan masalah dan batasan masalah, tujuan, manfaat dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Bab ini berisi teori-teori terkait hal-hal yang mendasari dan yang berhubungan dengan Pengolahan Citra, Pemrosesan Citra dan metode-metode yang digunakan pada penelitian ini.

BAB III METODOLOGI PENELITIAN

Bab ini memaparkan tahapan penelitian, waktu dan lokasi penelitian, instrumen penelitian, teknik pengambilan data dan rancangan sistem serta penerapan metode *Local Binary Pattern* dan *Optical Character Recognition* serta teknik pengolahan data.

BAB IV HASIL DAN PEMBAHASAN

Bab ini berisi tentang hasil penelitian dan pembahasan terkait pengolahan data yang telah dilakukan yang disertai dengan hasil penelitian.

BAB V PENUTUP

Bab ini berisi kesimpulan yang diperoleh dari hasil penelitian yang telah dilakukan serta saran-saran untuk pengembangan sistem lebih lanjut.

BAB II

TINJAUAN PUSTAKA

2.1 Tanda Nomor Kendaraan Bermotor

Berdasarkan peraturan kepala kepolisian RI Nomor 5 Tahun 2012 Tentang Registrasi dan Identifikasi Kendaraan Bermotor, Tanda Nomor Kendaraan Bermotor (TNKB) atau sering disebut pelat nomor merupakan tanda resident Ranmor yang berfungsi sebagai bukti legitimasi pengoperasian Ranmor berupa pelat atau berbahan lain dengan spesifikasi tertentu yang diterbitkan Polri dan berisikan kode wilayah, nomor registrasi, serta masa berlaku dan dipasang pada Ranmor (KAPOLRI, 2012)

Bahan baku TNKB adalah aluminium dengan ketebalan 1 mm. ukuran TNKB untuk kendaraan bermotor roda 2 adalah 250x105 mm. Korps Lantas Mabes Polri terhitung mulai April 2011 mengganti desain pelat nomor kendaraan. Ukuran lebih panjang 5 sentimeter dari pelat nomor sebelumnya. Ukuran TNKB untuk kendaraan roda 2 sekarang menjadi 275 mm dengan lebar 110 mm (Aditya, 2016).

Indonesia memiliki karakteristik pelat nomor tersendiri. Biasanya berformat pada nomor di Indonesia adalah **HH AAAA HH**, dimana **H** merepresentasikan huruf dan **A** merepresentasikan angka 0-9. Satu atau dua huruf pertama merepresentasikan area dimana kendaraan tersebut didaftarkan. Huruf ini diikuti dengan satu sampai empat angka, kemudian diakhiri oleh satu atau dua huruf. Misalnya, **BM 6015 TV** merupakan kendaraan dari Pekanbaru, karena dua huruf pertamanya, **BM**, merupakan kode untuk Pekanbaru. Sejak 2008, daerah

Jakarta dan sekitarnya memperkenalkan format baru untuk pelat mobil, yaitu **HH AAAA HHH**. Sebelumnya, format ini telah diperkenalkan terlebih dahulu untuk pelat motor (Aditya, 2016).

2.2 Pengolahan Citra Digital

Pengolahan citra (*image Processing*) merupakan proses mengolah piksel-piksel di dalam citra digital untuk tujuan tertentu. Pada awalnya pengolahan citra ini dilakukan untuk memperbaiki kualitas citra, namun dengan berkembangnya dunia komputasi yang ditandai dengan semakin meningkatnya kapasitas dan kecepatan proses komputer serta munculnya ilmu-ilmu komputasi yang memungkinkan manusia dapat mengambil informasi dari suatu citra (Steven, 2010).

Citra biasa digunakan dalam dunia teknologi informasi sehingga citra memiliki definisi khusus dimana citra merupakan sebuah gambar yang berisi informasi data, baik data berupa citra yang ditampilkan maupun data yang menyusun informasi citra itu sendiri. Citra-citra inilah jika kita ingin merubahnya dari satu jenis ke jenis yang lain memiliki sebuah seni tersendiri. Citra memiliki jenis file atau code dalam bentuk digital (citra yang berada di laptop atau komputer). Perbedaan-perbedaan jenis citra inilah jika kita ingin merubahnya dari satu jenis ke jenis yang lain memiliki sebuah seni tersendiri, sehingga menimbulkan dinamika file image. Dalam definisi yang lebih luas, pengolahan citra digital juga mencakup semua data dua dimensi. Citra digital adalah barisan bilangan nyata maupun kompleks yang diwakili oleh bit-bit tertentu (Steven, 2010).

Dalam melakukan pengolahan citra ada beberapa metodologi yang perlu dilakukan untuk mencapai sebuah tujuan. Adapun langkah – langkah sebagai berikut, (Ikhsanuddin, 2014) :

1. Pembentukan Citra (*Data Acquisition*)

Menentukan data yang diperlukan dan memilih metode perekam citra digital. Akuisisi citra merupakan proses menangkap (*capture*) atau memindai (*scan*) suatu citra analog sehingga diperoleh citra digital. Beberapa faktor yang perlu diperhatikan dalam proses akuisisi citra antara lain adalah: jenis alat akuisisi, resolusi kamera, teknik pencahayaan, perbesaran atau zooming, jarak, dan sudut pengambilan citra.

2. Pengolahan Citra Tingkat Awal (*Image Preprocessing*)

Pada tahap ini dimana dalam meningkatkan kualitas citra dapat meningkatkan kualitas kontras, brightness, menghilangkan noise, perbaikan citra, transformasi, menghilangkan gangguan geometrik / radiometrik, menentukan bagian citra yang akan diobservasi.

