



DISERTASI TE-143597

SISTEM PELACAKAN DAN PENGENALAN PELAT NOMOR KENDARAAN BERBASIS VIDEO MENGGUNAKAN HYBRID-CNN MEAN SHIFT

**BASUKI RAHMAT
07111160010002**

DOSEN PEMBIMBING
Prof. Dr. Ir. Mauridhi Hery Purnomo, M.Eng.
Dr. I Ketut Eddy Purnama, S.T., M.T.

PROGRAM DOKTOR
DEPARTEMEN TEKNIK ELEKTRO
FAKULTAS TEKNOLOGI ELEKTRO
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2018

Halaman ini sengaja dikosongkan

LEMBAR PENGESAHAN

SISTEM PELACAKAN DAN PENGENALAN PELAT NOMOR KENDARAAN BERBASIS VIDEO MENGGUNAKAN HYBRID-CNN MEAN SHIFT

Disertasi ini disusun untuk memenuhi salah satu syarat memperoleh gelar Doktor
di Institut Teknologi Sepuluh Nopember Surabaya

Oleh :

Basuki Rahmat (NRP. 07111160010002)

Tanggal Ujian : 16 Oktober 2017

Periode Wisuda : Maret 2018

Prof. Dr. Ir. Mauridhi Hery Purnomo, M.Eng
NIP. 19580916 198601 1 001

(Pembimbing I)

Dr. I Ketut Eddy Purnama, S.T., M.T.
NIP. 19690730 199512 1 001

(Pembimbing II)

Mochamad Hariadi, S.T., M.Sc., Ph.D.
NIP. 19691209 199703 1 002

(Penguji I)

Dr. Supeno Mardi Susiki Nugroho, ST., M.T.
NIP. 19700313 199512 1 001

(Penguji II)

Ir. Endra Joelianto, Ph.D.
NIP. 19660708 199203 1 003

(Penguji III)



Dr. Tri Arief Sardjono, ST., MT.
NIP. 19700212 199512 1001

Halaman ini sengaja dikosongkan

SURAT PERNYATAAN KEASLIAN DISERTASI

Yang bertandatangan di bawah ini:

Nama : Basuki Rahmat
Program Studi : Teknik Multimedia dan Jaringan
NRP : 07111160010002

dengan ini menyatakan bahwa isi sebagian maupun keseluruhan disertasi dengan judul:

SISTEM PELACAKAN DAN PENGENALAN PELAT NOMOR KENDARAAN BERBASIS VIDEO MENGGUNAKAN HYBRID-CNN MEAN SHIFT

adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diizinkan 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, 12 Januari 2018

Yang Membuat Pernyataan,

Basuki Rahmat

NRP. 07111160010002

Halaman ini sengaja dikosongkan

SISTEM PELACAKAN DAN PENGENALAN PELAT NOMOR KENDARAAN BERBASIS VIDEO MENGGUNAKAN HYBRID-CNN MEAN SHIFT

Nama	:	Basuki Rahmat
NRP	:	07111160010002
Promotor	:	Prof. Dr. Ir. Mauridhi Hery Purnomo, M.Eng.
Co-Promotor	:	Dr. I Ketut Eddy Purnama, S.T., M.T.

ABSTRAK

Penelitian ini mengusulkan metode baru dalam sistem pelacakan dan pengenalan pelat nomor kendaraan berbasis video secara cerdas. Tiga bagian besar yang diteliti, yaitu: bagian ekstraksi pelat nomor kendaraan, bagian pelacakan pelat nomor kendaraan sepanjang *frame* video, dan bagian ekstraksi karakter, yang di dalamnya terdapat unit segmentasi karakter dan pengenalan karakter pelat nomor kendaraan.

Pada bagian ekstraksi pelat nomor kendaraan digunakan metode *Smearing Algorithm*. Pada bagian pelacakan pelat nomor kendaraan digunakan metode Basuki-I Ketut-Mauridhi (BIM) Mean Shift (BIM-Mean Shift) dengan *Switching Kernel* dan *Adaptive Fuzzy Gaussian Kernel*. Pada bagian ekstraksi karakter, yang terdiri dari unit segmentasi karakter pelat nomor kendaraan digunakan metode Basuki-I Ketut-Mauridhi (BIM) *Hybrid Cellular Neural Network* (BIM-HCNN), dan unit pengenalan karakter pelat nomor kendaraan digunakan metode *Extreme Learning Machine* (ELM).

Dari hasil serangkaian percobaan simulasi menunjukkan hasil-hasil sebagai berikut. *Smearing Algorithm* dapat digunakan sebagai salah satu metode ekstraksi pelat nomor kendaraan. Pelacakan pelat nomor kendaraan dengan menggunakan metode BIM-Mean Shift dengan *Switching Kernel* dan *Adaptive Fuzzy Gaussian Kernel* menghasilkan kinerja yang lebih baik daripada Mean Shift standar. Ekstraksi karakter pelat nomor kendaraan, menggunakan metode BIM-HCNN dengan Fuzzy Adaptif dan Neuro-Fuzzy memberikan hasil segmentasi yang lebih unggul dibandingkan CNN standar. Serta pengenalan Karakter Pelat Nomor Kendaraan dengan ELM dari hasil penelitian ini mencapai kinerja yang sama baik dengan metode *template matching*.

Kata kunci : *Smearing Algorithm*, BIM-Mean Shift, CNN, BIM-HCNN, ELM.

Halaman ini sengaja dikosongkan

VIDEO-BASED VEHICLE LICENSE PLATE TRACKING AND RECOGNITION SYSTEM USING HYBRID-CNN MEAN SHIFT

Name : Basuki Rahmat
Student ID : 07111160010002
Promotor : Prof. Ir. Dr. Mauridhi Hery Purnomo, M.Eng.
Co-Promotor 1 : Dr. I Ketut Eddy Purnama, S.T., M.T.

ABSTRACT

This research proposes a new method for video-based vehicle license plate tracking and recognition system intelligently. Three major sections were researched: the vehicle license plate extraction, the vehicle license plate tracking along the video frame, and the vehicle license plate character extraction, in which there are the license plate character segmentation unit and the license plate character recognition unit.

In the vehicle license plate extraction section is by using Smearing Algorithm method. In the vehicle license plate tracking section, using Basuki-I Ketut-Mauridhi (BIM) Mean Shift (BIM-Mean Shift) method with Switching Kernel and Adaptive Fuzzy Gaussian Kernel. In the character extraction section, which consists of the vehicle license plate character segmentation unit, using Basuki-I Ketut-Mauridhi (BIM) Hybrid Cellular Neural Network (BIM-HCNN) method, and the vehicle license plate character recognition unit, using Extreme Learning Machine (ELM) method.

From the results of a series of simulation experiments show the following results. Smearing Algorithm can be used as a method of the vehicle license plates extraction. The vehicle license plates tracking using the BIM-Mean Shift methods with Switching Kernel and Adaptive Fuzzy Gaussian Kernel result in better performance than the standard Mean Shift. The vehicle license plate characters extraction, using the BIM-HCNN method with Adaptive Fuzzy and Neuro-Fuzzy gives superior segmentation results compared to the standard CNN. As well as the vehicle license plates character recognition using ELM from the results of this research achieved the same performance with the template matching method.

Keywords: Smearing Algorithm, BIM-Mean Shift, CNN, BIM-HCNN, ELM.

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Puji syukur Alhamdulillah senantiasa penulis panjatkan kehadirat Allah SWT yang telah memberikan Rahmat, karunia, berkah dan hidayah-Nya, dan shalawat serta salam senantiasa tercurahkan kepada junjungan kita Rasulullah SAW, hingga terselesaikannya penulisan disertasi yang berjudul "Sistem Pelacakan dan Pengenalan Pelat Nomor Kendaraan Berbasis Video Menggunakan Hybrid-CNN Mean Shift". Disertasi ini disusun untuk memenuhi salah satu syarat akademik Program Doktor Jurusan Teknik Elektro, Fakultas Teknologi Elektro, Institut Teknologi Sepuluh Nopember Surabaya.

Banyak tantangan dan hambatan yang penulis hadapi dalam penulisan disertasi ini. Alhamdulillah atas pertolongan Allah SWT dan bantuan dari berbagai pihak akhirnya penulisan ini dapat kami selesaikan. Pada kesempatan ini, penulis menyampaikan terimakasih yang sebesar-besarnya kepada:

1. Bapak Prof. Dr. Ir. Mauridhi Hery Purnomo, M.Eng. selaku pembimbing dan Bapak Dr. I Ketut Eddy Purnama, S.T., M.T. selaku co-pembimbing yang telah dengan sabar memberikan arahan, pembimbingan dan semangat dalam menempuh studi Program Doktor di Teknik Elektro ITS Surabaya. Beliau-beliau telah membimbing kami hingga dapat mempublikasikan hasil penelitian kami di Seminar Internasional dan Jurnal Internasional terindeks Scopus dan memberikan berbagai solusi yang sangat bermanfaat dalam pengembangan hasil penelitian dan amanat Tri Dharma Perguruan Tinggi. Beliau telah membimbing penulis hingga mendapatkan berbagai hibah riset yang mendukung kegiatan studi dan mendapatkan kesempatan mengikuti IEEE International Conference on Computer Science and Automation Engineering (CSAE 2013), 01-03 November 2013, di Guangzhou, China, dan Joint Workshop for Global Engineers in Asia (JWGE 2015), 25-28 July 2015, di Ritsumeikan University Biwako Kusatsu Campus Japan. Serta mendapatkan Letter of Invitation sebagai Letter of Acceptance (LOA) dari Prof. Bart Verkerke, Professor in Medical Product Design Technical-Scientific director SPRINT, University Medical Center Groningen Netherlands.

2. Bapak Mochamad Hariadi, ST., M.Sc., Ph.D., Bapak Dr. Supeno Mardi Susiki Nugroho, ST.,M.T., dan Bapak Ir. Endra Joelianto, Ph.D., selaku penguji yang telah memberikan masukan dan kritikan yang sangat bermanfaat dalam membantu penulis untuk mencapai tujuan, manfaat, kontribusi dan orisinalitas dari penelitian disertasi ini.
3. Bapak Prof. Dr. Ir. Yoyon Kusnendar Suprapto, M.Sc, Bapak Dr. Ir. Wirawan, DEA, Dr. Surya Sumpeno, ST., M.Sc., Dr. Ir. Margo Pujiantara, MT., Dr. Eng Ardyono Priyadi, S.T., M.Eng., Dr. Vita Lystianingrum Budiharto Putri, ST., M.Sc., Prof. Dr. Ir. Gamantyo Hendrantoro, Ph.D., Dr. Eko Mulyanto Yuniarno, ST., MT., Dr. Ir. Endroyono, DEA, dan Bapak/Ibu Dosen Teknik Elektro yang lain yang tidak bisa disebutkan satu per satu, yang telah memberikan masukan dan kritikan, baik pada saat progress kemajuan S3 maupun pada saat revisi Hasil Rapat Komisi Pertimbangan Fakultas (KPF) yang sangat bermanfaat dalam perbaikan dan orisinalitas dari penelitian disertasi ini.
4. Bapak Prof. Ir. Djauhar Manfaat, M.Sc, Ph.D (alm) dan Prof. Dr. Ir. Tri Widjaja, M.Eng., selaku Direktur dan Asisten Direktur Program Pascasarjana Institut Teknologi Sepuluh Nopember Surabaya.
5. Bapak Dr. Tri Arief Sardjono, ST., MT., selaku Dekan Fakultas Teknologi Elektro Institut Teknologi Sepuluh Nopember Surabaya.
6. Bapak Dr. Eng Ardyono Priyadi, S.T., M.Eng. dan Bapak Ir. Djoko Purwanto, M.Eng., Ph.D. selaku Kepala dan Sekretaris Departemen Teknik Elektro Institut Teknologi Sepuluh Nopember Surabaya.
7. Bapak Dr. Ir. Wirawan, DEA, selaku Koordinator Program Pasca Sarjana Teknik Elektro.
8. Bapak Dr. Rony Seto, ST., MT, selaku Sekretaris Program Doktor Teknik Elektro Institut Teknologi Sepuluh Nopember Surabaya.
9. Segenap pengelola Program Pascasarjana (PPs) ITS, dosen dan karyawan PP_s Jurusan Teknik Elektro ITS, Mbak Rany, Bapak Hartono, Bapak Supari serta Staff Administrasi lainnya yang telah memberikan banyak

kelancaran dan bantuannya selama kami menempuh program pendidikan ini.

10. Beasiswa Program Pasca Sarjana (BPPS) DIKTI (2011-2013), Perpanjangan Beasiswa Program Pasca Sarjana (BPPS) DIKTI (2014), Beasiswa Joint Workshop for Global Engineers in Asia (JWGE 2015), 25-28 July 2015, di Ritsumeikan University Biwako Kusatsu Campus Japan, Hibah Penelitian KKP3T Litbang Deptan (2011-2012), Hibah Penelitian Unggulan Perguruan Tinggi (2015-2016), Hibah Penelitian Disertasi Doktor (2016), dan Universitas Pembangunan Nasional “Veteran” Jawa Timur yang telah memberikan ijin Tugas Belajar dan bantuan Tunjangan Studi selama menempuh program studi doktor di ITS Surabaya.
11. Bapak Prof. Dr. Ir. Teguh Soedarto MP. selaku Rektor Universitas Pembangunan Nasional “Veteran” Jawa Timur dan seluruh Pimpinan dan Pejabat di lingkungan Universitas Pembangunan Nasional “Veteran” Jawa Timur, terutama di lingkungan Fakultas Teknik dan Fakultas Ilmu Komputer, dan Program Studi Teknik Informatika dan Sistem Informasi. Atas dukungan moril maupun materiil kepada kami selama studi.
12. Seluruh Dosen dan Karyawan di lingkungan Universitas Pembangunan Nasional “Veteran” Jawa Timur, terutama di Fakultas Teknik dan Fakultas Ilmu Komputer, dan Program Studi Teknik Informatika dan Sistem Informasi. Terutama Bpk. Prof. Ahmad Fauzi, Bpk. Prof. Djohan Mashudi, Bpk. Sutiono, Ibu Ni Ketut Sari, Bpk. Purnomo Edi Sasongko, Ibu Asti Dwi Irfianti, Bpk. Tuhu Agung Rachmanto, Bpk. Minto Waluyo, Bpk. I Gede Susrama Mas Diyasa, Ibu Sintha Soraya Santi, Bpk. Budi Nugroho, Bpk. Doddy Ridwandono, Bpk. Mohammad Irwan Afandi, Bpk. Nur Cahyo Wibowo, Bpk. Sumaidi Wijaya, Bpk. Muhammad Pranoto, dan Bpk/Ibu dosen yang lain. Dan seluruh teman-teman di Laboratorium Robotika dan Sistem Cerdas Program Studi Teknik Informatika. Atas dukungan dan kerjasamanya selama kami menempuh studi.
13. Seluruh teman-teman di PUSKOM, Pusat Bahasa, Perpustakaan dan Biro Admik Universitas Pembangunan Nasional “Veteran” Jawa Timur,

terutama Bpk. Sunarto, S.Kom yang banyak membantu kelancaran studi kami.

14. Seluruh teman-teman di Pusat Studi Pengembangan ICT dan Komunitas Robotika Universitas Pembangunan Nasional “Veteran” Jawa Timur, atas dukungan dan kerjasamanya selama kami menempuh studi.
15. Teman-teman Institut Bisnis dan Informatika STIKOM Surabaya, atas dukungan moril maupun materiil selama kami menempuh studi, terutama buat Bpk Antok Supriyanto, Bpk. Arifin, Bpk. Anjik Sukmaaji, Bpk. Susijanto Tri Resmana, Bpk. Helmi Widyantara, Bpk. Harianto, dan lain-lain.
16. Teman-teman alumni Fisika MIPA ITS Surabaya dan alumni Program Instrumentasi dan Kontrol ITB, atas dukungan moril maupun materiil selama kami menempuh studi.
17. Teman-teman alumni SMA Negeri Pandaan, yang masih terus berkomunikasi dan memberikan support agar anak desa ini sukses dalam menyelesaikan studi.
18. Seluruh teman-teman alumni JWGE Biwako Jepang, alumni Workshop DIKTI, yang masih terus berkomunikasi dan memberikan support agar sama-sama sukses dalam menyelesaikan studi.
19. Seluruh Panitia Studium Generale Penulisan dan Publikasi Jurnal Ilmiah Auditorium Pascasarjana ITS, 1 September 2016, Pelatihan Penulisan Karya Ilmiah dan Publikasi Internasional UPN Veteran Jawa Timur (15 September 2016), Workshop dan Klinik Penulisan Manuskip untuk Peningkatan Kualifikasi Dosen Tahun 2016 – Wilayah Surabaya (19 - 21 September 2016), Workshop How To Be A Successful Doctoral Student Program S3 Teknik Elektro-ITS (21 September 2016), The 3rd Research Workshop on Operations Research and Optimization Modeling, UNPAD Training Center Bandung (October 6-7, 2016), Workshop on Intelligent Transportation Systems in Smart Cities, Multimedia Room, 1st Floor, TP. Rachmat Bld, Jalan Ganesha 10, Bandung (23 November 2016), atas pemberian bekal-bekal penting sebagai penunjang studi Program Doktor.

20. Segenap Dewan Pembina, Pengawas, Pimpinan dan Pengurus Yayasan Baabussalaam Al Islami Surabaya, Guru-guru dan Ustadz-ustadz kami, atas dukungan moril maupun materiil selama kami menempuh studi.
21. Seluruh teman-teman di CV. Puslit Indonesia (puslit.com), Riset Teknologi Informasi dan Komunikasi Indonesia (RISTEKOM) (ristekom.com), PT. Solusi Teknologi Informasi (solusiti.com), PT. Arta Kuat Saka Mandiri (Arkusama Indonesia) (alurkria.net), PT. Citra Roda Wahana Nusantara (CROWN) (crownparking.co.id), Gajah Media (gajahmedia.com), CV. Duta Sukses (duta.web.id), PT. Bangun Tirta Nusantara (batin.co.id), PT. IEF Consultant, CV. Guna Harsa Consultant, atas dukungan moril maupun materiil dan kerjasamanya selama kami menempuh studi.
22. Seluruh pimpinan dan pegawai terutama yang berkaitan dengan Pengembangan Teknologi Informasi untuk Badan Pengelolaan Infrastruktur Pemerintah Provinsi Papua, Biro Pemberdayaan Perempuan Sekretariat Daerah Provinsi Papua, Badan Penanggulangan Bencana Daerah Provinsi Papua, Komite Olahraga Nasional Indonesia (KONI) Provinsi papua, dan Pemerintah Kabupaten/Kota: Jayapura, Tarakan, Teluk Bintuni, Pamekasan, Keerom, Yalimo, Merauke, dan Pegunungan Bintang, atas dukungan moril maupun materiil dan kerjasamanya selama kami menempuh studi.
23. Guru-guru SMP kami yang masih terus menjalin komunikasi, terutama Ibu Ulina Primanti dan Ibu Pipit, atas doa dan nasehat-nasehatnya selama kami menempuh studi.
24. Keluarga Bpk. Ir. H. Muchayat (alm), keluarga Bpk. Kusyadi (alm), keluarga Bpk. Aliman, keluarga Ibu Projo, orang tua asuh kami, atas dukungan moril maupun materiil selama kami menempuh studi.
25. Bapak S. Moejiharto (alm), Ibu Katiningsih (alm), orang tua kami, Bapak H. Abd. Wahab Rosyid (alm) dan Hj. Ibu Sri Hery Wahyuni, mertua kami, dan semua saudara-saudara kami adik Sri Rahayu dan Dik Warsono sekeluarga, mas Bambang Prayitno dan mbak Yayuk Sri Mulyani sekeluarga, mas H. Totok Sudjarwanto dan mbak Hj. Atik sekeluarga, mas

H. Danu Darmaji dan mbak Hj. Umi Kulsum sekeluarga, mas Muadji Sugiarto (alm) dan mbak Sumiati sekeluarga, mbak Ruminingsih dan mas Subriyanto (alm) sekeluarga, mas Margi Hartono (alm), mbak Anik dan mas Miono sekeluarga, mas Gatot Budi Santoso (alm) dan mbak Lilit Zulaikhah sekeluarga, mbak Luluk dan Mas Chairuddin Muchtar sekeluarga, dan mbak Sugiharti (alm) sekeluarga. Poespita Koesoema Wardhani (istri) dan anak-anak tercinta Aisyah Firdausi Rahma, Hafshah Sakinah dan Hanin Azka Zahirah, yang senantiasa mendukung dan tak kering akan doa hingga selesainya studi ini.