3. Segmentasi Citra (*Image Segmentation*) dan Deteksi Sisi (*Edge Detection*)

Proses dimana melakukan partisi citra menjadi wilayah – wilayah objek (*internal properties*) atau menentukan garis batas wilayah obyek (*external shape characteristics*). Tahap ini bertujuan untuk mempartisi citra menjadi bagian-bagian pokok yang mengandung informasi penting misalnya memisahkan objek dan latar belakang segmentasi terdiri dari downsampling, penapisan dan deteksi tepian. Tahap downsampling merupakan proses untuk menurunkan jumlah piksel dan menghilangkan

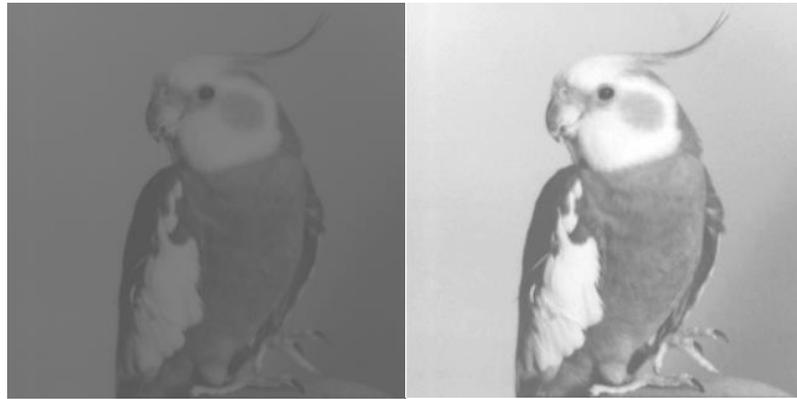
sebagian informasi dari citra. Dengan resolusi citra yang tetap, downsampling menghasilkan ukuran citra yang lebih kecil. Tahap segmentasi selanjutnya adalah penapisan dengan filter median, hal ini dilakukan untuk menghilangkan derau yang biasanya muncul pada frekuensi tinggi pada spectrum citra. Pada penapisan dengan filter median, gray level citra pada setiap piksel digantikan dengan nilai median dari gray level pada piksel yang terdapat pada window filter. Tahap terakhir pada proses segmentasi yaitu deteksi tepian. Pendekatan tepi ini dirancang untuk mempresentasikan sebuah tepian yang ideal, dengan ketebalan yang diinginkan. Secara umum, proses segmentasi sangat penting dan secara langsung akan menentukan keakuratan sistem.

4. Seleksi dan Ekstraksi Ciri (*Feature Extraction and Selection*)

Seleksi ciri dimaksudkan untuk memilih informasi kuantitatif dari ciri yang ada, yang dapat membedakan kelas – kelas obyek secara baik. Ekstraksi ciri digunakan untuk mengukur besaran kuantitatif ciri di setiap piksel dari sebuah citra.

Pengolahan citra adalah ilmu untuk memanipulasi gambar, yang melingkupi teknik-teknik untuk memperbaiki atau mengurangi kualitas gambar, menampilkan bagian tertentu dari gambar, membuat sebuah gambar yang baru dari beberapa bagian gambar yang sudah ada, dan beberapa teknik manipulasi gambar lainnya, pemrosesan citra, khususnya dengan menggunakan komputer, menjadi citra yang kualitasnya lebih baik. **Gambar 2.1** merupakan contoh pengolahan citra. Pada **Gambar 2.1** (a) memperlihatkan citra burung nuri yang

agak gelap. Setelah dilakukan perbaikan kontras, citra burung nuri menjadi lebih terang seperti yang terlihat pada **Gambar 2.1** (b).



(a) Gambar asli gelap (b) Gambar terang setelah perbaikan kontras

Gambar 2.1 Pengolahan Citra (Syuhada, 2015)

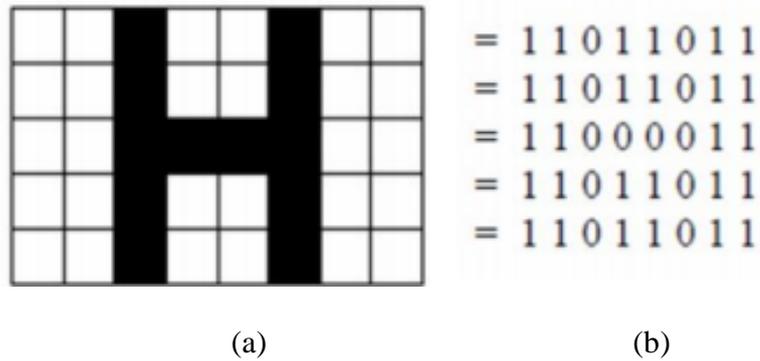
2.3 Jenis Jenis Citra

Nilai suatu piksel memiliki nilai dalam rentang tertentu, dari nilai minimum sampai nilai maksimum. Jangkauan yang digunakan berbeda-beda tergantung dari jenis warnanya. Namun secara umum untuk citra 8-bit jangkauannya adalah 0 – 255. Citra dengan penggambaran seperti ini digolongkan ke dalam citra integer. Berikut ini jenis-jenis citra berdasarkan nilai pikselnya (SAHTONI, 2012) :

1. Citra Biner (*Monokrom*)

Citra biner adalah citra digital yang hanya memiliki dua kemungkinan nilai piksel yaitu hitam dan putih. Citra biner juga disebut sebagai citra B&W (*black and white*) atau citra monokrom. Pada citra biner setiap titik

(piksel) bernilai 0 atau 1, masing-masing mempresentasikan warna tertentu. Contoh yang paling lazim, warna hitam bernilai 0 dan warna putih bernilai 1. Setiap titik (piksel) pada citra hanya membutuhkan media penyimpanan 1 bit, sehingga setiap byte dapat menampung informasi 8 titik (piksel)



Gambar 2.2 Citra biner dan representasinya dalam data digital (a) citra biner (b) nilai penyimpanan di memori (SAHTONI, 2012).