26. Seluruh rekan-rekan mahasiswa S3 terutama S3 Teknik Elektro seperjuangan : Aris Tjahyanto, Lie, Nyoman, Gede, Endang, Susyanto, Cahyo, Firman, Arifin, Fanani, Solekhan, Delta, Diah, Donny, Helmy, Harianto, Anjik, Max Muljono, Atik, Radi, Iwan, Aris Nasuha, Lukman, Tamaji, Tanwir, Suaib Halim, Pujiono, Suyanto, Fanani Polije, Anang, Alamsyah, Asti, dan lainnya.
27. Dan seluruh teman-teman, kenalan dan saudara-saudara yang tidak dapat kami sebutkan satu persatu, atas doa dan dukungannya selama ini.

Penulis mengucapkan banyak terima kasih telah membantu selama melakukan kegiatan riset dan studi S3 ini. Semoga Allah SWT memberi balasan dengan pahala kebaikan yang sempurna. Demikian Laporan Disertasi ini disusun, segala masukan dan koreksi sangat diharapkan untuk penyempurnaan ke depan. Semoga bermanfaat.

Surabaya, Januari 2018

Penulis

DAFTAR ISI

LEMBAR PENGESAHAN	Error! Bookmark not defined.
SURAT PERNYATAAN KEASLIAN DISERTASI.....	v
ABSTRAK	vii
KATA PENGANTAR.....	xi
DAFTAR GAMBAR.....	xxi
DAFTAR TABEL.....	xxvii
DAFTAR NOTASI.....	xxix
BAB 1.....	1
PENDAHULUAN.....	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	4
1.3. Tujuan Penelitian	6
1.4. Manfaat Penelitian	7
1.5. Kontribusi Penelitian.....	7
1.6. Gambaran Umum dan Posisi Penelitian.....	8
1.7. Sistematika Penulisan Disertasi	12
BAB 2.....	13
EKSTRAKSI PELAT NOMOR KENDARAAN	13
2.1. Teori Umum.....	13
2.2. Pemrosesan Awal	13
2.2.1. <i>Grayscale</i>	13
2.2.2. <i>Black and White</i>	14
2.2.3. <i>Median Filtering</i>	14
2.2.4. Dilasi	14
2.2.5. Erosi	14
2.2.6. Konvolusi	15
2.2.7. <i>Flood fill</i>	15
2.3. Fitur-fitur Ekstraksi Pelat Nomor Kendaraan	15
2.4. <i>Smearing Algorithm</i>	16
2.5. Ekstraksi Pelat Nomor Kendaraan dengan <i>Smearing Algorithm</i>	20

2.6.	Hasil Ekstraksi Pelat Nomor Kendaraan dengan <i>Smearing Algorithm</i>	20
2.7.	Ringkasan Hasil	22
BAB 3		25
PELACAKAN PELAT NOMOR KENDARAAN.....		25
3.1.	Fitur-fitur Pelacakan Pelat Nomor Kendaraan	25
3.2.	Estimasi Densitas.....	26
3.2.1.	<i>Probability Density Function (PDF)</i>	26
3.2.2.	Estimasi Densitas Kernel.....	27
3.2.3.	Fungsi Kernel	30
3.2.4.	Pemilihan Lebar Pita atau Radius Kernel.....	30
3.3.	Mean Shift Dasar	32
3.4.	Pelacakan Pelat Nomor Kendaraan dengan Mean Shift	34
3.5.	Pelacakan Pelat Nomor Kendaraan dengan BIM-Mean Shift	37
3.5.1.	BIM-Mean Shift dengan <i>Switching Kernel</i>	38
3.5.2.	BIM-Mean Shift dengan <i>Adaptive Fuzzy Gaussian Kernel</i>	39
3.6.	Pengujian Pelacakan Pelat Nomor Kendaraan dengan BIM-Mean Shift	44
3.6.1.	Pengujian BIM-Mean Shift dengan <i>Switching Kernel</i>	44
3.6.2.	Pengujian BIM-Mean Shift dengan <i>Adaptive Fuzzy Gaussian Kernel</i>	54
3.7.	Ringkasan Hasil	62
BAB 4		63
EKSTRAKSI KARAKTER PELAT NOMOR KENDARAAN.....		63
4.1.	Fitur-fitur Ekstraksi Karakter Pelat Nomor Kendaraan.....	63
4.1.1.	Fitur-fitur Segmentasi Karakter Pelat Nomor Kendaraan	63
4.1.2.	Fitur-fitur Pengenalan Karakter Pelat Nomor Kendaraan	64
4.2.	Segmentasi Karakter Pelat Nomor Kendaraan	66
4.2.1.	<i>Cellular Neural Network (CNN)</i>	66
4.2.2.	Arsitektur CNN.....	71
4.2.3.	<i>Hybrid CNN</i>	72
4.2.4.	Segmentasi Karakter Pelat Nomor Kendaraan dengan BIM-HCNN	73
4.2.4.1.	Optimasi <i>Template CNN</i> dengan <i>Adaptive Fuzzy</i>	74
4.2.4.2.	Optimasi <i>Template CNN</i> dengan <i>Neuro-Fuzzy</i>	78

4.2.5. Pengujian Segmentasi Karakter Pelat Nomor Kendaraan dengan BIM-HCNN	81
4.2.6. Ringkasan Hasil	85
4.3. Pengenalan Karakter Pelat Nomor Kendaraan.....	86
4.3.1. <i>Extreme Learning Machine (ELM)</i>	86
4.3.2. Pengenalan Karakter Pelat Nomor Kendaraan dengan ELM.....	89
4.3.3. Pengujian Pengenalan Karakter Pelat Nomor Kendaraan dengan ELM	89
4.3.4. Ringkasan Hasil	92
BAB 5.....	93
KESIMPULAN & RENCANA PENELITIAN SELANJUTNYA.....	93
5.1. Kesimpulan	93
5.2. Rencana Penelitian Selanjutnya	94
DAFTAR PUSTAKA.....	95

Halaman ini sengaja dikosongkan

DAFTAR GAMBAR

Gambar :	Halaman
Gambar 1.1. Contoh Gambar Pelat Nomor kendaraan.....	2
Gambar 1.2. Gambaran umum penelitian pelacakan dan pengenalan pelat nomor kendaraan berbasis video....	10
Gambar 1.3. <i>Fishbone Diagram</i> posisi penelitian pelacakan dan pengenalan pelat nomor kendaraan berbasis video diantara penelitian yang ada saat ini didunia.....	11
Gambar 2.1. Contoh keluaran <i>Smearing Algorithm</i>	18
Gambar 2.2. Diagram alir <i>Smearing Algorithm</i>	19
Gambar 2.3. Ekstraksi pelat nomor kendaraan dengan metode <i>Smearing Algorithm</i>	20
Gambar 2.4. Contoh gambar kendaraan.....	21
Gambar 2.5. Hasil proses <i>grayscale</i>	21
Gambar 2.6. Hasil proses binerisasi.....	21
Gambar 2.7. Hasil proses erosi, dilasi dan konvolusi.....	22
Gambar 2.8. Hasil proses pengolesan (<i>smearing</i>) dan penentuan lokasi.....	22
Gambar 3.1. PDF Gaussian dengan $c = 0, \sigma = 1$	26
Gambar 3.2. <i>Hypercube</i> persegi 2-D.....	28
Gambar 3.3. <i>Hypercube</i> kubus 3-D.....	29
Gambar 3.4. Pengaruh lebar pita atau radius kernel.....	31
Gambar 3.5. Bentuk topeng <i>Parzen window kernel</i> dengan menggunakan fungsi kernel Gaussian masing-masing dengan lebar pita atau radius kernel, (a) $h = 65\%$, (b) $h = 75\%$, (c) $h = 85\%$ dan (d) $h = 95\%$	31
Gambar 3.6. Bentuk topeng <i>Parzen window kernel</i> dengan menggunakan fungsi kernel: (a) Seragam (b) Segitiga (c) Epanechnikov (d) Gaussian, masing-	

masing dengan lebar pita atau radius kernel $h =$ 85%.....	32
Gambar 3.7. Distribusi bola bilyar identik.....	33
Gambar 3.8. BIM-Mean Shift dengan <i>Switching Kernel</i>	40
Gambar 3.9. BIM-Mean Shift dengan <i>Adaptive Fuzzy Gaussian Kernel</i>	42
Gambar 3.10. Fungsi keanggotaan fuzzy masukan <i>error</i> dan <i>delta error</i>	43
Gambar 3.11. Fungsi keanggotaan fuzzy keluaran.....	43
Gambar 3.12. Bentuk keluaran fungsi kesamaan, rata-rata keluaran fungsi kesamaan dan plot <i>Switching Kernel</i> dari hasil data Video 1.....	45
Gambar 3.13. Hasil proses pelacakan Video 1 (dari atas ke bawah) menggunakan Fungsi Kernel: Seragam, Segitiga, Epanechnikov, Gaussian dan <i>Switching Kernel</i> . Hasil pelacakan pada frame ke-16 dan ke-32.....	46
Gambar 3.14. Bentuk keluaran fungsi kesamaan, rata-rata keluaran fungsi kesamaan dan plot <i>Switching Kernel</i> dari hasil data Video 2.....	46
Gambar 3.15. Hasil proses pelacakan Video 2 (dari atas ke bawah) menggunakan Fungsi Kernel: Seragam, Segitiga, Epanechnikov, Gaussian dan <i>Switching Kernel</i> . Hasil pelacakan pada frame ke-16 dan ke-32.....	47
Gambar 3.16. Bentuk keluaran fungsi kesamaan, rata-rata keluaran fungsi kesamaan dan plot <i>Switching Kernel</i> dari hasil data Video 3.....	48
Gambar 3.17. Hasil proses pelacakan Video 3 (dari atas ke bawah) menggunakan Fungsi Kernel: Seragam, Segitiga, Epanechnikov, Gaussian dan <i>Switching Kernel</i> .	

Hasil pelacakan pada frame ke-20 dan ke-40.....	48
Gambar 3.18. Bentuk keluaran fungsi kesamaan, rata-rata keluaran fungsi kesamaan dan plot <i>Switching Kernel</i> dari hasil data Video 4.....	49
Gambar 3.19. Hasil proses pelacakan Video 4 (dari atas ke bawah) menggunakan Fungsi Kernel: Seragam, Segitiga, Epanechnikov, Gaussian dan <i>Switching Kernel</i> . Hasil pelacakan pada frame ke-15 dan ke-30.....	50
Gambar 3.20. Bentuk keluaran fungsi kesamaan, rata-rata keluaran fungsi kesamaan dan plot <i>Switching Kernel</i> dari hasil data Video 5.....	50
Gambar 3.21. Hasil proses pelacakan Video 5 (dari atas ke bawah) menggunakan Fungsi Kernel: Seragam, Segitiga, Epanechnikov, Gaussian dan <i>Switching Kernel</i> . Hasil pelacakan pada frame ke-15 dan ke-30.....	51
Gambar 3.22. Contoh hasil simulasi pelacakan.....	52
Gambar 3.23. Contoh objek yang berhasil dilacak pada frame ke-18.....	53
Gambar 3.24. Contoh objek yang gagal dilacak pada frame ke-30...	53
Gambar 3.25. Bentuk keluaran fungsi kesamaan, rata-rata keluaran fungsi kesamaan dan plot lebar pita atau radius kernel dari hasil data Video 1.....	55
Gambar 3.26. Fungsi Keanggotaan Fuzzy awal dan akhir hasil data Video 1.....	55
Gambar 3.27. Hasil proses pelacakan Video 1 sebelah kiri menggunakan <i>Static Gaussian Kernel</i> dan sebelah kanan menggunakan <i>Adaptive Fuzzy Gaussian Kernel</i> . Hasil pelacakan (dari atas ke bawah) pada frame ke-25, 50, 75 dan ke-100.....	56

Gambar 3.28.	Bentuk keluaran fungsi kesamaan, rata-rata keluaran fungsi kesamaan dan plot lebar pita atau radius kernel dari hasil data Video 2.....	57
Gambar 3.29.	Fungsi Keanggotaan Fuzzy awal dan akhir hasil data Video 2.....	57
Gambar 3.30.	Hasil proses pelacakan Video 2 sebelah kiri menggunakan <i>Static Gaussian Kernel</i> dan sebelah kanan menggunakan <i>Adaptive Fuzzy Gaussian Kernel</i> . Hasil pelacakan (dari atas ke bawah) pada frame ke-30, 60, 90 dan ke-120.....	58
Gambar 3.31.	Bentuk keluaran fungsi kesamaan, rata-rata keluaran fungsi kesamaan dan plot lebar pita atau radius kernel dari hasil data Video 3.....	58
Gambar 3.32.	Fungsi Keanggotaan Fuzzy awal dan akhir hasil data Video 3.....	59
Gambar 3.33.	Hasil proses pelacakan Video 3 sebelah kiri menggunakan <i>Static Gaussian Kernel</i> dan sebelah kanan menggunakan <i>Adaptive Fuzzy Gaussian Kernel</i> . Hasil pelacakan (dari atas ke bawah) pada frame ke-25, 50, 75 dan ke-100.....	59
Gambar 3.34.	Bentuk keluaran fungsi kesamaan, rata-rata keluaran fungsi kesamaan dan plot lebar pita atau radius kernel dari hasil data Video 4.....	60
Gambar 3.35.	Fungsi Keanggotaan Fuzzy awal dan akhir hasil data Video 4.....	60
Gambar 3.36.	Hasil proses pelacakan Video 4 sebelah kiri menggunakan <i>Static Gaussian Kernel</i> dan sebelah kanan menggunakan <i>Adaptive Fuzzy Gaussian Kernel</i> . Hasil pelacakan (dari atas ke bawah) pada frame ke-25, 50, 75 dan ke-100.....	61
Gambar 4.1.	Arsitektur CNN standar.....	66
Gambar 4.2.	(a) lingkungan $r=1(3 \times 3)$. (b) lingkungan $r=2(5 \times 5)$	68

Gambar 4.3.	Sel perbatasan.....	69
Gambar 4.4.	Nonlinearitas standar.....	70
Gambar 4.5.	Blok diagram CNN.....	71
Gambar 4.6.	Blok diagram HCNN.....	74
Gambar 4.7.	Blok diagram BIM-HCNN untuk segmentasi karakter pelat nomor kendaraan.....	75
Gambar 4.8.	(a) Fungsi Keanggotaan Error. (b) Fungsi Keanggotaan Delta Error.....	75
Gambar 4.9.	Pembangkitan variasi template CNN menggunakan <i>Adaptive Fuzzy</i>	77
Gambar 4.10.	Pembangkitan variasi template CNN menggunakan ANFIS.....	78
Gambar 4.11.	Struktur ANFIS.....	79
Gambar 4.12.	Fungsi Keanggotaan Fuzzy awal dan akhir BIM-HCNN dengan <i>Adaptive Fuzzy</i>	82
Gambar 4.13.	Fungsi Keanggotaan Fuzzy awal dan akhir BIM-HCNN dengan ANFIS.....	82
Gambar 4.14.	Hasil segmentasi karakter pelat nomor kendaraan Indonesia.....	84
Gambar 4.15.	Struktur ELM.....	87
Gambar 4.16.	Pengenalan karakter pelat nomor kendaraan dengan ELM.....	90

Halaman ini sengaja dikosongkan

DAFTAR TABEL

Tabel :		Halaman
Tabel 1.1.	Matriks posisi penelitian pada penelitian terkait.....	3
Tabel 2.1.	Perbandingan metode ekstraksi pelat nomor kendaraan.....	17
Tabel 2.2.	Hasil ekstraksi pelat nomor kendaraan.....	23
Tabel 3.1.	Aturan Sistem Fuzzy.....	44
Tabel 3.2.	Parameter Mean Shift.....	44
Tabel 3.3.	Akurasi pelacakan BIM-Mean Shift dengan <i>Switching Kernel (%)</i>	54
Tabel 3.4.	Akurasi pelacakan BIM-Mean Shift dengan <i>Adaptive Fuzzy Gaussian Kernel (%)</i>	61
Tabel 4.1.	Perbandingan metode segmentasi karakter pelat nomor kendaraan.....	65
Tabel 4.2.	Perbandingan metode pengenalan karakter pelat nomor kendaraan.....	67
Tabel 4.3.	Nilai template hasil optimasi.....	81
Tabel 4.4.	Hasil pengukuran kualitas gambar.....	84
Tabel 4.5.	Hasil proses pelatihan ELM.....	91
Tabel 4.6.	Hasil pengenalan karakter pelat nomor kendaraan dengan ELM.....	91

Halaman ini sengaja dikosongkan

DAFTAR NOTASI

u	: fitur warna.
s	: lokasi pusat objek (target) pada frame video sebelumnya.
s_i	: lokasi piksel dalam kernel pada frame video saat ini.
$g(.)$: fungsi kernel.
C	: faktor normalisasi model objek.
C_h	: faktor normalisasi model kandidat.
$b(s_i)$: indeks warna (1..m) dari s_i .
h	: lebar pita atau radius kernel.
q_u	: probabilitas fitur warna u dari model objek.
$p_u(s)$: probabilitas fitur warna u dari model kandidat.
$f_1(s)$: fungsi kesamaan.
$f_2(s)$: koefisien Bhattacharyya.
$m(s)$: vektor Mean Shift yang diinginkan dalam satu iterasi dalam kernel.
km	: topeng (<i>mask</i>) Parzen <i>window</i> .
gx, gy	: gradien Parzen <i>window</i> terhadap sumbu x dan y.
tm, lm	: tinggi dan lebar ukuran topeng Parzen <i>window</i> .
$graph$: plot gambar topeng (<i>mask</i>) jika $graph=1$.
$loss$: objek (target) keluar dari pelacakan.
(f, f_idx)	: penyimpanan hasil fungsi kesamaan selama proses pelacakan.
$I2$: frame berikutnya.
$height, width$: ukuran $I2$.
$Lmap$: Panjang <i>colormap</i> .
f_thresh	: nilai ambang dari fungsi kesamaan.
max_it	: iterasi maksimal.
ε	: rata-rata keluaran fungsi kesamaan hasil proses pelacakan menggunakan Mean Shift standar dengan fungsi kernel Gaussian.
Z	: jumlah frame.
n_e	: nilai <i>error</i> atau nilai <i>delta error</i> .
μ	: fungsi keanggotaan fuzzy.