Gambar 2.3 Citra Biner (SAHTONI, 2012)

2. Citra Skala Keabuan (*GrayScale*)

Citra grayscale merupakan citra digital yang hanya memiliki satu nilai kanal pada setiap pikselnya, dengan kata lain nilai bagian red, green dan

blue memiliki warna yang sama, yaitu warna dari hitam, keabuan, dan putih. Nilai tersebut digunakan untuk menunjukkan tingkat intensitas. Tingkatan keabuan disini merupakan warna abu dengan berbagai tingkatan dari hitam hingga mendekati putih.



Gambar 2.4 Citra *Grayscale* (SAHTONI, 2012)

3. Citra Warna (*True Color*)

Pada citra warna, setiap titik mempunyai warna yang spesifik yang merupakan kombinasi dari 3 warna dasar, yaitu merah, hijau, dan biru. Format citra ini sering disebut sebagai citra RGB (red-green-blue). Setiap warna dasar mempunyai intensitas sendiri dengan nilai maksimum 255 (8 bit), dan warna minimum adalah putih. Red memiliki warna minimum putih dan warna maksimum merah. Green memiliki warna minimum putih dan warna maksimum hijau. Blue memiliki warna minimum putih dan warna maksimum biru.



Gambar 2.5 Citra warna (*true color*) (SAHTONI, 2012)

2.4 Konversi Citra

Dasar dari pengolahan citra adalah pengolahan warna RGB pada posisi tertentu. Bila membaca citra RGB dengan ukuran tinggi x lebar, akan menghasilkan matriks tiga dimensi dengan jumlah baris adalah tinggi dan jumlah kolom sejumlah tiga keping. Citra grayscale atau citra keabuan adalah citra yang hanya menggunakan warna pada tingkatan warna abu-abu. Warna abu-abu adalah satu satunya warna pada ruang RGB dengan komponen merah, hijau, dan biru mempunyai intensitas yang sama. Pada citra beraras keabuan hanya perlu menyatakan nilai intensitas untuk tiap piksel sebagai nilai tunggal, sedangkan pada citra berwarna perlu tiga nilai intensitas untuk setiap pikselnya. Berbeda dengan citra biner, citra grayscale disebut juga dengan citra 8-bit karena memiliki 28(256) kemungkinan nilai pada masing-masing pikselnya. Nilai tersebut dimulai dari nol untuk warna hitam dan 255 untuk warna putih (Putra, 2010).

Wijaya dan Tjiharjadi menjelaskan Proses awal yang sering dilakukan pada image processing adalah mengubah citra berwarna menjadi citra grayscale (Wijaya & Tjiharjadi, 2009). Hal ini dilakukan untuk menyederhanakan model

citra. Di dalam suatu gambar true color (RGB) terdapat tiga layer matriks, yaitu R-layer, G-layer, dan B-layer. Pada image processing dilakukan proses-proses terhadap ketiga layer tersebut, berarti dilakukan perhitungan yang sama pada setiap layer. Dengan demikian konsep grayscale adalah mengubah tiga layer tersebut menjadi satu layer matriks grayscale, yang menghasilkan satu citra grayscale. Di dalam citra ini tidak ada lagi warna, yang ada adalah derajat keabuan.

Dalam melakukan konversi citra dengan mengubah nilai piksel yang semula mempunyai 3 nilai yaitu R, G, B menjadi satu nilai keabuan atau biner. Untuk operasi konversi citra digunakan persamaan berikut (Putra, 2010) :

$$K_o = R + G + B \div 3 \dots \dots \dots (1)$$

Bisa juga dengan memberi bobot (w) pada RGB dikarenakan mata manusia lebih sensitif pada warna hijau, merah dan biru. Operasi tersebut dapat digunakan persamaan sebagai berikut (Putra, 2010):

$$K_o = w_r R_i + w_g G_i + w_b B_i \dots \dots \dots (2)$$

Di mana :

K_o : nilai keabuan pada piksel ke 0

w_r : bobot untuk elemen warna merah

w_g : bobot untuk elemen warna hijau

w_b : bobot untuk elemen warna biru

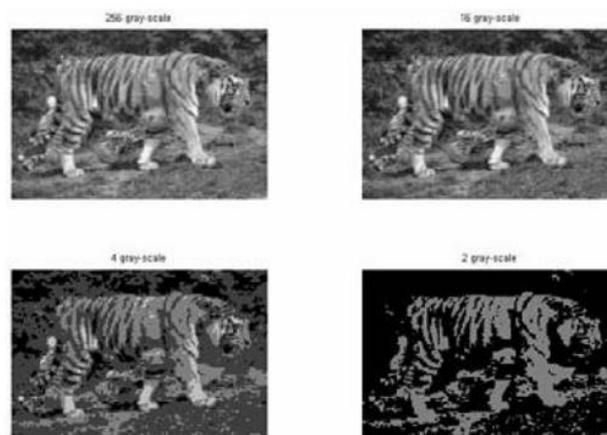
R_i : nilai intensitas elemen warna merah

B_i : nilai intensitas elemen warna biru

G_i : nilai intensitas elemen warna hijau

2.5 Thresholding

Thresholding merupakan sebuah metode yang dapat digunakan untuk mengubah sebuah gambar *grayscale* menjadi gambar biner, yaitu gambar yang hanya memiliki nilai 0 (hitam) atau 1 (putih) untuk setiap pikselnya. Pada proses *thresholding*, sebuah piksel dinyatakan sebagai “objek” atau *foreground* jika memiliki nilai derajat keabuan lebih besar dari batas (*threshold*) dan dinyatakan sebagai “latar” atau *background* jika nilai derajat keabuannya lebih kecil dari *threshold*. Pada *threshold* yang tinggi tidak dapat perbedaan karena keterbatasan mata manusia, pada **Gambar 2.6** terlihat perbedaan nilai 2, 4, 16, dan 256 sehingga akan tampak perbedaannya (Tahir, 2017).



Gambar 2.6 Proses *Threshold* beberapa tingkat (Tahir, 2017).

Nilai *threshold* yang optimal pada setiap gambar berbeda-beda. Nilai *threshold* pada gambar A tidak dapat digunakan pada gambar B, karena sebuah piksel yang harusnya dianggap objek menjadi dianggap latar akibat dari *threshold* yang tidak tepat. Terdapat beberapa metode untuk menentukan nilai *threshold* dari

suatu gambar, salah satunya adalah metode Otsu. Metode Otsu menghitung nilai *threshold* secara otomatis berdasarkan citra masukan. Pendekatan yang digunakan oleh metode Otsu adalah dengan melakukan analisis diskriminan yaitu menentukan suatu variabel yang dapat membedakan antara dua atau lebih kelompok yang muncul secara alami. Analisis diskriminan akan memaksimalkan variabel tersebut agar dapat memisahkan objek dengan latar belakang (Tahir, 2017).

2.6 Morfologi Citra

Morfologi di dunia digital dapat diartikan sebuah cara untuk mendeskripsikan ataupun menganalisa bentuk dari objek digital. Morfologi dalam citra digital adalah suatu *tool* untuk ekstraksi komponen *image* yang berguna dalam representasi dan deskripsi dari bentuk daerah (*region shape*) dengan *structuring element* (SE) untuk menentukan *properties of interest* dari *image* (Putra, 2010).

Pada morfologi, suatu citra dinyatakan dengan himpunan koordinat diskrit (kontinu). Dalam hal ini, himpunan tersebut berhubungan dengan *point* atau piksel objek pada citra. Karena objek dianggap sebagai suatu himpunan maka operasi-operasi himpunan seperti gabungan (*union*), irisan (*intersection*), dan komplemen (*complement*) dapat dilakukan (Putra, 2010).

Operasi morfologi menggunakan dua input himpunan yaitu suatu citra (pada umumnya citra biner) dan suatu kernel. Khusus dalam morfologi, istilah kernel biasa disebut *structuring elements* (elemen pembentuk struktur). *SE* merupakan suatu matrik yang pada umumnya berukuran kecil. Elemen dari *SE*

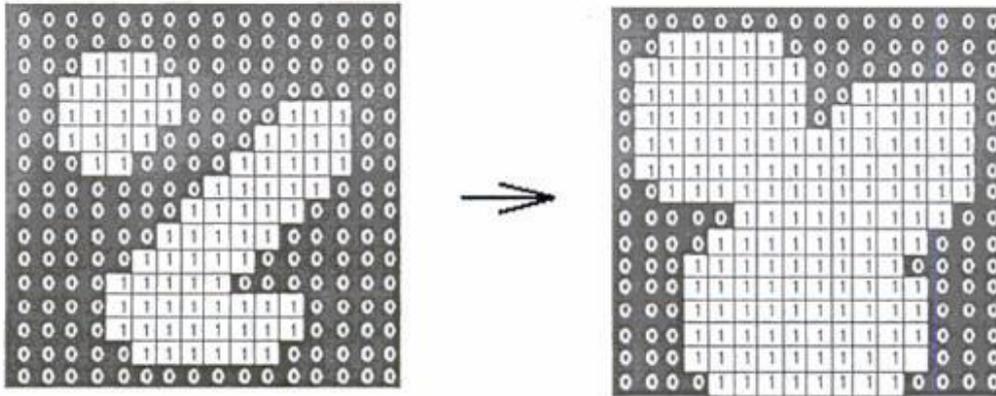
dapat bernilai 1, 0, dan *don't care*. Nilai *don't care* biasanya ditandai dengan nilai elemen dikosongkan atau diberi tanda silang. Terdapat dua operasi dasar morfologi yaitu dilasi (*dilation*) dan erosi (*erosion*). Operasi-operasi ini menjadi dasar untuk membuat berbagai operasi morfologi yang sangat berguna untuk pengolahan citra digital, seperti *opening*, *closing*, *hit and miss transform*, *thinning*, dan *thickening* (Putra, 2010).

Jika suatu objek (citra input) dinyatakan dengan A dan SE dinyatakan dengan B serta Bx menyatakan translasi B sedemikian sehingga pusat B terletak pada x . Operasi dilasi A dan B dapat dinyatakan sebagai berikut.

$$(A, B) = A \oplus B = \{x: Bx \cap A \neq \emptyset\} \dots \dots \dots (3)$$

Dengan \emptyset menyatakan himpunan kosong.

Dilasi (*dilation*) berguna untuk memperluas atau menebalkan objek pada *image* biner. Proses dilasi dilakukan dengan membandingkan setiap piksel citra input dengan nilai pusat SE dengan cara melapiskan (*superimpose*) SE dengan citra sehingga pusat SE tepat dengan posisi piksel citra yang diproses. Jika paling sedikit ada 1 piksel pada SE sama dengan nilai piksel objek (*foreground*) citra maka piksel input diset nilainya dengan nilai piksel *foreground* dan bila semua piksel yang berhubungan adalah *background* maka input piksel diberi nilai piksel *background* (Putra, 2010).



Gambar 2.7 Proses dilasi dengan SE berukuran 3×3 dengan semua elemen SE bernilai 1 (Putra, 2010)

Semakin besar ukuran SE maka semakin besar perubahan yang terjadi. SE berukuran kecil juga dapat memberikan hasil yang sama dengan SE berukuran besar dengan cara melakukan dilasi berulang kali. Efek dilasi terhadap citra biner adalah memperbesar batas dari objek yang ada sehingga objek terlihat semakin besar dan lubang-lubang yang terdapat di tengah objek akan tampak mengecil (Putra, 2010).