- σ_i, a_i, b_i, c_i : parameter fungsi keanggotaan fuzzy.
- O_i : Objek yang berhasil dilacak pada *frame* ke-i.
- $C(i, j)$: Sel CNN.
- r : radius dari sel CNN.
- $S_r(i, j)$: Lingkup pengaruh sel CNN sejauh radius r .
- $x_{ij}, y_{kl}, u_{kl}, z_{ij}$: keadaan (*state*), keluaran, masukan, nilai ambang dari sel CNN.
- \mathfrak{R} : ruang keadaan.
- $A(i, j ; k, l)$: *template* umpanbalik.
- $B(i, j ; k, l)$: *template* kontrol.
- ρ_{ij} : baris ke-i dan kolom ke-j piksel dari gambar target,
- $\overline{\rho_{ij}}$: baris ke-i dan kolom ke-j piksel dari gambar hasil segmentasi,
- ψ : rentang dinamis nilai piksel, atau nilai maksimum piksel (setara dengan (255) untuk gambar 8-bit).
- PDF : *Probability Density Function*.
- KDE : *Kernel Density Estimation*.
- ROI : *Region of Interest*.

BAB 1

PENDAHULUAN

1.1. Latar Belakang

Sistem pelacakan (*tracking system*) dan pengenalan pelat nomor kendaraan (*license plate recognition system*) selalu menjadi bahan penelitian menarik dan menantang untuk dilakukan. Sampai beberapa tahun terakhir ini masih banyak paper-paper publikasi baik dari hasil-hasil seminar atau pun konferensi maupun jurnal internasional yang membahas tentang penelitian ini. Diantaranya dapat disebutkan antara lain sebagai berikut: (Luvizon, Nassu and Minetto, 2017), (Fernández-Ares *et al.*, 2017), (Hyun, Tok and Ritchie, 2017), (Khakpour, Pazzi and El-Khatib, 2017), (Saini and Saini, 2017), dan (Anandhalli and Baligar, 2017).

Banyak penerapan maupun penelitian terkait yang memanfaatkan hasil sistem pendektsian dan pengenalan pelat nomor kendaraan ini, diantaranya adalah sistem pembayaran tol otomatis, sistem transportasi cerdas, sistem parkir cerdas, sistem pengaturan lalu-lintas cerdas, sistem keamanan kendaraan, sistem pencegahan kecelakaan kendaraan, sistem pemanduan kendaraan otomatis, sistem pengawasan pelanggaran lalu-lintas, sistem pelacakan dan pengenalan pelat nomor kendaraan yang dicurigai (*suspicious vehicle*) berdasarkan pembacaan kamera yang terpasang di tiap-tiap perempatan jalan raya, dan lain-lain.

Sistem pendektsian dan pengenalan pelat nomor kendaraan didalam banyak penelitian dikenal dengan istilah *Automatic License Plate Recognition* (ALPR) (pengenalan pelat nomor kendaraan otomatis) (Khoshki and Ganesan, 2015), *Automatic Vehicle Identification* (AVI) (identifikasi kendaraan otomatis) (Chang, Su and Qian-Yu, 2012), *Car Plate Recognition* (CPR) (pengenalan plat mobil) atau *Car Number Plate Recognition* (CNPR) (pengenalan pelat nomor mobil) (Khan and Shah, 2016), *Automatic Number Plate Recognition* (ANPR) (pengenalan pelat nomor otomatis) (Zhu *et al.*, 2016), dan *Optical Character Recognition* (OCR) (pengenalan karakter optik) untuk mobil (Jain and Sharma, 2014).

Pendekatan umum yang digunakan untuk pendektsian dan pengenalan pelat nomor kendaraan biasanya menggunakan teknik berbasis visi (*vision-based techniques*) dari gambar atau pun video. Bagaimana pun, sehubungan dengan banyaknya permasalahan pelat nomor kendaraan meliputi warna, ukuran, orientasi,

bentuk dan pose pelat nomor kendaraan, menjadikan pengembangan sistem pendekripsi dan pengenalan pelat nomor kendaraan yang otomatis, tegar (*robust*) dan efektif menjadi sebuah tantangan tersendiri. Belum lagi beragamnya pelat nomor kendaraan berbagai negara menjadikan penggunaan metode pendekripsi dan pengenalan pelat nomor kendaraan tidak bisa digunakan secara universal. Umumnya masing-masing negara memiliki pelat nomor kendaraan dan metode pendekripsi dan pengenalan pelat nomor sendiri secara spesifik dan khas. Contoh gambar pelat nomor kendaraan seperti diperlihatkan pada Gambar 1.1.



(a)



(b)



(c)



(d)

Gambar 1.1. Contoh Gambar Pelat Nomor kendaraan

Dari contoh gambar pelat nomor kendaraan pada Gambar 1.1, (a) contoh pelat nomor kendaraan Indonesia standar, (b) contoh pelat nomor kendaraan Indonesia tidak standar, (c) contoh pelat nomor kendaraan Alberta, Kanada standar, dan (d) contoh pelat nomor kendaraan Alberta, Kanada tidak standar (Du et al., 2013).

Hal-hal di atas menjadikan penelitian pendekripsi dan pengenalan pelat nomor kendaraan seperti tidak pernah selesai dan selalu ada celah untuk terus dikembangkan untuk kemajuan di masa depan. Di sisi lain pengembangan komputasi cerdas atau *soft computing* memberikan peluang dimunculkan atau ditemukannya metode-metode baru dalam pengembangan algoritma maupun pemodelan matematis untuk menyelesaikan permasalahan pelacakan dan pengenalan pelat nomor kendaraan tersebut.

Beberapa penelitian telah dilakukan terkait dengan pelacakan dan pengenalan pelat nomor kendaraan ini. Data penelitian, topik penelitian dan fitur serta metode yang digunakan ditunjukkan pada Tabel 1.1.

Tabel 1.1. Matriks posisi penelitian pada penelitian terkait

Peneliti/Tahun	Topik Penelitian	Fitur dan Metode yang Digunakan
(Du <i>et al.</i> , 2013) (Anakavej, Kawewong and Patanukhom, 2014), (Gou <i>et al.</i> , 2014).	Ekstraksi pelat nomor kendaraan	Informasi batas / tepi, informasi gambar global, fitur tekstur, fitur warna, fitur karakter, dll. <i>Smearing Algorithm</i>
(Sahoo and Ari, 2015), (Owczarek, Baranski and Strumillo, 2015) (Chan, Zhou and Chen, 2015) (Yang <i>et al.</i> , 2015) (Owczarek, Baranski and Strumillo, 2015) (Li <i>et al.</i> , 2015), (Setyawan, Purwanto and Mardiyanto, 2015), (Yang <i>et al.</i> , 2015)	Pelacakan (<i>tracking</i>) pelat nomor kendaraan	Fitur warna, fitur superpiksel, <i>Oriented FAST and Rotated BRIEF</i> (ORB) <i>Feature Matching</i> , fitur tepi, <i>optical flow feature</i> , fitur tekstur, dll. BIM-Mean Shift
(Du <i>et al.</i> , 2013) (Duraisamy and Jane, 2014), (Zeng <i>et al.</i> , 2014)	Ekstraksi karakter pelat nomor kendaraan (segmentasi karakter)	Konektivitas piksel, profil proyeksi, Pengetahuan karakter sebelumnya, kontur karakter, fitur kombinasi, dll. BIM-HCNN
(Du <i>et al.</i> , 2013) (Saabni, 2015), (Gou <i>et al.</i> , 2014) (Rabee and Barhumi, 2014), (Ghahnavieh, Amirkhani-Shahraki and Raie, 2014), (Yuan <i>et al.</i> , 2017)	Ekstraksi karakter pelat nomor kendaraan (pengenalan karakter)	Data mentah, ekstraksi fitur, dll. <i>Extreme Learning Machine</i> (ELM)

1.2. Rumusan Masalah

Usaha penyelesaian permasalahan yang ingin dilakukan pada penelitian ini secara umum adalah bagaimana melakukan pelacakan (*tracking*) dan pengenalan (*recognition*) pelat nomor kendaraan Indonesia, berbasis video rekaman kendaraan yang berjalan di jalan raya. Terdapat tiga bagian besar yang dikembangkan dalam melakukan proses pelacakan dan pengenalan pelat nomor kendaraan ini, yaitu:

- Bagian ekstraksi pelat nomor kendaraan (*license plate extraction*) untuk mendapatkan lokasi pelat nomor kendaraan (*license plate localization*) yang terdapat dalam *frame* video.
- Bagian pelacakan (*tracking*) lokasi pelat nomor kendaraan sepanjang beberapa *frame* video, dan
- Bagian ekstraksi karakter yang terdapat pada pelat nomor kendaraan (*vehicle license plate character extraction*), yang didalamnya terdapat permasalahan segmentasi karakter pelat nomor kendaraan (*vehicle license plate character segmentation*) dan pengenalan karakter pelat nomor kendaraan (*vehicle license plate character recognition*).

Didalam ketiga bagian besar di atas, terdapat permasalahan lain yang harus disiapkan atau diatasi agar ketiganya dapat diwujudkan, permasalahan penting diantaranya:

- Pemrosesan awal (*preprocessing*) diantaranya proses *grayscale*, proses binerisasi (*binarization*), proses *median filtering*, proses *dilation*, proses *convolution*, proses *erosion*, dan proses *flood fill*.
- Metode-metode atau algoritma-algoritma yang digunakan untuk ekstraksi pelat nomor kendaraan, pelacakan pelat nomor kendaraan, dan ekstraksi karakter pelat nomor kendaraan yang terdiri dari segmentasi karakter pelat nomor kendaraan dan pengenalan karakter pelat nomor kendaraan.

Permasalahan lain berkaitan dengan data berupa hasil rekaman video dalam proses ekstraksi pelat nomor kendaraan, pelacakan pelat nomor kendaraan sepanjang beberapa *frame* video, dan ekstraksi karakter pelat nomor kendaraan, bisa saja akan muncul objek lain selain latar belakang mungkin mirip dengan penampilan target sehingga dapat mengganggu pengamatan. Dalam kasus seperti itu, mungkin sulit

untuk membedakan fitur dari target yang diharapkan. Fenomena ini dikenal dengan *clutter* (kekacauan). Selain tantangan akibat *clutter*, juga perlu diperhatikan faktor-faktor berikut (Maggio and Cavallaro, 2011):

- a. Perubahan pose (*changes in pose*). Sebuah target yang bergerak bervariasi penampilannya ketika diproyeksikan ke bidang gambar, misalnya ketika berputar atau berubah arah.
- b. Lingkungan pencahayaan (*ambient illumination*). Arah, intensitas dan warna lingkungan cahaya berpengaruh terhadap penampilan target. Selain itu, perubahan iluminasi global sering menjadi tantangan untuk data hasil rekaman video di luar ruangan. Misalnya, lingkungan cahaya berubah ketika awan mengaburkan matahari. Juga, sudut antara arah cahaya dan normal ke permukaan objek berbeda dengan pose objek, sehingga mempengaruhi bagaimana kita melihat objek melalui lensa kamera.
- c. Derau (*noise*). Proses akuisisi serangkaian gambar dalam *frame* video bisa terdapat derau, tergantung pada kualitas kamera yang digunakan. Observasi objek akibat derau bisa saja menjadikan sebagian datanya rusak sehingga berpengaruh terhadap kinerja sistem.
- d. Oklusi (*occlusions*). Target mungkin gagal untuk diamati ketika sebagian atau seluruhnya tersumbat (tertutupi) oleh benda-benda lain dalam adegan (*scene*).

Selain itu, permasalahan penting lain berkaitan dengan pelat nomornya sendiri. Perlu diperhatikan beberapa permasalahan dari jenis-jenis pelat nomor kendaraan atau lingkungan yang menjadi tantangan dalam pendekripsi dan pengenalan pelat nomor kendaraan. Permasalahan-permasalahan tersebut antara lain sebagai berikut (Du *et al.*, 2013):

1). Variasi pelat nomor kendaraan

- a) lokasi (posisi) : pelat nomor kendaraan berada di lokasi (di posisi) yang berbeda untuk tiap kendaraan;
- b) kuantitas: sebuah gambar kendaraan bisa saja tidak terdapat atau banyak (lebih dari satu) plat;

- c) ukuran: pelat nomor kendaraan bisa terdapat berbagai variasi ukuran tergantung dari jarak pengambilan gambar oleh kamera dan faktor zoom (*zoom factor*);
- d) warna: pelat nomor kendaraan bisa terdapat bermacam karakter (huruf dan angka) dan warna latar belakang sesuai dengan jenis plat atau peralatan pengambilan gambar.
- e) huruf: pelat nomor di satu negara dan negara lain bisa berbeda penggunaan jenis huruf dan bahasa yang berbeda;
- f) standar dan tidak standar: tiap negara memiliki aturan standar pelat nomor kendaraan, namun demikian banyak dijumpai kendaraan menggunakan pelat nomor kendaraan yang tidak standar;
- g) oklusi: pelat nomor kendaraan dapat dikaburkan oleh kotoran;
- h) Kecenderungan: pelat nomor kendaraan dapat dimiringkan;
- i) lain-lain: penambahan karakter, pelat nomor kendaraan diberi bingkai dan skrup.

2). Variasi Lingkungan

- a) iluminasi: gambar masukan mungkin memiliki berbeda jenis pencahayaan, terutama karena lingkungan pencahayaan dan lampu kendaraan;
- b) latar belakang: latar belakang gambar mungkin memiliki pola yang mirip dengan pelat nomor kendaraan, seperti nomor yang tertera pada kendaraan, bumper dengan pola vertikal, dan lantai bertekstur.

1.3. Tujuan Penelitian

Tujuan utama dari penelitian ini adalah mengembangkan metode komputasi cerdas untuk pelacakan dan pengenalan pelat nomor kendaraan Indonesia berbasis video rekaman kendaraan yang berjalan di jalan raya. Sedangkan untuk tujuan dalam setiap proses yang dilakukan adalah:

- Melakukan pendekripsi pelat nomor kendaraan atau ekstraksi pelat nomor kendaraan (*license plate extraction*) untuk mendapatkan lokasi pelat nomor kendaraan (*license plate localization*) yang terdapat dalam *frame* video. Untuk mendapatkan hasil ekstraksi pelat nomor ini digunakan metode *Smearing Algorithm*.

- Melakukan pelacakan (*tracking*) lokasi pelat nomor kendaraan sepanjang beberapa *frame* video. Untuk keperluan proses pelacakan pelat nomor sepanjang *frame* video ini digunakan metode Basuki-I Ketut-Mauridhi (BIM) Mean Shift (BIM-Mean Shift) dengan *Switching Kernel* dan *Adaptive Fuzzy Gaussian Kernel*.
- Melakukan proses ekstraksi karakter yang terdapat pada pelat nomor kendaraan (*vehicle license plate extraction*). Didalamnya sudah termasuk proses segmentasi karakter pelat nomor kendaraan, dan pengenalan karakter pelat nomor kendaraan. Untuk keperluan segmentasi karakter pelat nomor kendaraan digunakan metode Basuki-I Ketut-Mauridhi (BIM) *Hybrid Cellular Neural Network* (BIM-HCNN). Sedangkan untuk keperluan pengenalan karakter pelat nomor kendaraan digunakan metode *Extreme Learning Machine* (ELM).

1.4. Manfaat Penelitian

Penelitian yang dilakukan mempunyai manfaat untuk masyarakat, baik masyarakat akademik maupun non akademik. Adapun beberapa manfaat dari penelitian ini, diharapkan hasil penelitian pelacakan dan pengenalan pelat nomor kendaraan ini dapat digunakan untuk aplikasi pada bidang-bidang lain yang terkait, antara lain untuk: sistem pembayaran tol otomatis, sistem transportasi cerdas, sistem parkir cerdas, sistem pengaturan lalu-lintas cerdas, sistem keamanan kendaraan, sistem pencegahan kecelakaan kendaraan, sistem pemanduan kendaraan otomatis, sistem pengawasan pelanggaran lalu-lintas, sistem pelacakan dan pengenalan pelat nomor kendaraan yang dicurigai (*suspicious vehicle*) berdasarkan pembacaan kamera yang terpasang di tiap-tiap perempatan jalan raya, *Smart City*, dan lain-lain.

1.5. Kontribusi Penelitian

Penelitian ini berkontribusi terhadap perkembangan teknologi pelacakan dan pengenalan pelat nomor kendaraan Indonesia.

Kebaruan dari penelitian ini:

1. Pengembangan metode pelacakan pelat nomor kendaraan Indonesia menggunakan metode BIM-Mean Shift, yaitu algoritma Mean Shift yang ditingkatkan kinerjanya menggunakan teknik *Switching Kernel* dan *Adaptive Fuzzy Gaussian Kernel*.

2. Pengembangan algoritma segmentasi karakter pelat nomor kendaraan Indonesia menggunakan metode BIM-HCNN, yaitu CNN yang dioptimasi dengan algoritma *Adaptive Fuzzy* dan *Neuro-Fuzzy*.

1.6. Gambaran Umum dan Posisi Penelitian

Tiga bagian besar yang dikembangkan dalam proses pelacakan dan pengenalan pelat nomor kendaraan ini, yaitu: ekstraksi pelat nomor kendaraan, pelacakan pelat nomor kendaraan sepanjang *frame* video, dan ekstraksi karakter pelat nomor kendaraan, yang terdiri dari segmentasi karakter pelat nomor kendaraan dan pengenalan karakter pelat nomor kendaraan. Masing-masing dijelaskan sebagai berikut.