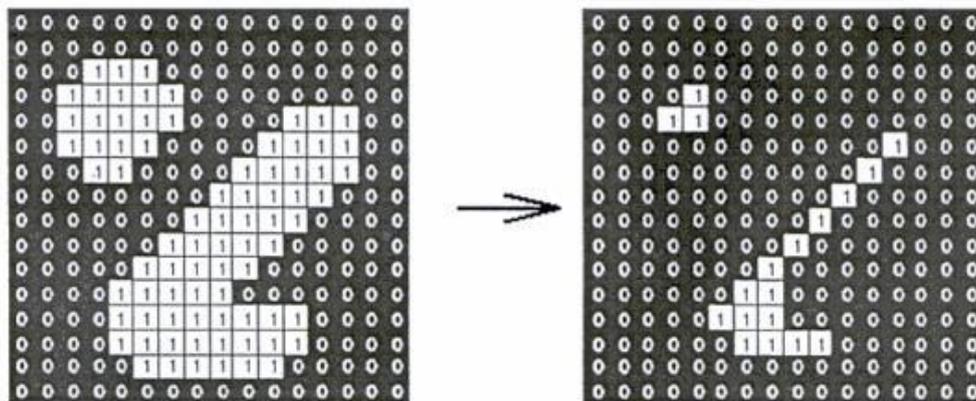
Secara matematis, operasi erosi dapat dinyatakan sebagai berikut.

$$(A, B) = A \ominus B = \{x: Bx \subset X\} \dots \dots \dots (4)$$

Sama seperti dilasi, proses erosi dilakukan dengan membandingkan setiap piksel citra input dengan nilai pusat SE dengan cara melapiskan SE dengan citra sehingga pusat SE tepat dengan posisi piksel citra yang diproses. Jika semua piksel pada SE tepat sama dengan semua nilai piksel objek (*foreground*) citra maka piksel input diset nilainya dengan nilai piksel *foreground*, jika tidak maka

input piksel diberi nilai piksel *background*. Proses serupa dilanjutkan dengan menggerakkan *SE* piksel demi piksel pada citra input (Putra, 2010).

Proses erosi merupakan kebalikkan dari proses dilasi. Jika dalam proses dilasi menghasilkan objek yang lebih luas maka dalam proses erosi akan menghasilkan objek yang menyempit (mengecil). Lubang pada objek juga akan membesar seiring menyempitnya objek tersebut (Putra, 2010).



Gambar 2.8 Proses erosi dengan *SE* berukuran 3×3 dengan semua elemen *SE* bernilai 1. (Putra, 2010)

Dari **Gambar 2.8** terlihat hasil proses erosi menyebabkan objek mengecil. Semakin besar kernel yang digunakan maka hasil yang akan didapatkan akan semakin kecil. Begitu pula juga apabila proses erosi dilakukan berulang-ulang akan terus mengecilkan objek walaupun hanya menggunakan *SE* berukuran kecil (Putra, 2010).

Di samping itu operasi morfologi *grayscale*, terdapat juga algoritma dasar morfologi *grayscale* adalah *Top-Hat Transform*. Teknik dengan menggunakan *Top Hat Transform* telah banyak digunakan dalam penelitian *image processing*,

khususnya pada penelitian yang memiliki pencahayaan tidak teratur. Teknik *Top Hat Transform* ini mengombinasikan pengurangan citra *grayscale* dengan *opening* (Putra,2010).

- *Top Hat Transform*

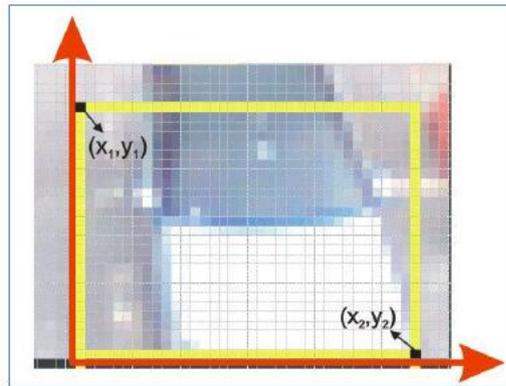
Transformasi *top hat* merupakan perbedaan antara citra masukan (citra *grayscale*) dan citra setelah mengalami operasi *opening* (Solomon dan Breckon, 2011). Secara matematis seperti berikut :

$$T_{hat}(f) = f - (f \circ b) \dots \dots \dots (5)$$

Pada rumus diatas, f adalah citra masukan (citra *grayscale*) dan $(f \circ b)$ adalah citra hasil operasi *opening*. Transformasi ini berguna untuk mendapatkan bentuk global suatu objek yang mempunyai intensitas yang bervariasi.

2.7 Analisis Blob

Analisis blob merupakan teknik yang digunakan untuk menyatakan luas area piksel dari suatu *image* yang menjadi fokus deteksi. Untuk menentukan nilai Blob, ada beberapa hal yang harus diketahui untuk menghasilkan sebuah blob yang optimal. Penentuan luas blob pada setiap objek pada proses segmentasi *foreground* perlu dianalisis karena nilai blob pada tiap objek akan berbeda. Hal ini dipengaruhi oleh fitur objek seperti ukuran, jenis, dan teknik pengambilan data video. Prosesnya dimulai dari penandaan area *foreground* yang dianggap objek, kemudian pengumpulan data area menjadi blob seperti posisi piksel awal, panjang terhadap sumbu x dan sumbu y, dan luas area piksel (Basri, 2015).



Gambar 2.9 Analisis Luasan Blob Sebuah Objek Kendaraan (Basri, 2015).

2.8 Tracking

Proses mencari objek bergerak dalam urutan *frame* yang dikenal sebagai pelacakan (*tracking*). Pelacakan ini dapat dilakukan dengan menggunakan ekstraksi ciri benda dan mendeteksi objek atau benda bergerak di urutan *frame*. Dengan menggunakan nilai posisi objek di setiap *frame*, bisa digunakan untuk menghitung posisi dan kecepatan objek bergerak. Tujuan dari proses *tracking* adalah untuk mengetahui objek yang sama pada urutan *frame* (D. W. Wicaksono, 2017)

2.9 Kalman Filter

Pengertian umum dari *Kalman Filter* adalah solusi rekursif yang menggunakan teknik asimilasi data. Asimilasi data adalah salah satu teknik estimasi yang banyak digunakan untuk menyelesaikan berbagai masalah sistem dinamik (M. A. R. Wicaksono & L. Lasmadi, 2020)

Kalman Filter merupakan suatu algoritma yang menggabungkan model dan pengukuran. Data pengukuran terbaru menjadi bagian penting dari algoritma *Kalman*

Filter karena data terakhir akan mengoreksi hasil prediksi pengukuran, sehingga hasil estimasi selalu mendekati kondisi yang sebenarnya (M. A. R. Wicaksono & L. Lasmadi, 2020)

Kalman Filter mengestimasi proses dengan menggunakan kontrol umpan balik dalam bentuk pengukuran *noise*. Dengan demikian, persamaan untuk *Kalman Filter* dibagi menjadi dua kelompok: persamaan *update* waktu dan persamaan *update* pengukuran. Persamaan *update* waktu bertanggung jawab untuk memproyeksikan kedepan (waktu). keadaan saat ini dan estimasi kovarian *error* mengestimasi untuk mendapatkan sebuah estimasi apriori untuk langkah waktu berikutnya. Persamaan *update* pengukuran bertanggung jawab atas umpan balik untuk menggabungkan pengukuran baru ke dalam estimasi apriori untuk mendapatkan estimasi aposteriori yang lebih baik (M. A. R. Wicaksono & L. Lasmadi, 2020)

Kalman Filter terdiri dari dua tahap yaitu (M. A. R. Wicaksono & L. Lasmadi, 2020) :

1. Tahap prediksi

Tahap prediksi dipengaruhi oleh dinamika sistem dengan memprediksi variabel keadaan dengan menggunakan persamaan estimasi variabel keadaan dan tingkat akurasi dihitung menggunakan kovarian *error* (M. A. R. Wicaksono & L. Lasmadi, 2020).

2. Tahap koreksi

Pada tahap koreksi hasil estimasi variabel keadaan yang diperoleh pada tahap prediksi dikoreksi menggunakan model pengukuran. Salah satu bagian dari tahap ini yaitu matriks *Kalman Filter* yang digunakan untuk

meminimumkan kovarian *error* (M. A. R. Wicaksono & L. Lasmadi, 2020).

2.10 Gaussians Mixture Model (GMM)

GMM merupakan fungsi *density* probabilitas parametrik yang direpresentasikan sebagai jumlah komponen *gaussian*. GMM umumnya digunakan sebagai model parametrik dari distribusi probabilitas suatu pengukuran kontinyu atau fitur dari suatu sistem biometrik, seperti pelacakan berdasarkan warna suatu objek dalam video sedangkan pada citra RGB digunakan nilai vektornya. Penggunaan model yang lebih banyak pada setiap piksel akan menyebabkan proses *background extraction* lebih bersifat adaptif karena lebih banyak komponen warna yang dapat dimodelkan dalam setiap pikselnya. Jika suatu piksel tidak cocok dengan semua model distribusi *Gaussian*, maka model *Gaussian* yang memiliki probabilitas terkecil akan dihapus dan digantikan dengan model *Gaussian* untuk warna piksel yang baru. Langkah selanjutnya adalah menentukan piksel mana yang termasuk dalam objek *background* dan *foreground*, maka dilakukan seleksi. Jika warna piksel masuk dalam kategori salah satu kandidat background, maka piksel tersebut akan dianggap sebagai *background* (piksel diberi nilai 0/warna hitam). Di luar itu, piksel yang tidak termasuk dalam kategori *background* akan dianggap sebagai *foreground* (piksel diberi nilai 1/putih) (Indrabayu dkk, 2015).

Background adalah objek yang tidak bergerak di dalam *frame* video. Sedangkan *foreground* adalah objek bergerak dalam *frame* video. Objek yang terdeteksi sebagai *foreground* ditandai dengan blob yang merupakan sekumpulan piksel yang memiliki hubungan tetangga. Akurasi sistem deteksi bisa ditingkatkan

dengan melakukan optimalisasi parameter blob dan penentuan *Region of Interest* (ROI) objek yang dideteksi (Indrabayu dkk, 2015).

2.11 Local Binary Pattern (LBP)

Local Binary Pattern (LBP) merupakan suatu operasi gambar yang mentransformasikan sebuah citra menjadi sebuah susunan label *integer* yang menggambarkan kenampakan skala kecil dari suatu citra. Label atau statistik tersebut, biasanya merupakan sebuah histogram, kemudian digunakan lagi untuk analisis citra yang lebih lanjut (Afrizal Firdaus, 2016).

Tahapan dalam mendapatkan nilai *LBP* adalah dengan mendapatkan nilai transformasi biner yang akan disusun menjadi desimal. Jika intensitas tetangga yang dibandingkan lebih besar atau sama dengan intensitas pusatnya maka nilai transformasi binernya adalah 1. Sebaliknya, jika intensitas tetangga yang dibandingkannya lebih kecil daripada intensitas pusatnya maka nilai transformasi binernya adalah 0. Setelah ke-8 tetangganya dibandingkan, nilai transformasi tersebut akan disusun dan dikonversikan ke dalam bentuk decimal (Afrizal Firdaus, 2016).

6	5	2
7	6	1
9	8	7

(a)

1	0	0
1		0
1	1	1

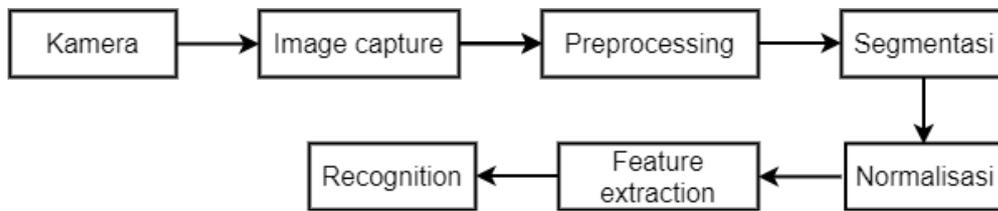
(b)

Gambar 2.10 Contoh Implementasi *LBP* (a) Nilai Pikel pada Citra, (b) Nilai *LBP* dari Citra (a) (Afrizal Firdaus, 2016).