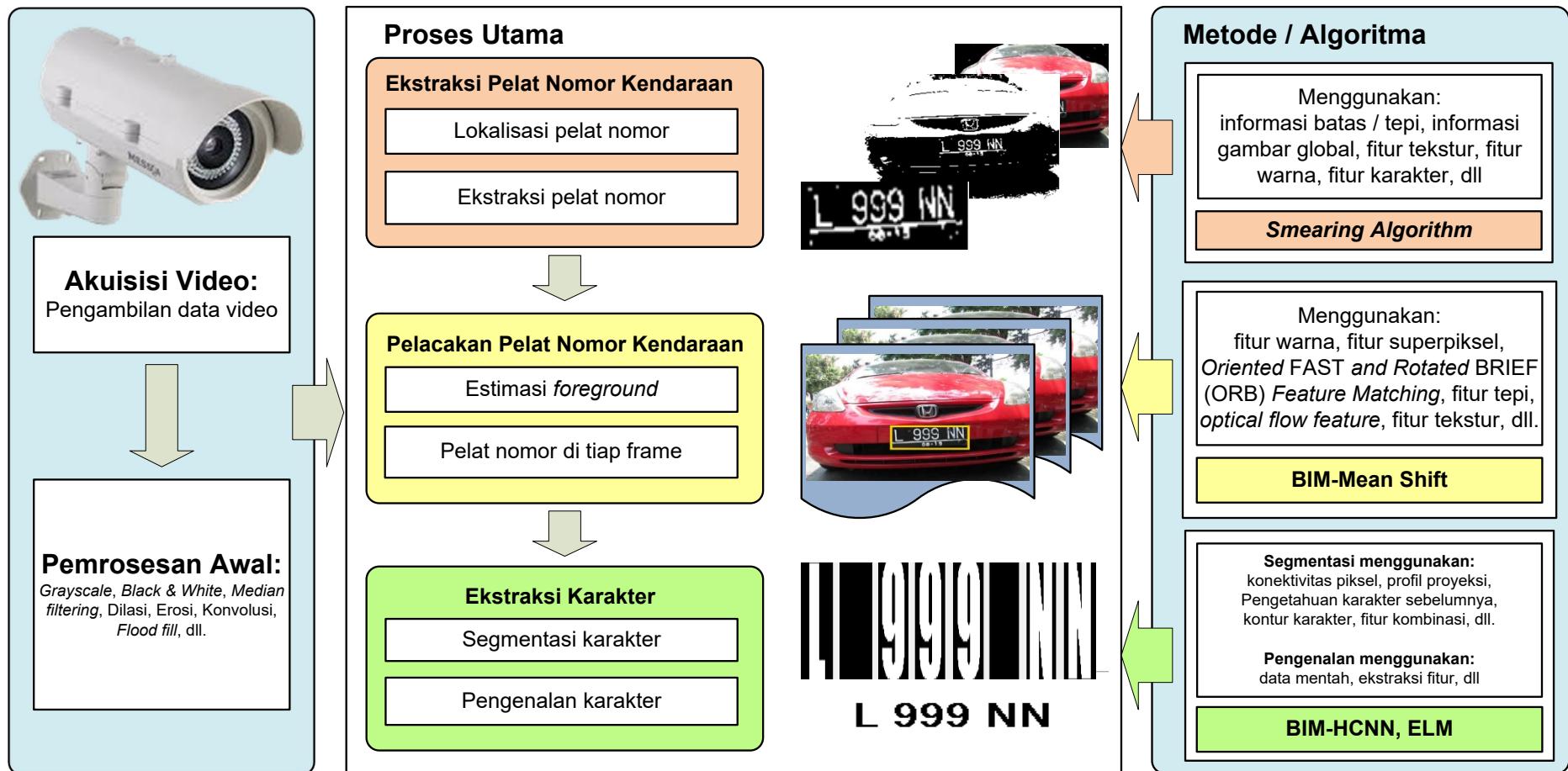
Bagian ekstraksi pelat nomor kendaraan, beberapa metode ekstraksi pelat nomor yang ada didasarkan pada fitur-fitur yang digunakan di dunia, antara lain: *Boundary/Edge Information*, *Global Image Information*, *Texture Features Color Features*, dan *Character Features* (Du *et al.*, 2013). Dalam penelitian ini digunakan metode *Smearing Algorithm* berdasarkan fitur-fitur yang sesuai (*Boundary/Edge Information* dan *Global Image Information*). Selama ini metode *Smearing Algorithm* sudah berhasil digunakan untuk proses ekstraksi pelat nomor kendaraan (Teo and Wong, 2009).

Bagian pelacakan pelat nomor kendaraan sepanjang *frame* video, beberapa metode pelacakan pelat nomor atau pelacakan objek secara umum yang ada didasarkan pada fitur-fitur yang digunakan di dunia, antara lain: *Color Feature* (Sahoo and Ari, 2015), (Owczarek, Baranski and Strumillo, 2015), *Superpixels Feature* (Chan, Zhou and Chen, 2015), *Oriented FAST and Rotated BRIEF (ORB) Feature Matching* (Yang *et al.*, 2015), *Edges Feature*, *Optical flow Feature*, dan *Texture Feature* (Owczarek, Baranski and Strumillo, 2015). Dalam penelitian ini digunakan metode BIM-Mean Shift yang termasuk dalam pengembangan *Improved Mean Shift* yang bekerja berdasarkan fitur yang sesuai (*Color Feature*). Pengembangan yang dilakukan adalah dengan meningkatkan kinerja algoritma Mean Shift standar dengan teknik *Switching Kernel* dan *Adaptive Fuzzy Gaussian Kernel*. Selama ini metode *Improved Mean Shift* sudah banyak digunakan untuk proses pelacakan objek berbasis video (Li *et al.*, 2015), (Setyawan, Purwanto and Mardiyanto, 2015), (Yang *et al.*, 2015).

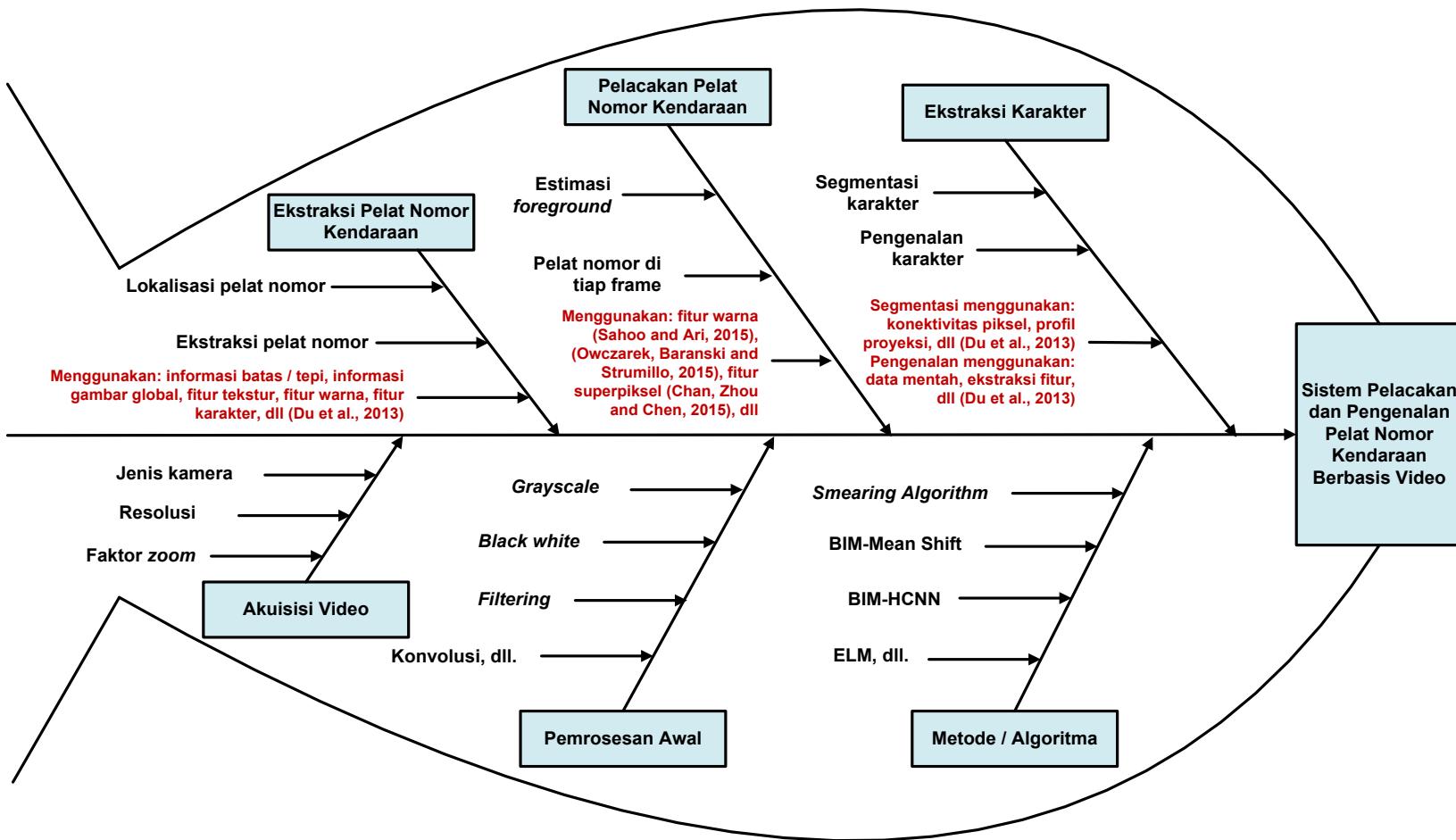
Bagian ekstraksi karakter yang terdapat pada pelat nomor kendaraan, dalam penelitian ini terdapat proses segmentasi karakter selain pengenalan karakter pelat nomor kendaraan. Untuk proses segmentasi karakter pelat nomor kendaraan, beberapa metode segmentasi karakter pelat nomor yang ada didasarkan pada fitur-fitur yang digunakan di dunia, antara lain: *Pixel Connectivity*, *Projection Profiles*, *Prior Knowledge of Characters*, *Character Contours*, dan *Combined Features* (Du *et al.*, 2013). Dalam penelitian ini digunakan metode BIM-HCNN, berdasarkan fitur yang sesuai (*Pixel Connectivity* berupa *Edge Detection* atau Deteksi Tepi). Selama ini metode HCNN sudah berhasil digunakan untuk proses segmentasi citra (Duraisamy and Jane, 2014), (Zeng *et al.*, 2014). Metode BIM-HCNN untuk segmentasi karakter pelat nomor kendaraan, adalah metode CNN yang ditingkatkan kinerjanya dengan cara dioptimasi dengan algoritma *Adaptive Fuzzy* dan *Neuro-Fuzzy*.

Sedangkan untuk proses pengenalan karakter pelat nomor kendaraan, beberapa metode pengenalan karakter pelat nomor yang ada didasarkan pada fitur-fitur yang digunakan di dunia, antara lain: *Raw Data*, dan *Extracted Features* (Du *et al.*, 2013). Dalam penelitian ini digunakan metode *Extreme Learning Machine* (ELM), berdasarkan fitur yang sesuai (*Extracted Features*). Metode ELM sudah banyak digunakan untuk pengenalan pola dan pengenalan karakter termasuk pengenalan karakter pelat nomor kendaraan (Saabni, 2015), (Gou *et al.*, 2014).

Dengan demikian, gambaran umum dari penelitian yang dilakukan oleh penulis seperti diperlihatkan pada Gambar 1.2. *Fishbone Diagram* yang menunjukkan posisi penelitian yang dilakukan oleh penulis terhadap penelitian-penelitian yang telah ada atau yang dilakukan sebelumnya oleh penulis maupun peneliti lain, diperlihatkan pada Gambar 1.3.



Gambar 1.2. Gambaran umum penelitian pelacakan dan pengenalan pelat nomor kendaraan berbasis video



Gambar 1.3. *Fishbone Diagram* posisi penelitian pelacakan dan pengenalan pelat nomor kendaraan berbasis video diantara penelitian yang ada saat ini didunia

1.7. Sistematika Penulisan Disertasi

Sistematika penulisan dari buku disertasi ini terdiri dari:

- Bab 1. Memuat hierarki pendahuluan yang terdiri dari latar belakang penelitian dilakukan, rumusan masalah, tujuan penelitian, manfaat penelitian, kontribusi penelitian yang dihasilkan dari kegiatan penelitian dan posisi penelitian.
- Bab 2. Membahas tentang Ekstraksi Pelat Nomor Kendaraan. Dimulai dengan pembahasan Teori Umum, Pemrosesan Awal (*Preprocessing*) yang terdiri dari: *Grayscale*, *Black and White*, *Median Filtering*, Dilasi, Erosi, Konvolusi, dan *Flood fill*. Kemudian diuraikan juga mengenai Fitur-fitur sebagai Dasar Ekstraksi Pelat Nomor Kendaraan, dan Ekstraksi Pelat Nomor Kendaraan dengan *Smearing Algorithm*.
- Bab 3. Membahas tentang Pelacakan Pelat Nomor Kendaraan. Yang dibahas dalam bab ini meliputi: Fitur-fitur Pelacakan Pelat Nomor Kendaraan, Estimasi Densitas, Mean Shift Dasar, Pelacakan Pelat Nomor Kendaraan dengan Mean Shift, Pelacakan Pelat Nomor Kendaraan dengan BIM-Mean Shift, dan Pengujian Pelacakan Pelat Nomor Kendaraan dengan BIM-Mean Shift.
- Bab 4. Membahas tentang Ekstraksi Karakter Pelat Nomor Kendaraan, yang terdiri dari Segmentasi Karakter Pelat Nomor Kendaraan dan Pengenalan Karakter Pelat Nomor Kendaraan. Diawali dengan pembahasan Fitur-fitur sebagai Dasar Ekstraksi Karakter Pelat Nomor Kendaraan, yang terdiri dari Fitur-fitur Segmentasi Karakter Pelat Nomor Kendaraan, dan Fitur-fitur Pengenalan Karakter Pelat Nomor Kendaraan. Dilanjutkan pembahasan Ekstraksi Karakter Pelat Nomor Kendaraan dan Pengenalan Karakter Pelat Nomor Kendaraan. Masing-masing didalamnya dibahas mengenai Segmentasi Karakter Pelat Nomor Kendaraan dengan BIM-HCNN dan Pengenalan Karakter Pelat Nomor Kendaraan dengan ELM.
- Bab 5. Berisikan kesimpulan dari kegiatan hasil penelitian yang telah dicapai dan rencana penelitian selanjutnya.

BAB 2

EKSTRAKSI PELAT NOMOR KENDARAAN

2.1. Teori Umum

Secara umum gambaran sistem pelacakan dan pengenalan pelat nomor kendaraan berbasis video seperti diperlihatkan pada Gambar 1.2 di Bab 1. Akuisisi Video dilakukan untuk mendapatkan data video rekaman kendaraan yang berjalan di jalan raya. Sebelum dilakukan Proses Utama sebelumnya dilakukan Pemrosesan Awal. Pemrosesan Awal yang umum dilakukan antara lain proses *Grayscale*, *Black and White*, *Median Filtering*, Dilasi, Erosi, Konvolusi, dan *Flood fill*. Selanjutnya Proses Utama terdiri dari Ekstraksi Pelat Nomor Kendaraan, Pelacakan Pelat Nomor Kendaraan, dan Ekstraksi Karakter Pelat Nomor Kendaraan berupa Segmentasi Karakter Pelat Nomor Kendaraan dan Pengenalan Karakter Pelat Nomor Kendaraan.

Setelah Pemrosesan Awal dan Proses Utama agar sistem dapat bekerja sesuai tujuan maka dibutuhkan metode. Metode dalam bentuk komputasi bisa berupa algoritma. Khusus untuk kebutuhan penelitian ini dibutuhkan algoritma sistem cerdas. Proses Utama yang terdiri dari Ekstraksi Pelat Nomor Kendaraan, Pelacakan Pelat Nomor Kendaraan, dan Ekstraksi Karakter Pelat Nomor Kendaraan yang berupa Segmentasi Karakter Pelat Nomor Kendaraan dan Pengenalan Karakter Pelat Nomor Kendaraan, semua bekerja berdasarkan fitur-fitur yang sesuai.

2.2. Pemrosesan Awal

2.2.1. *Grayscale*

Grayscale adalah berbagai nuansa warna monokromatik dari hitam menjadi putih. Oleh karena itu, gambar *grayscale* hanya memiliki warna keabuan dan tidak berwarna. Citra skala keabuan mempunyai nilai minimum (biasanya=0) dan nilai maksimum. Banyaknya kemungkinan nilai minimum dan maksimum bergantung pada jumlah bit yang digunakan (umumnya menggunakan 8 bit). Contohnya untuk skala keabuan 4 bit, maka jumlah kemungkinan nilainya adalah $2^4 = 16$, dan nilai maksimumnya adalah $2^4 - 1 = 15$, sedangkan untuk skala keabuan 8 bit, maka jumlah kemungkinan nilainya adalah $2^8 = 256$, dan nilai maksimumnya adalah $2^8 - 1 = 255$.

2.2.2. *Black and White*

Black and White dalam teori citra digital juga dikenal dengan citra biner yang hanya memiliki dua kemungkinan nilai piksel yaitu hitam dan putih. Citra B & W (*black and white*) dikenal juga citra monokrom. Citra *black and white* atau citra biner sering kali muncul sebagai hasil dari proses pengolahan seperti segmentasi, pengambangan, morphologi, ataupun *dithering*.

2.2.3. *Median Filtering*

Dalam citra digital, *Median filtering* (Filter median) adalah teknik penyaringan digital nonlinier, sering digunakan sebagai teknik pengurangan derau pada gambar. Pengurangan derau tersebut adalah langkah khas pemrosesan awal untuk meningkatkan hasil pengolahan selanjutnya (misalnya, deteksi tepi pada gambar).

2.2.4. Dilasi

Dalam teori citra digital ada istilah morfologi. Morfologi ini mempunyai dua operator dasar, yaitu Dilasi dan Erosi yang biasa digunakan untuk mengekstrak komponen yang diinginkan dalam sebuah citra. Morfologi adalah satu set luas operasi pengolahan citra berdasarkan bentuk. Operasi morfologi menerapkan *structuring element* ke gambar masukan, menciptakan keluaran gambar dengan ukuran yang sama. Dalam operasi morfologi, nilai setiap piksel dalam gambar keluaran berdasarkan perbandingan yang sesuai antara piksel pada gambar masukan dengan tetangga-tetangganya. Dengan memilih ukuran dan bentuk lingkungan, dapat dibangun sebuah operasi morfologi yang sensitif terhadap bentuk tertentu pada gambar masukan. Dilasi adalah nilai keluaran piksel maksimum dari semua masukan piksel tetangganya. Dalam gambar biner, jika salah satu piksel diatur ke nilai 1, maka piksel keluaran diatur ke 1.

2.2.5. Erosi

Erosi adalah operator dasar morfologi yang lain, yang memberikan nilai keluaran piksel minimum dari semua masukan piksel tetangganya. Dalam gambar biner, jika salah satu piksel diatur ke 0, maka keluaran piksel diatur ke 0.

2.2.6. Konvolusi

Konvolusi adalah sebuah proses manipulasi citra dengan menggunakan *external mask* atau *subwindows* untuk menghasilkan citra yang baru. Bedanya dengan *filtering* tanpa menggunakan *external mask* tetapi hanya menggunakan piksel tetangga untuk mendapatkan piksel yang baru. Konvolusi sangat banyak dipergunakan dalam pengolahan citra untuk memperhalus (*smoothing*), penajaman (*sharpening*), deteksi tepi (*edge detection*), serta efek lainnya.

2.2.7. Flood fill

Flood fill, adalah metode penentuan daerah yang terhubung ke node yang diberikan dalam *array* multi-dimensi. *Flood Fill* merupakan salah satu metode untuk keperluan *region labeling*. *Region labeling* adalah langkah penghitungan objek dalam sebuah citra, dengan mengutamakan identifikasi sebuah piksel dalam region tertentu. Dengan menggunakan metode ini, sistem dapat menghitung jumlah objek yang terdapat pada sebuah citra.

2.3. Fitur-fitur Ekstraksi Pelat Nomor Kendaraan

Ekstraksi pelat nomor kendaraan mempengaruhi keakuratan sistem pelacakan dan pengenalan pelat nomor kendaraan berbasis video. Proses ekstraksi pelat nomor kendaraan adalah mendapatkan lokasi pelat nomor kendaraan dari masukan berupa video atau pun gambar kendaraan. Plat bisa ada di mana saja di gambar. Alih-alih memproses setiap piksel dalam gambar, yang meningkatkan waktu pemrosesan, pelat dapat dibedakan oleh fitur-fiturnya. Karena itu proses sistem hanya pada piksel yang memiliki fitur ini. Fitur-fitur ini dihasilkan dari format dan karakter pelat. Warna pelat adalah salah satu fitur. Beberapa negara memiliki warna tertentu untuk pelat nomor kendaraan mereka. Bentuk persegi panjang dari batas pelat adalah fitur lain yang digunakan untuk mengekstraksi pelat nomor kendaraan. Perubahan warna antara karakter dan latar belakang pelat, yang dikenal sebagai tekstur, digunakan untuk mengekstraksi wilayah pelat dari gambar. Keberadaan karakter dapat digunakan sebagai fitur untuk mengidentifikasi wilayah pelat. Dua atau lebih banyak fitur dapat dikombinasikan untuk mengidentifikasi pelat nomor kendaraan.

Berikut ini, beberapa metode ekstraksi pelat nomor kendaraan yang ada didasarkan pada fitur-fitur yang digunakan (Du *et al.*, 2013).

- a. Ekstraksi pelat nomor kendaraan berdasarkan informasi batas/tepi (*License Plate Extraction Using Boundary/Edge Information*).
- b. Ekstraksi pelat nomor kendaraan berdasarkan informasi gambar keseluruhan (*License Plate Extraction Using Global Image Information*).
- c. Ekstraksi pelat nomor kendaraan berdasarkan fitur tekstur (*License Plate Extraction Using Texture Features*).
- d. Ekstraksi pelat nomor kendaraan berdasarkan fitur warna (*License Plate Extraction Using Color Features*).
- e. Ekstraksi pelat nomor kendaraan berdasarkan fitur karakter (*License Plate Extraction Using Character Features*).
- f. Ekstraksi pelat nomor kendaraan berdasarkan kombinasi dua atau lebih fitur (*License Plate Extraction Combining Two or More Features*).

Dari hasil-hasil penelitian sebelumnya berikut ini ringkasan perbandingan dari beberapa metode ekstraksi pelat yang ada didasarkan pada fitur-fitur yang digunakan. Perbandingan disajikan dalam penilaian kelebihan dan kekurangan dari metode masing-masing, seperti terlihat pada Tabel 2.1 (Du *et al.*, 2013).

2.4. Smearing Algorithm

Salah satu metode yang bisa digunakan untuk ekstraksi pelat nomor kendaraan adalah *Smearing Algorithm* (Teo and Wong, 2009). Algoritma *smearing* adalah metode untuk ekstraksi area karakter pada gambar pelat nomor kendaraan. Dengan algoritma *smearing* ini, gambar diproses sepanjang putaran horizontal dan vertikal sesuai ukuran panjang dan lebar gambar.

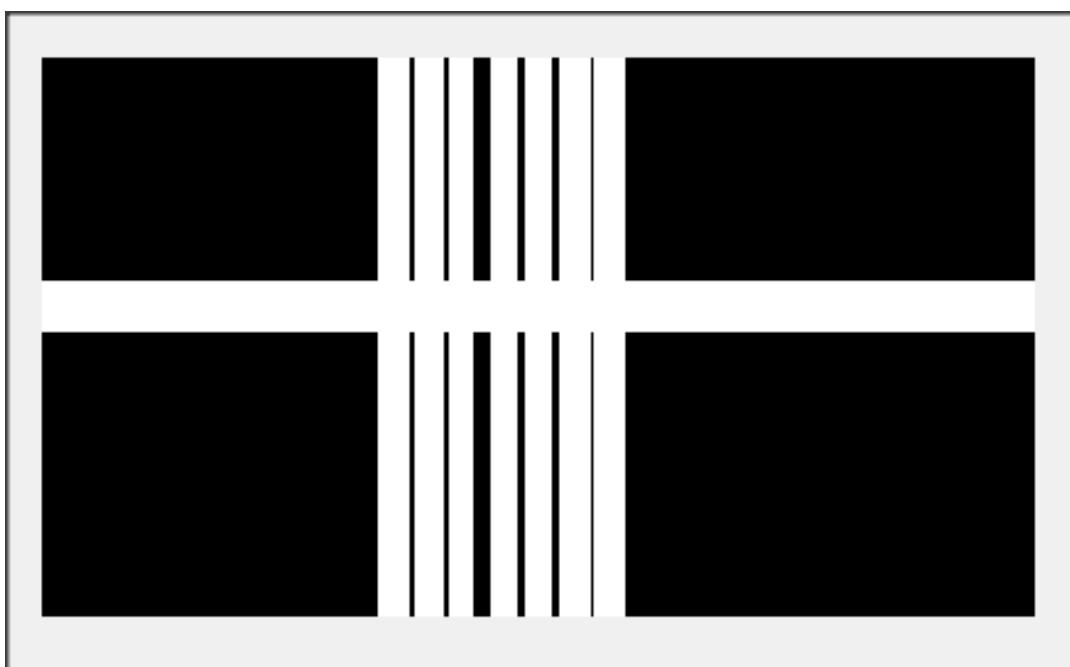
Metode yang dilakukan dengan cara memproses gambar menjadi garis horizontal dan vertikal untuk mengekstrak area hanya bagian tertentu dan menerjemahkan area tersebut menjadi warna putih dan hitam dari warna aslinya, dengan cara dibandingkan dengan nilai ambang batas (*threshold*). Jika ukuran piksel kurang dari nilai ambang batas yang diinginkan, maka piksel akan diubah menjadi warna putih. Sebaliknya jika lebih besar dari nilai ambang batas yang diinginkan, maka piksel akan diubah menjadi hitam. Dalam sistem ini, nilai ambang batas dipilih antara nilai 10 dan 100 untuk kedua tangkapan horizontal dan vertikal. Pemilihan nilai

ambang batas dilakukan secara empiris disesuaikan dengan data pelat nomor yang digunakan.

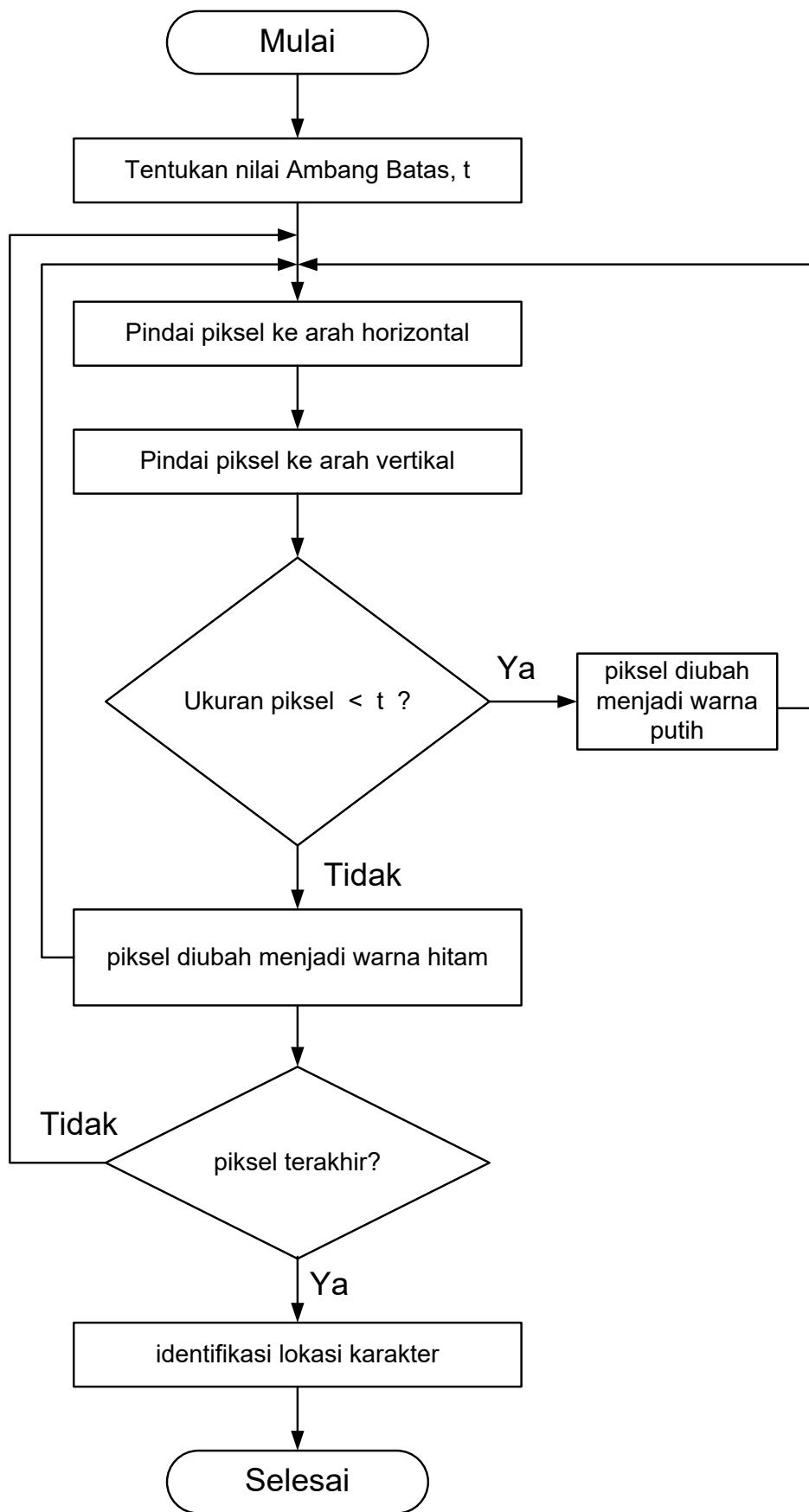
Tabel 2.1. Perbandingan metode ekstraksi pelat nomor kendaraan

Metode	Dasar Rasional	Kelebihan	Kekurangan
Menggunakan fitur batas (<i>boundary features</i>)	Batas pelat nomor persegi empat.	Sederhana, cepat dan lugas.	Hampir tidak diterapkan pada gambar kompleks karena terlalu sensitif terhadap tepi yang tidak diinginkan.
Menggunakan fitur gambar global (<i>global image features</i>)	Temukan objek terhubung yang dimensinya seperti pelat nomor.	Langsung, independen dari posisi pelat nomor.	Dapat menghasilkan objek yang rusak.
Menggunakan fitur tekstur (<i>texture features</i>)	Transisi warna yang sering terjadi pada pelat nomor.	Mampu mendeteksi bahkan jika batasnya cacat.	Komputasional kompleks bila ada banyak sisi.
Menggunakan fitur warna (<i>color features</i>)	Warna khusus pada pelat nomor.	Mampu mendeteksi pelat nomor miring dan cacat.	RGB terbatas pada kondisi iluminasi, HLS sensitif terhadap derau.
Menggunakan fitur karakter (<i>character features</i>)	Harus ada karakter di pelat nomor.	Tegar terhadap rotasi.	Mengkonsumsi waktu (memproses semua benda biner), menghasilkan kesalahan deteksi saat teks lain di gambar.
Menggunakan dua atau lebih fitur (<i>two or more features</i>)	Menggabungkan fitur lebih efektif.	Lebih andal.	Komputasional kompleks.

Proses *smearing* mengidentifikasi dan menghitung piksel putih, dan selanjutnya dioles dengan warna putih. Dilakukan sepanjang putaran horizontal dan vertikal sesuai ukuran panjang dan lebar gambar. Selanjutnya perpotongan pengolesan piksel putih searah horizontal dan vertikal, diidentifikasi sebagai lokasi karakter pelat nomor kendaraan. Setiap karakter diidentifikasi lokasi sebenarnya dan dipisahkan dari penggabungan dengan karakter yang lain. Contoh keluaran hasil proses *Smearing Algorithm* diperlihatkan pada Gambar 2.1. Sedangkan diagram alir *Smearing Algorithm* untuk keperluan ekstraksi pelat nomor kendaraan diperlihatkan pada Gambar 2.2.



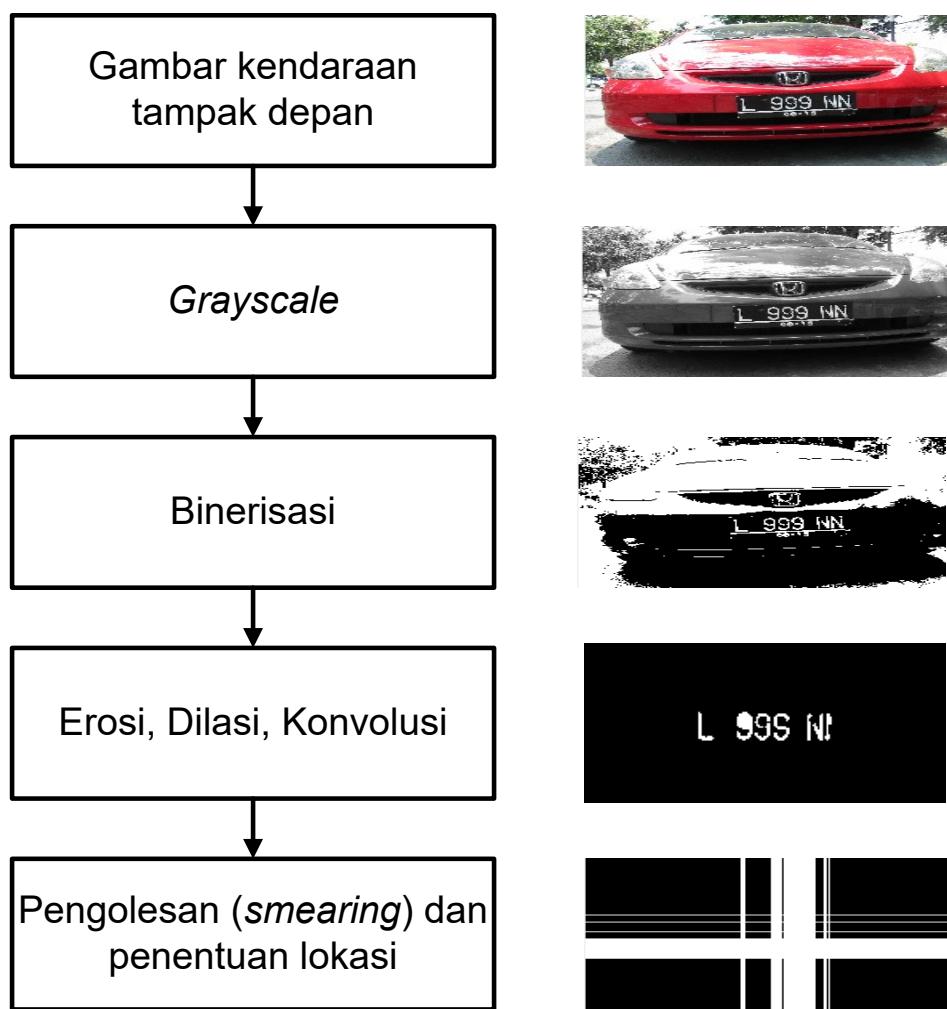
Gambar 2.1. Contoh keluaran *Smearing Algorithm*



Gambar 2.2. Diagram alir *Smearing Algorithm*

2.5. Ekstraksi Pelat Nomor Kendaraan dengan *Smearing Algorithm*

Algoritma *smearing* untuk kebutuhan ekstraksi pelat nomor kendaraan dilakukan setelah pemrosesan awal (*grayscale*, binerisasi, erosi, dilasi, dan konvolusi). Dengan demikian secara lengkap untuk kebutuhan ekstraksi pelat nomor kendaraan dengan metode *Smearing Algorithm* dirancang seperti diperlihatkan pada Gambar 2.3.



Gambar 2.3. Ekstraksi pelat nomor kendaraan dengan metode *Smearing Algorithm*

2.6. Hasil Ekstraksi Pelat Nomor Kendaraan dengan *Smearing Algorithm*

Berikut ini hasil-hasil pengujian metode *Smearing Algorithm* yang digunakan untuk ekstraksi pelat nomor kendaraan. Gambar 2.4 merupakan contoh gambar kendaraan yang akan diekstraksi pelat nomornya. Hasil proses *grayscale*, binerisasi, erosi, dilasi dan konvolusi, serta pengolesan (*smearing*) dan penentuan lokasi, masing-masing diperlihatkan pada Gambar 2.5 sampai dengan Gambar 2.8.



Gambar 2.4. Contoh gambar kendaraan



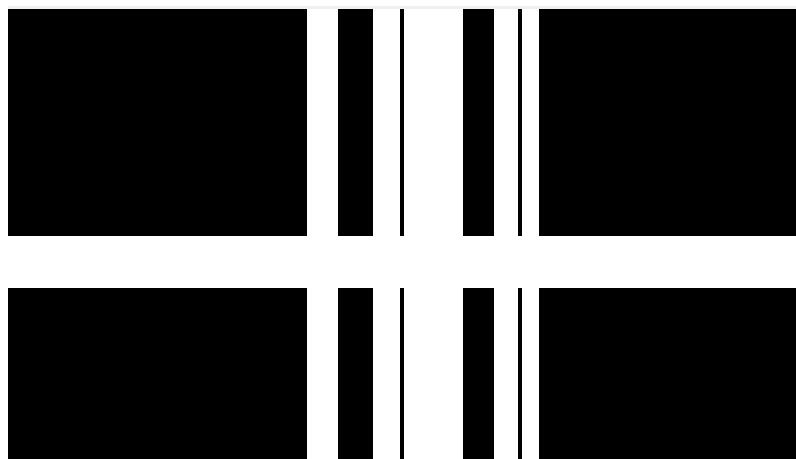
Gambar 2.5. Hasil proses grayscale



Gambar 2.6. Hasil proses binerisasi



Gambar 2.7. Hasil proses erosi, dilasi dan konvolusi



Gambar 2.8. Hasil proses pengolesan (*smearing*) dan penentuan lokasi

Selengkapnya hasil pengujian metode *Smearing Algorithm* yang digunakan untuk ekstraksi pelat nomor kendaraan, disajikan pada Tabel 2.2.

2.7. Ringkasan Hasil

Dari hasil pengujian dapat diperlihatkan bahwa metode *Smearing Algorithm* dapat digunakan sebagai salah satu metode ekstraksi pelat nomor kendaraan. Hasil ekstraksi pelat nomor kendaraan selanjutnya dapat digunakan untuk pengenalan pelat nomor kendaraan, dibahas di Bab 4.

Tabel 2.2. Hasil ekstraksi pelat nomor kendaraan

NO	Gambar JPG	Skala keabuan	Binerisasi	Erosi, Dilasi, Konvolusi	Pengolesan dan penentuan lokasi
1				L 900 D	
2				M 605 XN	
3				L 1003 RH	
4				L 1346 EO	
5				L 1783 P	
6				M 1233 RH	
7				L 500 RS	
8				L 999 KN	
9				L 1457 ES	
10				D 1367 MN	
11				L 5118 SV	
12				W 8881 LK	
13				B 8881 MN	
14				BK 238 CTI	
15				W 882 PZ	

Halaman ini sengaja dikosongkan

BAB 3

PELACAKAN PELAT NOMOR KENDARAAN

3.1. Fitur-fitur Pelacakan Pelat Nomor Kendaraan

Pelacakan video (*video tracking*) adalah pelacakan objek atau multi-objek dalam serangkaian urutan frame-frame video. Pelacakan objek adalah salah satu masalah mendasar yang menantang dalam aplikasi visi komputer karena kesulitan dalam pelacakan objek bisa muncul karena faktor intrinsik dan ekstrinsik seperti deformasi, gerak kamera, *motion blur* dan oklusi (Reddy, Priya and Neelima, 2015). Tantangan utama yang harus diperhitungkan saat merancang sistem pelacakan objek adalah terkait dengan kesamaan penampilan antara target dan benda-benda lain di tempat kejadian yang direkam dalam video, dan termasuk variasi penampilan target itu sendiri.