2.12 *Optical Character Recognition (OCR)*

Optical Character Recognition (OCR) adalah sebuah sistem komputer yang dapat membaca huruf, baik yang berasal dari sebuah pencetak (printer atau mesin ketik) maupun yang berasal dari tulisan tangan. OCR adalah aplikasi yang menerjemahkan gambar karakter (*image character*) menjadi bentuk teks dengan cara menyesuaikan pola karakter per baris dengan pola yang telah tersimpan dalam database aplikasi. Hasil dari proses OCR adalah berupa teks sesuai dengan gambar output scanner di mana tingkat keakuratan penerjemahan karakter tergantung dari tingkat kejelasan gambar dan metode yang digunakan (Trilaksono & Rizal, 2008).

Secara umum blok diagram kerja OCR dapat dilihat pada **Gambar 2.11** berikut.



Gambar 2.11 Blok Diagram Kerja OCR (Trilaksono & Rizal, 2008).

1. Kamera

sebagai perekam citra pelat nomor kendaraan roda dua, kemudian disimpan kedalam folder yang telah disiapkan, format untuk gambar berupa *.jpg dan *.png.

2. *Image capture*

adalah hasil dari pengambilan gambar dari kamera

3. *Preprocessing*

adalah proses dimana untuk menghilangkan bagian dari citra yang diperlukan untuk proses selanjutnya.

4. Segmentasi

berfungsi untuk membedakan atau memisahkan area yang diamati pada setiap karakter pelat kendaraan yang di deteksi.

5. Normalisasi

bertujuan untuk merubah dimensi citra hasil pemrosesan dan merubah ketebalan dari pemisah citra.

6. *Feature extraction*

merupakan proses untuk mengamati sebuah ciri-ciri tertentu dari citra

7. *Recognition*

Adalah proses untuk mengenali sebuah karakter yang telah diproses, dengan cara membandingkan citra hasil proses dengan karakter angka dan huruf yang terdapat pada sistem.

2.13 *License Plate Recognition*

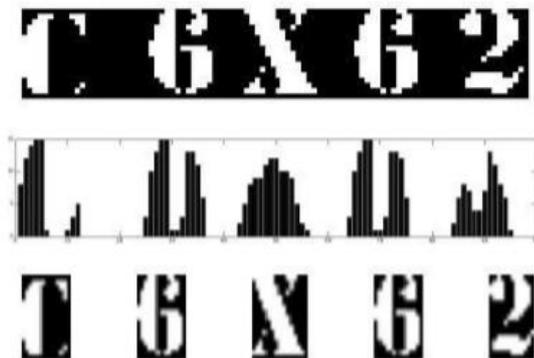
License Plate Recognition (LPR) merupakan salah satu teknologi yang digunakan sebagai implementasi dari *Automatic Vehicle Identification*. LPR diimplementasikan dengan memanfaatkan teknologi pengolahan citra untuk mengenali kendaraan dari sebuah kendaraan hanya dari pelat nomornya saja. Teknologi ini sudah banyak digunakan diberbagai aplikasi keamanan dan lalu lintas. Masalah-masalah yang sering digunakan untuk implementasi teknologi ini adalah sistem parkir, pintu tol *Traffic Management System*, dan keamanan lalu lintas. Terdapat tiga tahapan penting dalam *License Plate Recognition*, yaitu *Region Plate Extraction*, *Segmentation*, dan *Character Recognition* (Basri, 2015).

2.14 *Region Plate Extraction*

Region Plate Extraction atau juga dikenal sebagai *License Plate Localization* (LPL) merupakan tahapan dari LPR untuk mengesktrak atau memisahkan pelat nomor dari sebuah kendaraan. Tujuan dari tahap ini adalah untuk mengetahui dimana letak dan posisi sebuah pelat nomor dari sebuah kendaraan yang selanjutnya pelat nomor yang telah terekstrak akan dipakai untuk proses selanjutnya yaitu *segmentation* (Basri, 2015).

2.15 Segmentasi Pelat

Segmentasi didapat potongan pelat nomor dari proses sebelumnya, dalam tahap ini potongan pelat nomor dipecah-pecah sehingga nantinya didapatkan potongan-potongan citra per karakter. Metode yang sering digunakan dalam segmentasi karakter termasuk segmentasi karakter dalam kasus lain selain pelat nomor adalah proyeksi horizontal dan proyeksi vertikal. Dalam metode ini, dihitung nilai kemunculan piksel karakter secara horizontal maupun vertikal. Di mana baris ataupun kolom dengan kemunculan piksel karakter di atas 0 atau nilai *threshold* tertentu akan diidentifikasi sebagai karakter, sedangkan sisanya akan dihilangkan (Basri, 2015).

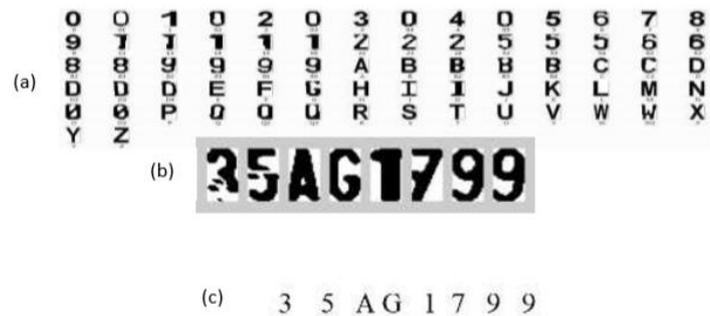


Gambar 2.12 Proses segmentasi karakter secara vertikal (Basri, 2015).