Berikut ini, beberapa metode pelacakan pelat nomor kendaraan atau pun pelacakan objek secara umum yang ada didasarkan pada fitur-fitur yang digunakan, antara lain:

- a. Fitur Warna (*Color Feature*) (Sahoo and Ari, 2015), (Owczarek, Baranski and Strumillo, 2015).
- b. Fitur Superpixsel (*Superpixels Feature*) (Chan, Zhou and Chen, 2015).
- c. *Oriented FAST and Rotated BRIEF (ORB) Feature Matching* (Yang *et al.*, 2015).
- d. Fitur Tepi, Fitur aliran Optik, dan Fitur Tekstur (*Edges Feature, Optical flow Feature, and Texture Feature*) (Owczarek, Baranski and Strumillo, 2015).

Sebelum dibahas lebih lanjut mengenai pelacakan pelat nomor kendaraan berbasis video, terlebih dahulu perlu dipahami beberapa pengetahuan penting seputar estimasi densitas, antara lain *Probability Density Function* (PDF), Estimasi Densitas Kernel atau *Kernel Density Estimation* (KDE), Fungsi Kernel dan pemilihan lebar pita atau radius kernel.

3.2. Estimasi Densitas

3.2.1. *Probability Density Function (PDF)*

Definisi matematis dari fungsi probabilitas kontinyu, $p(s)$, memenuhi sifat-sifat berikut (Hong, 2012):

1. Probabilitas bahwa s adalah antara dua titik a dan b .

$$p(a < s < b) = \int_a^b p(s) ds \quad (3.1)$$

2. Bernilai tidak negatif untuk semua s riil.
3. Integral dari fungsi probabilitas adalah satu, yaitu:

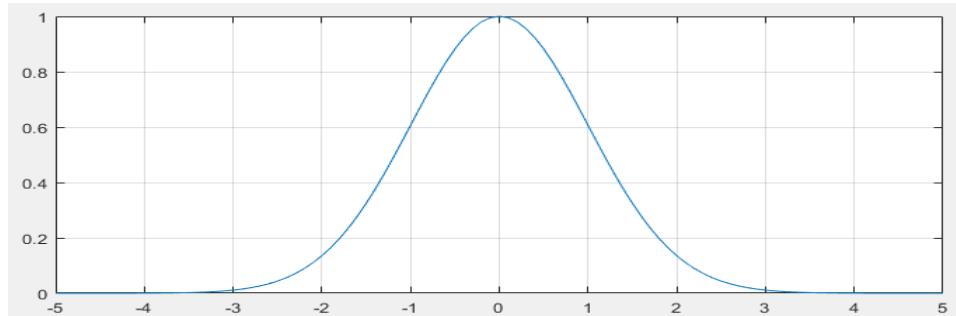
$$\int_{-\infty}^{\infty} p(s) ds = 1 \quad (3.2)$$

Fungsi probabilitas yang paling umum digunakan adalah fungsi Gaussian (juga dikenal sebagai distribusi normal), yaitu:

$$p(s) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(s-c)^2}{2\sigma^2}\right) \quad (3.3)$$

Dimana c adalah rata-rata, σ^2 adalah varian dan σ adalah standar deviasi.

Contoh gambaran untuk $c=0$ dan $\sigma = 1$ seperti terlihat pada Gambar 3.1.



Gambar 3.1. PDF Gaussian dengan $c = 0$, $\sigma = 1$

Memperluas ke kasus vektor s , dimiliki $p(s)$ non-negatif dengan sifat-sifat berikut (Hong, 2012):

1. Probabilitas bahwa s berada di dalam wilayah \mathfrak{R} yaitu:

$$P = \int_{\mathfrak{R}} p(s) ds \quad (3.4)$$

2. Integral dari fungsi probabilitas adalah satu, yaitu:

$$\int p(s) ds = 1 \quad (3.5)$$

Dengan sekumpulan n sampel data s_1, \dots, s_n , dapat diperkirakan fungsi densitas $p(s)$, sehingga diperoleh keluaran $p(s)$ untuk setiap sampel s baru. Ini disebut estimasi densitas. Ide dasar di balik banyak metode untuk memperkirakan probabilitas fungsi densitas yang tidak diketahui sangat sederhana. Teknik yang paling mendasar bergantung pada fakta bahwa probabilitas P bahwa sebuah vektor jatuh di suatu wilayah \mathcal{R} seperti diberikan oleh Persamaan (3.4).

Jika diasumsikan bahwa \mathcal{R} begitu kecil sehingga $p(s)$ tidak banyak berbeda di dalamnya, maka bisa dituliskan:

$$P = \int_{\mathcal{R}} p(s) ds \approx p(s) \int_{\mathcal{R}} ds = p(s)V \quad (3.6)$$

Dimana V adalah volume dari \mathcal{R} .

Di sisi lain, misalkan n sampel s_1, \dots, s_n ditarik secara independen sesuai dengan fungsi probabilitas densitas $p(s)$, dan ada k dari n sampel yang berada di dalam wilayah \mathcal{R} , maka diperoleh hubungan:

$$P = k/n \quad (3.7)$$

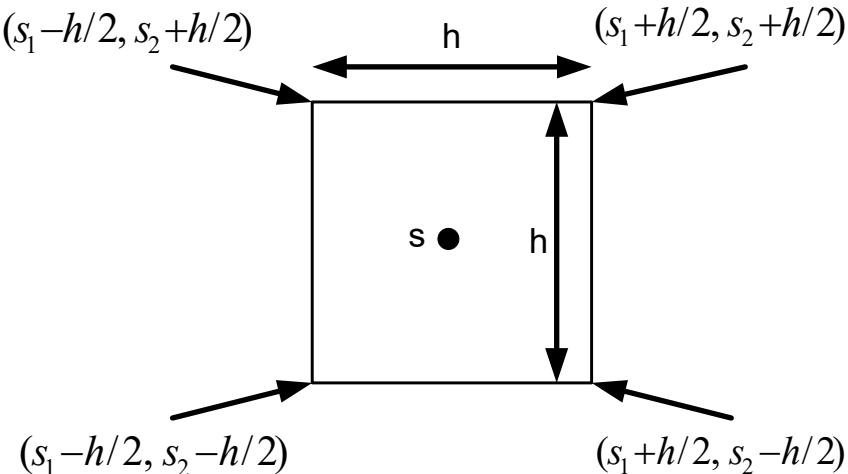
Jadi sampai pada perkiraan yang jelas berikut untuk $p(s)$, yaitu:

$$p(s) = \frac{k}{V} \quad (3.8)$$

3.2.2. Estimasi Densitas Kernel

Dalam statistik, estimasi densitas kernel atau *Kernel Density Estimation* (KDE) adalah cara non-parametrik untuk memperkirakan PDF dari variabel acak. Estimasi densitas kernel adalah masalah perataan data dasar dimana kesimpulan tentang populasi dilakukan, berdasarkan sampel data yang terbatas. Di beberapa bidang seperti pemrosesan sinyal dan ekonometrik, metode ini juga disebut metode Parzen–Rosenblatt *window*, setelah Emanuel Parzen dan Murray Rosenblatt, yang biasanya dikreditkan dengan pembuatannya secara independen dalam bentuknya saat ini (Rosenblatt, 1956), (Parzen, 1962).

Selanjutnya untuk memahami estimasi densitas kernel, dipertimbangkan bahwa \mathcal{R} adalah *hypercube* yang berpusat pada s (persegi 2-D), seperti terlihat pada Gambar 3.2. Misalkan h adalah panjang tepi *hypercube*, maka $V = h^2$ untuk persegi 2-D, dan $V = h^3$ untuk sebuah kubus 3-D (Hong, 2012).



Gambar 3.2. Hypercube persegi 2-D

Diperkenalkan sebuah persamaan (Hong, 2012):

$$g\left(\frac{s-s_i}{h}\right) = \begin{cases} 1 & \frac{|s_k - s_{ik}|}{h} \leq 1/2, k=1,2 \\ 0 & \text{lainnya} \end{cases} \quad (3.9)$$

Menunjukkan apakah s_i berada di dalam kotak (terpusat pada s , dengan lebar h) atau tidak. Jumlah sampel k yang jatuh di dalam wilayah \mathfrak{R} , dari n , diberikan oleh:

$$k = \sum_{i=1}^n g\left(\frac{s-s_i}{h}\right) \quad (3.10)$$

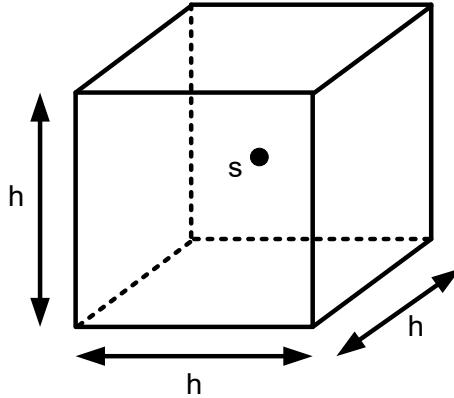
Sehingga dengan menggunakan Persamaan (3.8), dan $V = h^2$ untuk persegi 2-D, maka rumus estimasi densitas probabilitas Parzen (untuk 2-D) diperoleh (Hong, 2012):

$$\begin{aligned} p(s) &= \frac{k/n}{V} \\ &= \frac{1}{n} \sum_{i=1}^n \frac{1}{h^2} g\left(\frac{s-s_i}{h}\right) \end{aligned} \quad (3.11)$$

$g\left(\frac{s-s_i}{h}\right)$ dinamakan *window function* (Hong, 2012).

Kemudian untuk *hypercube* kubus 3-D, diasumsikan bahwa daerah \mathfrak{R} yang membungkus sampel k dari n , adalah *hypercube* dengan sisi panjang h berpusat pada s , seperti terlihat pada Gambar 3.3, dengan volume $V = h^3$. Kemudian jumlah total titik di dalam *hypercube*, seperti pada Persamaan (3.10). Dengan menggunakan Persamaan (3.8), dan $V = h^3$ untuk kubus 3-D, maka rumus estimasi densitas probabilitas Parzen (untuk 3-D) diperoleh:

$$\begin{aligned}
p(s) &= \frac{k/n}{V} \\
&= \frac{1}{n} \sum_{i=1}^n \frac{1}{h^3} g\left(\frac{s-s_i}{h}\right)
\end{aligned} \tag{3.12}$$



Gambar 3.3. Hypercube kubus 3-D

Sehingga secara umum dengan dimensi D , maka rumus estimasi densitas probabilitas Parzen diperoleh (Gutierrez-Osuna, 2013):

$$p(s) = \frac{1}{n h^D} \sum_{i=1}^n g\left(\frac{s-s_i}{h}\right) \tag{3.13}$$

Sebelumnya telah disebutkan, bahwa $g((s-s_i)/h)$ dinamakan *window function*. Dalam prakteknya, $g(.)$ dalam *window function* ini, biasanya digunakan fungsi kernel. Kernel ini, yang sesuai dengan unit *hypercube* yang berpusat pada titik asal, dikenal sebagai Parzen *window* atau naïve *estimator* (Gutierrez-Osuna, 2013). Kuantitas $g((s-s_i)/h)$ sama dengan kesatuan jika s_i berada di dalam *hypercube* dengan sisi h berpusat pada s , dan sama dengan nol jika berada di luar *hypercube*. Dengan $g(.)$ dalam *window function* menggunakan fungsi kernel, maka estimasi densitas probabilitas Parzen dengan dimensi D pada Persamaan (3.13) juga disebut estimasi densitas kernel atau *Kernel Density Estimation (KDE)* (Gutierrez-Osuna, 2013). Estimasi densitas kernel dinyatakan seperti pada Persamaan (3.14). Sisi h secara umum nantinya jika *window function* menggunakan fungsi kernel, maka disebut dengan lebar pita (*bandwidth*) atau radius kernel.

$$\begin{aligned}
p_{KDE} &= p(s) \\
&= \frac{1}{n h^D} \sum_{i=1}^n g\left(\frac{s-s_i}{h}\right)
\end{aligned} \tag{3.14}$$

3.2.3. Fungsi Kernel

Estimasi PDF dengan menggunakan KDE selalu melibatkan fungsi kernel. Fungsi kernel $g(\cdot)$ memenuhi sifat fungsi probabilitas kontinyu, yaitu bernilai riil tidak negatif, dan integral dari fungsi probabilitasnya adalah satu:

$$\int_{-\infty}^{\infty} g(u) du = 1 \quad (3.15)$$

Beberapa fungsi kernel yang banyak digunakan antara lain:

1. Seragam

$$g(u) = \begin{cases} 1/2, & \text{untuk } |u_i| \leq 1 \\ 0, & \text{untuk } |u_i| \text{ yang lain} \end{cases} \quad (3.16)$$

2. Segitiga

$$g(u) = \begin{cases} 1 - |u|, & \text{untuk } |u_i| \leq 1 \\ 0, & \text{untuk } |u_i| \text{ yang lain} \end{cases} \quad (3.17)$$

3. Epanechnikov

$$g(u) = \begin{cases} 3/4(1-u^2), & \text{untuk } |u_i| \leq 1 \\ 0, & \text{untuk } |u_i| \text{ yang lain} \end{cases} \quad (3.18)$$

4. Gaussian

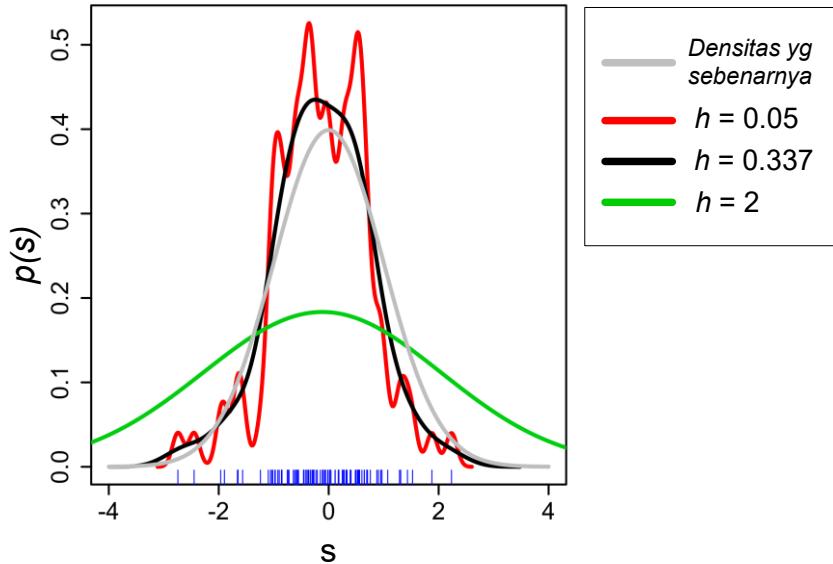
$$g(u) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}u^2\right), \text{ untuk } |u_i| < \infty \quad (3.19)$$

3.2.4. Pemilihan Lebar Pita atau Radius Kernel

Telah disebutkan sebelumnya, dari Persamaan (3.14) terdapat parameter h dalam estimasi PDF dengan menggunakan KDE. Parameter h ini sangat penting dan berpengaruh terhadap keberhasilan estimasi PDF. Parameter h dari fungsi kernel ini dinamakan lebar pita atau *bandwidth* kernel (Comaniciu, Ramesh and Meer, 2003), (Chen *et al.*, 2012) atau kadang juga disebut radius kernel (Sahoo and Ari, 2015).

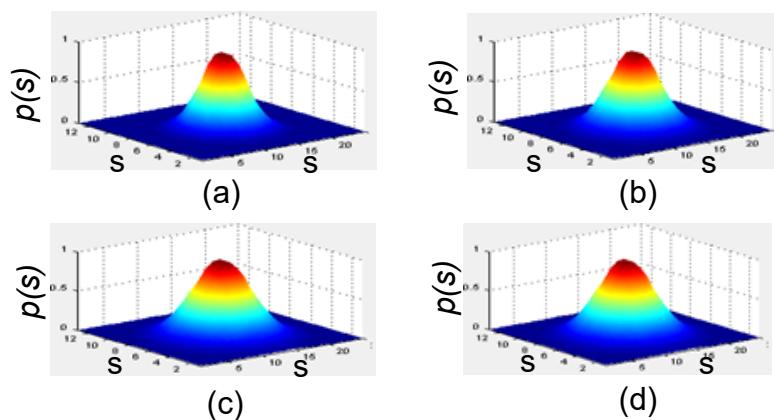
Untuk mengilustrasikan pengaruh parameter h terhadap keberhasilan estimasi PDF, diambil sampel acak dari distribusi normal standar seperti pada Persamaan (3.3) diplotkan pada sumbu horizontal, seperti diperlihatkan pada Gambar 3.4. Kurva abu-abu adalah densitas sebenarnya (kerapatan normal dengan rata-rata 0 dan varian 1). Sebagai perbandingan, kurva merah terlihat kurang bagus karena mengandung terlalu banyak artefak data palsu yang timbul dari penggunaan lebar pita atau radius kernel h

$= 0,05$, yang terlalu kecil. Kurva hijau *oversmooth* karena menggunakan lebar pita atau radius kernel $h = 2$ mengaburkan banyak struktur yang mendasarinya. Kurva hitam dengan lebar pita atau radius kernel $h = 0,337$ dianggap diperhalus secara optimal karena perkiraan densitasnya mendekati densitas sebenarnya.



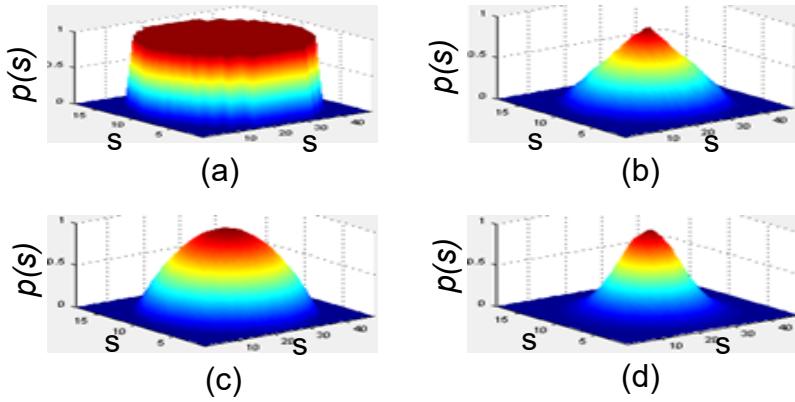
Gambar 3.4. Pengaruh lebar pita atau radius kernel

Jika lebar pita atau radius kernel h dinyatakan dalam persen, contoh gambaran dalam 3-D pengaruh lebar pita atau radius kernel terhadap bentuk topeng *Parzen window kernel* (*the mask of a Parzen window kernel*), dengan menggunakan lebar pita atau radius kernel masing-masing, $h = 65\%$, 75% , 85% dan 95% seperti diperlihatkan pada Gambar 3.5.



Gambar 3.5. Bentuk topeng *Parzen window kernel* dengan menggunakan fungsi kernel Gaussian masing-masing dengan lebar pita atau radius kernel, (a) $h = 65\%$, (b) $h = 75\%$, (c) $h = 85\%$ dan (d) $h = 95\%$

Contoh gambaran bentuk topeng *Parzen window kernel* menggunakan Fungsi Kernel: Seragam, Segitiga, Epanechnikov dan Gaussian, masing-masing dengan lebar pita atau radius kernel $h = 85\%$ seperti terlihat pada Gambar 3.6.



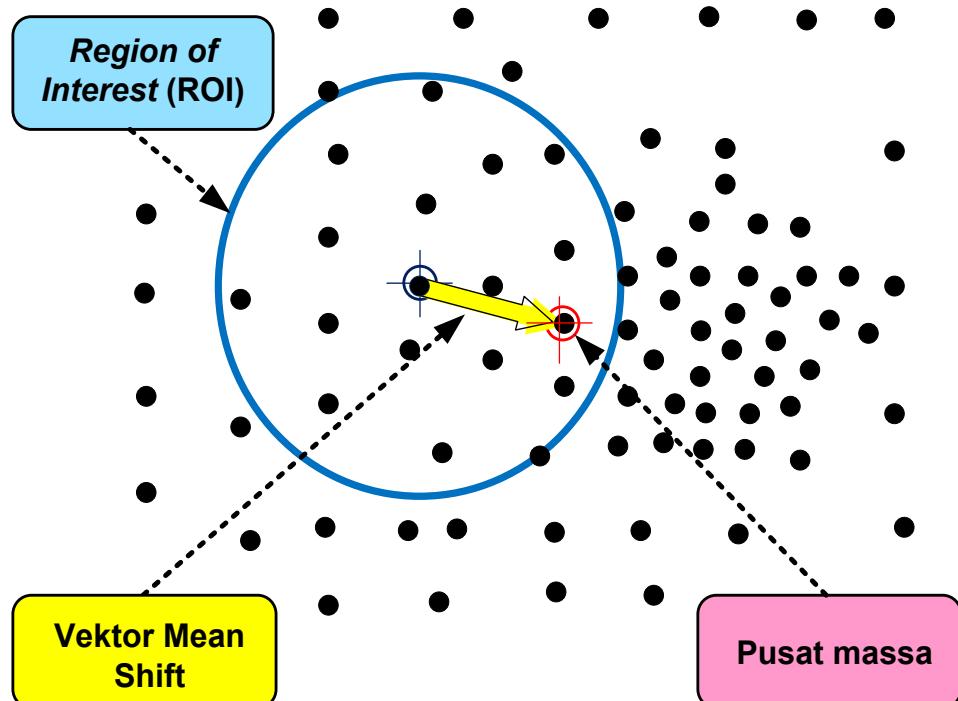
Gambar 3.6. Bentuk topeng *Parzen window kernel* dengan menggunakan fungsi kernel:
(a) Seragam (b) Segitiga (c) Epanechnikov (d) Gaussian, masing-masing dengan lebar pita atau radius kernel $h = 85\%$

3.3. Mean Shift Dasar

Mean Shift adalah metode estimasi non-parametrik awalnya diusulkan oleh Fukunaga dan Hostetler pada tahun 1975 (Fukunaga and Hostetler, 1975). Dikombinasikan dengan fungsi kesamaan (*similarity function*), dapat digunakan untuk mendapatkan nilai pencocokan optimal antara model objek dan model kandidat yang masing-masing dinyatakan dalam bentuk PDF (Comaniciu, Ramesh and Meer, 2003).

Untuk memahami cara kerja Mean Shift, diasumsikan secara intuitif seperti terdapat pada distribusi bola bilyar identik seperti terlihat pada Gambar 3.7 (Ukrainitz and Sarel, 2004). Tujuannya adalah menemukan daerah terpadat atau densitas distribusi bola bilyar tertinggi. Misalkan, dipilih lokasi sembarang, yaitu daerah yang menjadi pusat perhatian atau *Region of Interest* (ROI) seperti pada gambar. Dari posisi di titik pusat ROI, pusat massa dari distribusi bola diketahui. Selanjutnya dari titik pusat ROI bergeser ke arah pusat massa tersebut. Meninggalkan jejak vektor Mean Shift. Pusat massa distribusi bola sebelumnya sekarang menjadi titik pusat ROI. Dari posisi di titik pusat ROI yang baru ini, pusat massa dari distribusi bola kembali diketahui. Selanjutnya dari titik pusat ROI yang baru tadi bergeser lagi ke arah pusat massa yang baru tersebut. Meninggalkan jejak vektor Mean Shift lagi. Demikian seterusnya sampai titik pusat ROI sama dengan pusat massa distribusi bola, sehingga

tidak ada lagi pergeseran. Artinya densitas tertinggi dari distribusi bola bilyar tertinggi telah berhasil ditemukan.



Gambar 3.7. Distribusi bola bilyar identik

Pergeseran berarti mempertimbangkan ruang fitur sebagai PDF empiris. Jika masukan adalah satu set titik maka Mean shift menganggapnya sebagai sampel dari PDF yang mendasarinya. Jika daerah padat (atau *cluster*) ada di ruang fitur, maka berarti sesuai dengan mode (atau maksima lokal) dari PDF-nya. Kita juga bisa mengidentifikasi kelompok yang terkait dengan mode yang diberikan menggunakan Mean Shift.

Untuk setiap titik data, Mean shift mengaitkannya dengan puncak terdekat dari PDF dataset. Untuk setiap titik data, Mean shift menentukan sebuah *window* di sekitarnya dan menghitung rata-rata dari titik data. Kemudian menggeser pusat *window* ke rata-rata tersebut. Diulang-ulang sampai konvergen. Setiap iterasi, dapat dipertimbangkan bahwa *window* bergeser ke daerah dataset yang lebih padat.

Dengan demikian, cara kerja Mean Shift sebagai berikut:

1. Perbaiki *window* di sekitar setiap titik data.
2. Dihitung rata-rata data di dalam *window*.
3. Geser *window* ke arah rata-rata tersebut.
4. 1-3 diulang-ulang sampai konvergen.

3.4. Pelacakan Pelat Nomor Kendaraan dengan Mean Shift

Pada pelacakan pelat nomor kendaraan dengan Mean Shift, fungsi kesamaan digunakan untuk menilai kesamaan antara PDF model objek dan PDF model kandidat. Untuk pelacakan objek, teknik Mean Shift bekerja berdasarkan fitur warna. PDF sebagai fungsi probabilitas fitur warna diperoleh melalui Parzen *window* dengan cara konversi dari warna *Red, Green, Blue* (RGB) ke warna yang berindeks.

Probabilitas fitur warna u dari model objek q_u , dan probabilitas fitur warna u dari model kandidat $p_u(s)$, dapat dinyatakan seperti pada Persamaan (3.20) dan (3.21) (Ukrainitz and Sarel, 2004).

$$q_u = C \sum_{b(s_i)=u} g(\|s_i\|^2) \quad (3.20)$$

$$p_u(s) = C_h \sum_{b(s_i)=u} g\left(\left\|\frac{s-s_i}{h}\right\|^2\right) \quad (3.21)$$

Keterangan:

- u : fitur warna.
- s : lokasi pusat objek (target) pada frame video sebelumnya.
- s_i : lokasi piksel dalam kernel pada frame video saat ini.
- $g(\cdot)$: fungsi kernel.
- C : faktor normalisasi model objek.
- C_h : faktor normalisasi model kandidat.
- $b(s_i)$: indeks warna (1..m) dari s_i .
- h : lebar pita atau radius kernel.
- q_u : probabilitas fitur warna u dari model objek.
- $p_u(s)$: probabilitas fitur warna u dari model kandidat.

Probabilitas fitur warna u dari model objek q_u , dan probabilitas fitur warna u dari model kandidat $p_u(s)$, kadang juga dinyatakan dengan fungsi delta Kronecker δ seperti dalam persamaan berikut (Sahoo and Ari, 2015):

$$q_u = C \sum_{b(s_i)=u} g(\|s_i\|^2) \delta[b(s_i)-u] \quad (3.22)$$

$$p_u(s) = C_h \sum_{b(s_i)=u} g\left(\left\|\frac{s-s_i}{h}\right\|^2\right) \delta[b(s_i)-u] \quad (3.23)$$

Dimana, δ adalah fungsi delta Kronecker.

Hubungan antara kedua PDF yang tidak lain probabilitas fitur warna ini dapat dinyatakan dengan fungsi kesamaan $f_1(s)$ atau koefisien Bhattacharyya $f_2(s)$ seperti pada Persamaan (3.24) (Sahoo and Ari, 2015) dan (3.25) (Comaniciu, Ramesh and Meer, 2003) berikut:

$$f_1(s) = \sum_{u=1}^m \sqrt{q_u / p_u(s)} \quad (3.24)$$

$$f_2(s) = \sum_{u=1}^m \sqrt{p_u(s) q_u} \quad (3.25)$$

Potongan kecil (*patch*) dari kandidat akan digeser menuju Mean Shift secara berulang-ulang untuk memaksimalkan parameter kesamaan. Mean Shift memiliki iterasi konvergen lokal yang nilai konvergensi meningkat seiring kenaikan densitas dari distribusi probabilitas yang diberikan. Proses iterasi diulang terus menerus sampai densitas tertinggi dari estimasi optimal dari lokasi objek (target) tercapai, ini mengindikasikan jika objek berhasil dilacak.

Komputasi atau perhitungan vektor Mean Shift untuk menerjemahkan Kernel *window* dinyatakan dalam $m(s)$, diberikan seperti pada Persamaan (3.26) (Chen *et al.*, 2012) berikut ini.

$$m(s) = \left[\frac{\sum_{i=1}^n s_i g\left(\frac{\|s - s_i\|^2}{h}\right)}{\sum_{i=1}^n g\left(\frac{\|s - s_i\|^2}{h}\right)} - s \right] \quad (3.26)$$

Keterangan:

- $m(s)$: vektor Mean Shift yang diinginkan dalam satu iterasi dalam kernel.
- s : lokasi pusat objek (target) pada frame video sebelumnya.
- s_i : lokasi piksel dalam kernel pada frame video saat ini.
- $g(.)$: fungsi kernel.
- h : lebar pita atau radius kernel.

Selanjutnya untuk mendapatkan kemiringan gradien Mean Shift didasarkan pada kemiringan gradien kontur densitas. Rumus generik untuk kemiringan gradien adalah (Thirumuruganathan, 2010):

$$s_1 = s_0 + \eta f'(s_0) \quad (3.27)$$

Jika diterapkan ke persamaan KDE seperti pada Persamaan (3.14) maka akan diperoleh hubungan sebagai berikut.

$$p_{KDE} = \frac{1}{n h^D} \sum_{i=1}^n g\left(\frac{s-s_i}{h}\right)$$

$$\nabla p_{KDE} = \frac{1}{n h^D} \sum_{i=1}^n g'\left(\frac{s-s_i}{h}\right)$$

Di-setting ke 0 maka kita dapatkan (Thirumuruganathan, 2010):

$$\sum_{i=1}^n g'\left(\frac{s-s_i}{h}\right) \vec{s} = \sum_{i=1}^n g'\left(\frac{s-s_i}{h}\right) \vec{s}_i$$

Sehingga, diperoleh kemiringan gradien Mean Shift sebagai berikut (Thirumuruganathan, 2010):

$$\vec{s} = \frac{\sum_{i=1}^n g'\left(\frac{s-s_i}{h}\right) \vec{s}_i}{\sum_{i=1}^n g'\left(\frac{s-s_i}{h}\right)} \quad (3.28)$$

Dalam komputasi untuk mendapatkan topeng Parzen *window* dan gradiennya terhadap sumbu x dan sumbu y, dengan macam-macam fungsi kernel yang digunakan: Seragam, Segitiga, Epanechnikov, Gaussian, adalah dengan menggunakan persamaan fungsi Parzen_window seperti pada Persamaan (3.29) (Bernhardt, 2016).

$$[km, gx, gy] = \text{Parzen_window}(tm, lm, h, g(.), graph) \quad (3.29)$$

Keterangan:

- km : topeng (*mask*) Parzen *window*.
- gx, gy : gradien Parzen window terhadap sumbu x dan y.
- tm, lm : tinggi dan lebar ukuran topeng Parzen *window*.
- h : lebar pita atau radius kernel.
- $g(.)$: fungsi kernel.
- $graph$: plot gambar topeng (*mask*) jika $graph=1$.

Akhirnya untuk mendapatkan pelacakan Mean Shift adalah dengan cara menggunakan persamaan fungsi Meansif_tracking seperti pada Persamaan (3.30) (Bernhardt, 2016).

$$\begin{aligned}
 [s_i, loss, f, f_idx] = & \text{MeanShift_tracking} (q_u, I2, Lmap, \dots \\
 & height, width, f_thresh, max_it, s, \dots \\
 & tm, lm, km, gx, gy, f, f_idx, loss)
 \end{aligned} \tag{3.30}$$

Keterangan:

- s : lokasi pusat objek (target) pada frame video sebelumnya.
- s_i : lokasi piksel dalam kernel pada frame video saat ini.
- $loss$: objek (target) keluar dari pelacakan.
- (f, f_idx) : penyimpanan hasil fungsi kesamaan selama proses pelacakan.
- q_u : probabilitas fitur warna u dari model objek.
- $I2$: frame berikutnya.
- $height, width$: ukuran $I2$.
- $Lmap$: Panjang colormap.
- f_thresh : nilai ambang dari fungsi kesamaan.
- max_it : iterasi maksimal.
- tm, lm : tinggi dan lebar ukuran topeng Parzen window.
- km : topeng (mask) Parzen window.
- gx, gy : gradien Parzen window terhadap sumbu x dan y.

3.5. Pelacakan Pelat Nomor Kendaraan dengan BIM-Mean Shift

Dalam perkembangannya algoritma Mean Shift banyak diteliti untuk ditingkatkan kinerjanya menjadi *Improved Mean Shift*. Mean Shift dapat disesuaikan dan ditingkatkan kinerjanya dalam pemecahan berbagai permasalahan pelacakan objek berbasis video. Salah satu diantaranya untuk peningkatan kinerja Mean Shift dilakukan dengan cara pemanfaatan kontras antara objek dan latar belakang serta evaluasi kesamaan dalam rangka menghasilkan dan memperbarui model objek (Li *et al.*, 2015).

Peningkatan Mean Shift lainnya telah dilakukan dengan menggunakan deteksi objek dengan pemodelan melalui bintang skeletonisasi (*skeletonization star*), diikuti dengan algoritma pencarian blok dan penanganan oklusi (Sahoo and Ari, 2015). Perbaikan lain telah dilakukan dengan menggunakan *Background Corrected Weighted Histogram* yang diikutkan didalam model obyektif untuk mengurangi inferensi latar

belakang. Metode skala-adaptif diikutkan didalam Mean Shift untuk penskalaan dan deteksi oklusi kemudian ditangani oleh *Scaled Normalized Cross Correlation* (Setyawan, Purwanto and Mardiyanto, 2015).

Beberapa penelitian lain telah berhasil meningkatkan kinerja Mean Shift dengan menambahkan algoritma atau teknis lainnya, antara lain: (Maggio and Cavallaro, 2005; Mokti and Salam, 2008; He and Zhu, 2009; Linlin *et al.*, 2009; Azghani *et al.*, 2010; q. Li and z. Chen, 2011; Khattak *et al.*, 2014; Yang *et al.*, 2015; Li *et al.*, 2015; Sahoo and Ari, 2015; Setyawan, Purwanto and Mardiyanto, 2015).

Dalam penelitian ini peningkatan algoritma Mean Shift menjadi *Improved Mean Shift* dilakukan dengan cara penambahan teknik *Switching Kernel* dan *Adaptive Fuzzy Gaussian Kernel*. Untuk selanjutnya *Improved Mean Shift* dengan dua teknik ini disebut dengan Basuki-I Ketut-Mauridhi (BIM) Mean Shift (BIM-Mean Shift).

3.5.1. BIM-Mean Shift dengan *Switching Kernel*

Pelacakan pelat nomor kendaraan dengan BIM-Mean Shift dengan *Switching Kernel* dapat diuraikan sebagai berikut. Peningkatan kinerja Mean Shift standar dilakukan dengan cara pemanfaatan mekanisme *Switching Kernel* yang didasarkan pada nilai fungsi kesamaan (*similarity function*) antara PDF model objek dan PDF model kandidat. Mekanisme *Switching Kernel* didasarkan pertimbangan bahwa kernel adalah bagian yang sangat penting dan berpengaruh pada saat pembangkitan PDF model objek dan PDF model kandidat.

Parameter-parameter penting penentu keberhasilan proses pelacakan objek secara umum atau dalam hal ini pelat nomor kendaraan antara lain: fungsi kernel, lebar pita atau radius kernel, dan fungsi kesamaan (*similarity function*). Baik fungsi kernel maupun lebar pita atau radius kernel perubahannya diharapkan tetap bisa menjaga agar nilai kesamaan tetap atau meningkat dibandingkan dengan nilai tertentu, ε .

Switching Kernel berarti bahwa fungsi kernel yang bervariasi atau dapat berganti-ganti secara otomatis selama proses pelacakan sepanjang *frame* video. Fungsi kernelnya tidak statis seperti yang biasanya dilakukan selama ini. Kapan fungsi kernel harus diubah dengan fungsi kernel lain, yaitu ketika rata-rata keluaran dari fungsi kesamaan (*mean of the similarity function output*) lebih kecil dibandingkan dengan nilai tertentu, ε .