2.16 Character Recognition

Setelah didapatkan potongan-potongan karakter dari tahap segmentasi, selanjutnya potongan-potongan karakter itu akan dikenali sebagai karakter tertentu. Ada banyak metode yang dapat digunakan untuk mengenali karakter yang sudah disegmentasi (Basri, 2015).

Pengenalan karakter dilakukan dengan membandingkan karakter yang sesuai dengan standar karakter pelat Indonesia dan karakter pada teks yang telah disegmentasi.



Gambar 2.13 (a) Database Karakter. (b) Segmentasi Karakter. (c) Pengenalan Karakter (Basri, 2015).

2.17 Penelitian Terkait

Berikut ini merupakan beberapa penelitian terkait dengan penelitian yang dilakukan:

1. *Optical Character Recognition* Untuk deteksi Pelat Mobil dan Motor Kendaraan pada Kampus Teknik Gowa (Tahir, 2017)

Yusran Mansyur pada tahun 2017 melakukan penelitian deteksi pelat mobil dan motor dari arah belakang kendaraan, pengambilan data video menggunakan kamera Nikon Coolpix P610 dengan bantuan tripod. Pada penelitian ini menggunakan empat kondisi skenario yaitu dengan tinggi tripod 2,3 meter menghasilkan akurasi 23.5%. 2,5 meter menghasilkan akurasi 56.7%. 3 meter menghasilkan akurasi 80.9% dan 4 meter menghasilkan akurasi 44.8%. Penelitian ini menggunakan metode *Local*

Binary Pattern untuk mendeteksi wilayah pelat dan menggunakan *Optical Character Recognition* untuk mengenali teks pelat.

2. *Vehicle License Plate Localization based on Local Binary Pattern Features* (Ascar Davix dkk, 2019)

Penelitian ini menggunakan 450 gambar pelat mobil yang di foto dari arah belakang. Data yang di ambil kemudian dilatih menggunakan *Cascaded Adaboost classifier* dengan ekstraksi fitur *Local Binary Pattern* (LPB) untuk mendeteksi wilayah pelat. Ada 600 gambar yang telah diuji pada sistem dan menghasilkan tingkat deteksi 96.8%.

3. *Automatic License Plate Recognition Using Local Binary Pattern and Histogram Matching* (Ashutosh Kumar Bachechan dkk, 2017)

Penelitian ini membuat sistem deteksi dan mengenali pelat nomor kendaraan secara real time. Data direkam menggunakan kamera Moto G 3G (13 megapixel) dengan spesifikasi, *frame rate* = 30, tinggi = 1080, lebar = 1920, Video Format = MP4, *pan angle* = 7, 10 degree dan *tilt angle* = 0, 3°. Kamera dipasang pada ketinggian 800 mm. Kendaraan yang dideteksi yaitu mobil dengan melakukan pendeteksian dari arah depan kendaraan. Metode yang digunakan yaitu *Local Binary Pattern* (LBP) dan *Histogram Matching* untuk mendeteksi pelat nomor. Untuk pengenalan karakter menggunakan *KNN classifier*. Sistem ini diuji dengan 300 gambar menghasilkan akurasi deteksi 96.14% dan akurasi rekognisi 89.35%.

4. Rekognisi Pelat Kendaraan Bermotor dalam Keadaan Bergerak (Rahmiyanti Rusli, 2019)

Penelitian ini dilakukan menggunakan data video untuk mendeteksi dan mengenali teks pelat. pengambilan data menggunakan kamera Nikon Coolpix P610 dengan tinggi kamera 8,75 meter. Sistem ini menggunakan metode *Gaussians Mixture Models* untuk mendeteksi kendaraan, *Local Binary Pattern* untuk mendeteksi posisi pelat dan *Optical Character Recognition* untuk mengenali karakter. Data yang digunakan adalah data riil maka penelitian ini menggunakan metode *Top Hat Transform*, *Bottom Hat Transform*, *Top Bottom Hat Transform* sebagai *Image Enhancement* untuk memproses citra pelat untuk memudahkan sistem dalam mengenali teks. Tiga metode ini menghasilkan akurasi 83%, 50% dan 70%. Sistem yang dibangun pada penelitian ini mendeteksi kendaraan dengan single detection pada kecepatan 20 km/jam, 30 km/jam dan 40 km/jam.

5. Deteksi Plat Nomor Kendaraan dengan Menggunakan Metode *Hough Transform* dan *Support Vector Machine* (Nurcahyo dkk, 2020)

Penelitian ini menggunakan metode *Hough Transform* untuk deteksi lokasi pelat nomor dan *Support Vector Machine* (SVM) untuk pengenalan karakter. Data citra melalui *preprocessing* yaitu terdiri dari pembacaan citra, mengubah citra RGB menjadi *Grayscale* dan *filtering*. Setelah *preprocessing* citra masuk ke dalam tahap penggunaan metode *Hough Transform* yang terdiri dari transformasi ke parameter ruang dan mendeteksi kandidat garis horizontal pada pelat. kemudian dilakukan tahap *thresholding* untuk mencocokkan kandidat garis horizontal pada

pelat. Hasil dari *thresholding* akan langsung di proses ke dalam metode SVM untuk dilakukan pendeteksian nomor pelat. Proses pada tahap SVM yang didalamnya meliputi proses pengklasifikasian karakter dan pengenalan pada tiap karakter yang terdeteksi. Hasil dari pengenalan karakter nantinya akan berbentuk teks pada. Presentase rata-rata akurasi hasil dari pengujian *metode transformasi hough* dan SVM adalah 31% dan 56%. Dengan pengujian data uji sebanyak 52 citra pada kedua metode dengan hasil nilai rata-rata recall dengan nilai 1 dengan rata-rata nilai presisi 0,2 dan 0,11.