Penentuan nilai tertentu, ε , diperoleh dari nilai rata-rata keluaran fungsi kesamaan hasil proses pelacakan menggunakan Mean Shift standar dengan fungsi kernel Gaussian. Dengan Z sebagai jumlah *frame*, dan $f_1(s)$ sebagai fungsi kesamaan, maka nilai ε didefinisikan seperti Persamaan (3.31):

$$\varepsilon = \frac{\sum_{i=1}^Z f_1(s)_i}{Z} \quad (3.31)$$

Keterangan:

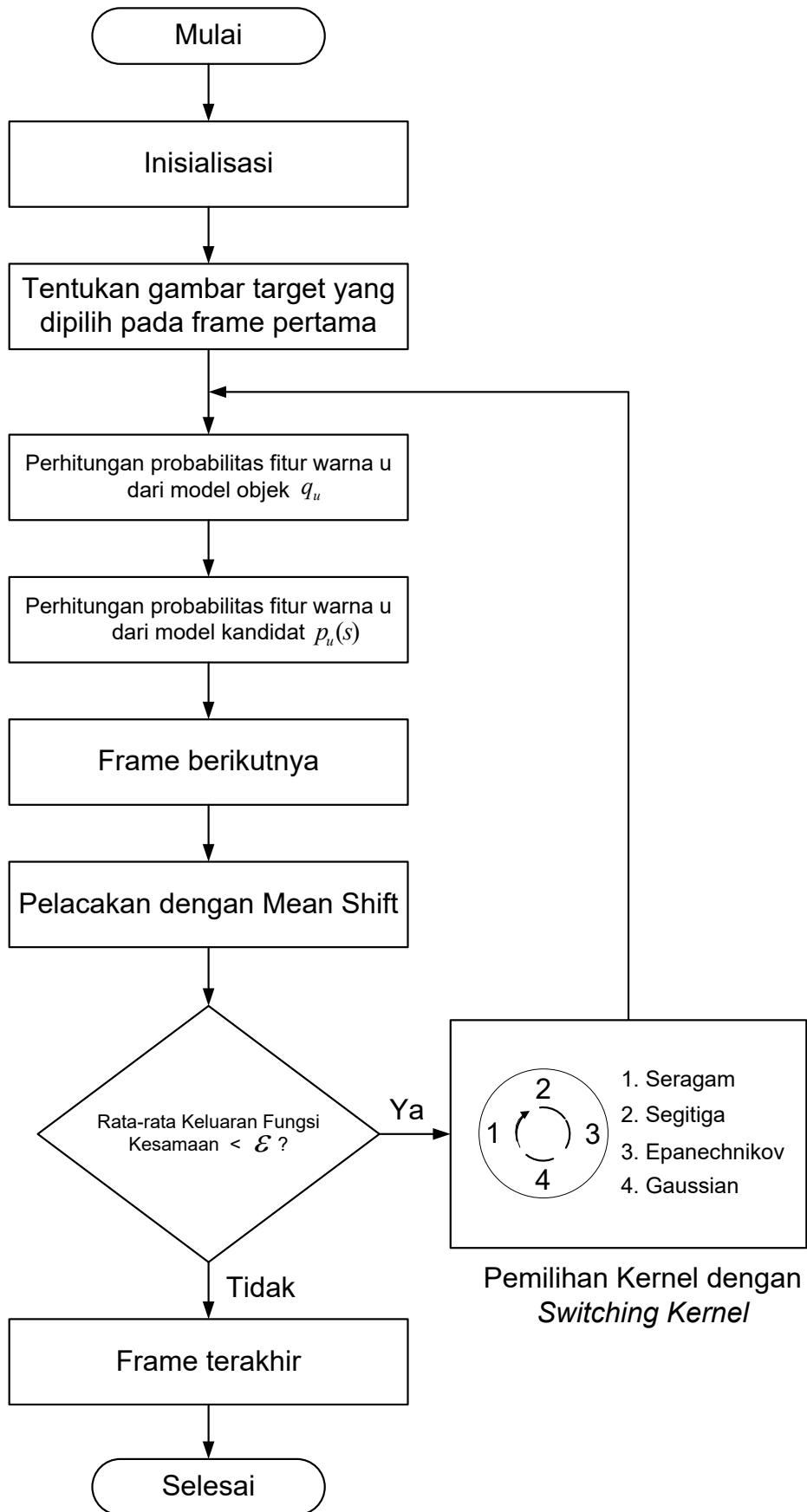
ε : rata-rata keluaran fungsi kesamaan hasil proses pelacakan menggunakan Mean Shift standar dengan fungsi kernel Gaussian.

Z : jumlah *frame*.

Sehingga diagram alir pelacakan pelat nomor kendaraan menggunakan BIM-Mean Shift dengan *Switching Kernel* ini seperti diperlihatkan pada Gambar 3.8.

3.5.2. BIM-Mean Shift dengan *Adaptive Fuzzy Gaussian Kernel*

Pelacakan pelat nomor kendaraan dengan BIM-Mean Shift dengan *Adaptive Fuzzy Gaussian Kernel*, idenya hampir sama dengan penggunaan *Switching Kernel*. Telah diuraikan sebelumnya, beberapa penentu keberhasilan proses pelacakan objek secara umum atau dalam hal ini pelat nomor kendaraan adalah fungsi kernel, lebar pita atau radius kernel dan fungsi kesamaan (*similarity function*). Fokus kali ini, adalah lebar pita atau radius kernel. Perubahan lebar pita atau radius kernel diharapkan tetap bisa menjaga agar nilai kesamaan tetap atau meningkat dibandingkan dengan nilai tertentu, ε . Definisi nilai ε sama dengan Persamaan (3.31). Kenaikan atau penurunan nilai kesamaan terhadap nilai ε dijadikan dasar untuk menambah atau mengurangi lebar pita atau radius kernel. Jika dinyatakan dalam bentuk *error* dan *delta error* dan diberi aturan fuzzy, dengan fungsi kernel yang digunakan yaitu fungsi kernel Gaussian, maka teknis ini dinamakan *Adaptive Fuzzy Gaussian Kernel*.



Gambar 3.8. BIM-Mean Shift dengan *Switching Kernel*

Diagram alir pelacakan pelat nomor kendaraan menggunakan BIM-Mean Shift dengan *Adaptive Fuzzy Gaussian Kernel* ini seperti diperlihatkan pada Gambar 3.9. Perancangan sistem fuzzy dari BIM-Mean Shift dengan *Adaptive Fuzzy Gaussian Kernel* untuk pelacakan pelat nomor kendaraan Indonesia ini diuraikan sebagai berikut. Sistem fuzzy yang digunakan menggunakan istilah-istilah berikut: *small error* (SE), *medium error* (ME), *big error* (BE), *small delta error* (SDE), *medium delta error* (MDE) dan *big delta error* (BDE), dan fungsi keanggotaan fuzzy (μ) yang digunakan yaitu Fungsi Gaussian. Sehingga masing-masing fungsi keanggotaannya dapat didefinisikan sebagai berikut:

$$\begin{aligned}\mu_{SE} &= \text{gaussian}(\text{error}; \sigma_1, c_1) & \mu_{SDE} &= \text{gaussian}(\text{delta error}; \sigma_4, c_4) \\ \mu_{ME} &= \text{gaussian}(\text{error}; \sigma_2, c_2) & \mu_{MDE} &= \text{gaussian}(\text{delta error}; \sigma_5, c_5) \\ \mu_{BE} &= \text{gaussian}(\text{error}; \sigma_3, c_3) & \mu_{BDE} &= \text{gaussian}(\text{delta error}; \sigma_6, c_6)\end{aligned}\quad (3.32)$$

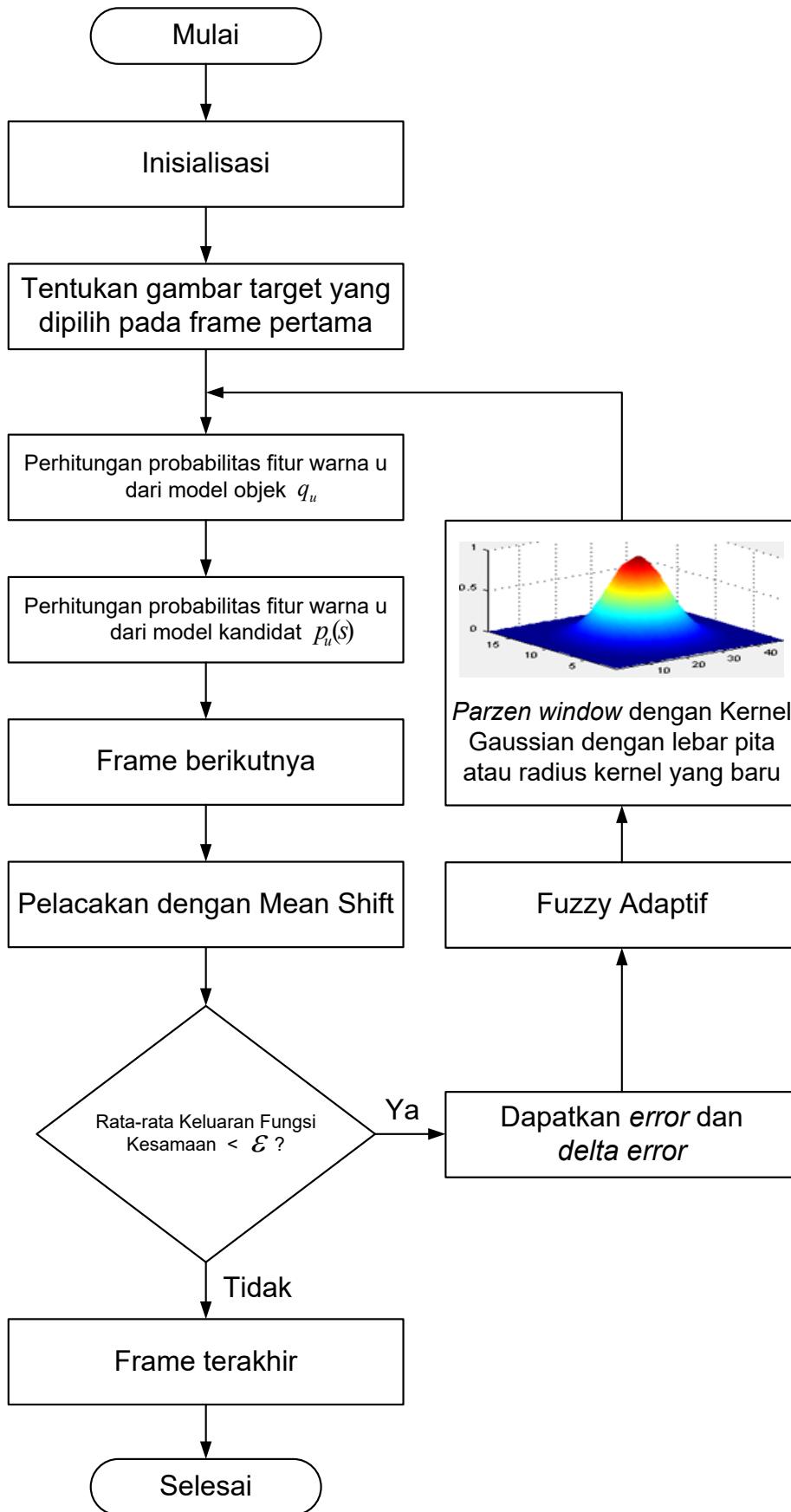
Persamaan Fungsi keanggotaan Gaussian dengan parameter fungsi keanggotaan σ_i, c_i dengan n_e sebagai *error* atau *delta error*, dinyatakan seperti pada Persamaan (3.33).

$$\text{gaussian}(n_e; \sigma_i, c_i) = \exp\left(-\frac{(n_e - c_i)^2}{2\sigma_i^2}\right), \text{ dimana } i=1..6 \quad (3.33)$$

Nilai-nilai parameter fungsi keanggotaan σ_i, c_i masing-masing didefinisikan sebagai Persamaan (3.34) dan (3.35).

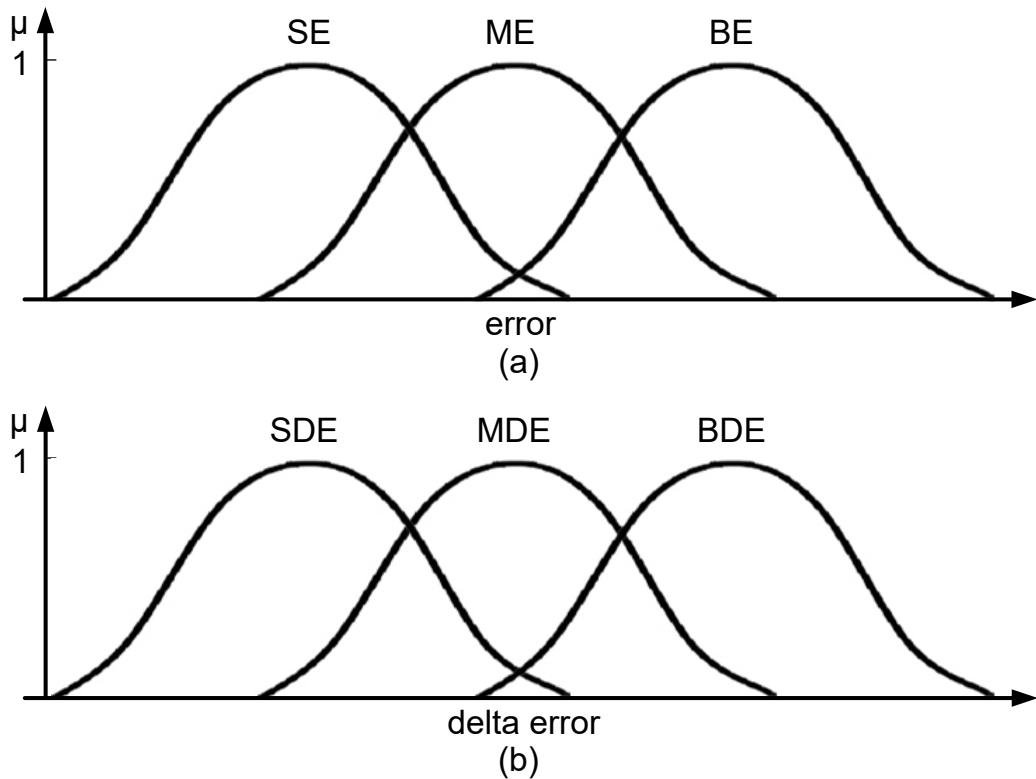
$$\begin{aligned}\sigma_1 &= \max(\text{error}); & c_1 &= \max(\text{error}) / 3; \\ \sigma_2 &= \max(\text{error}); & c_2 &= \max(\text{error}) / 2;\end{aligned}\quad (3.34)$$

$$\begin{aligned}\sigma_3 &= \max(\text{error}); & c_3 &= \max(\text{error}) / 1; \\ \sigma_4 &= \max(\text{delta error}); & c_4 &= \max(\text{delta error}) / 3; \\ \sigma_5 &= \max(\text{delta error}); & c_5 &= \max(\text{delta error}) / 2; \\ \sigma_6 &= \max(\text{delta error}); & c_6 &= \max(\text{delta error}) / 1;\end{aligned}\quad (3.35)$$



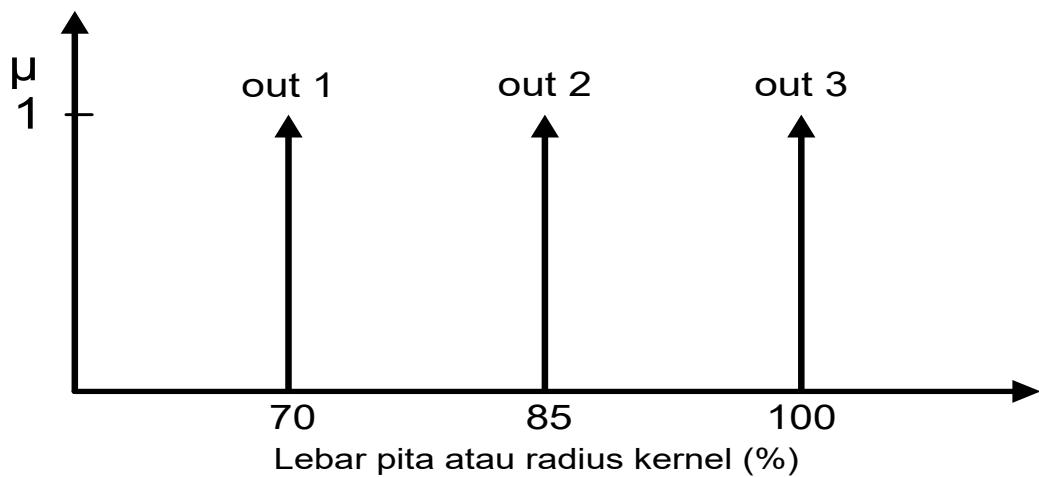
Gambar 3.9. BIM-Mean Shift dengan *Adaptive Fuzzy Gaussian Kernel*

Bentuk fungsi keanggotaan fuzzy masukan, seperti ditunjukkan pada Gambar 3.10.



Gambar 3.10. Fungsi keanggotaan fuzzy masukan *error* dan *delta error*

Sedangkan bentuk fungsi keanggotaan fuzzy keluaran, dirancang seperti ditunjukkan pada Gambar 3.11.



Gambar 3.11. Fungsi keanggotaan fuzzy keluaran

Sesuai dengan Gambar 3.11, nilai crisp dari masing-masing out1, out2 and ou3 didefinisikan seperti pada Persamaan (3.36). Berupa singleton untuk output Fuzzy berupa pilihan nilai radius dari *Parzen window* menggunakan Kernel Gaussian (%).

$$out1=70; \quad out2=85; \quad out3=100; \quad (3.36)$$

Dan aturan sistem fuzzy (*if then rules*) yang digunakan seperti terlihat pada Tabel 3.1.

Tabel 3.1. Aturan Sistem Fuzzy

		Error		
		SE	ME	BE
Delta Error	SDE	out3	out2	out1
	MDE	out3	out2	out1
	BDE	out3	out2	out1

3.6. Pengujian Pelacakan Pelat Nomor Kendaraan dengan BIM-Mean Shift

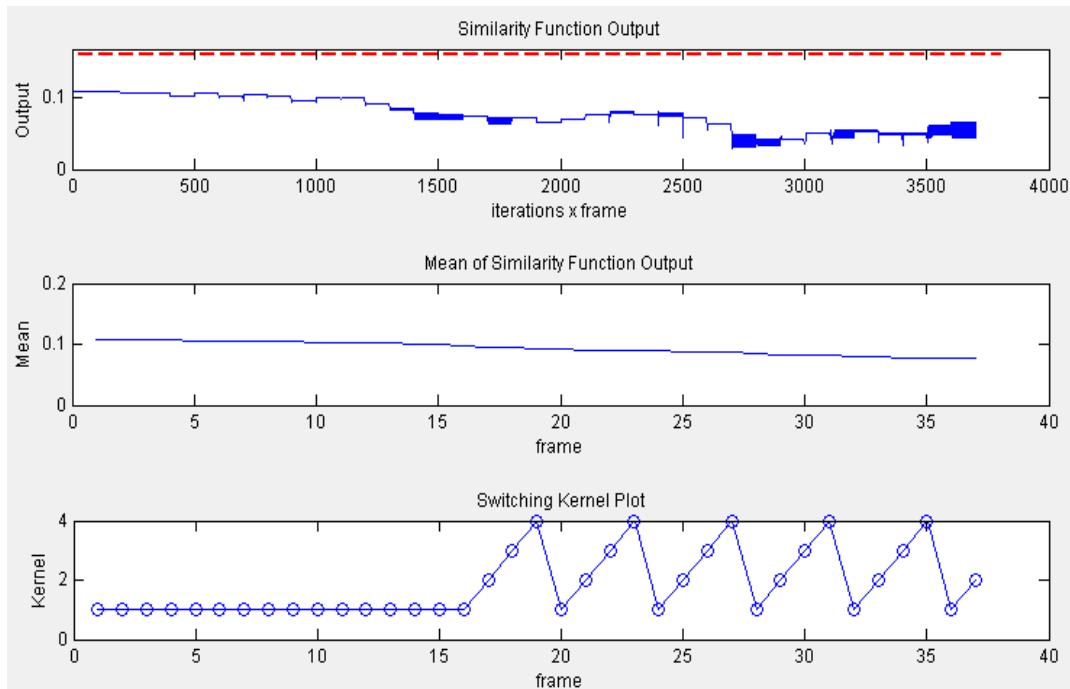
3.6.1. Pengujian BIM-Mean Shift dengan *Switching Kernel*

Algoritma BIM-Mean Shift dengan *Switching Kernel* diuji terhadap 5 rekaman video dari kendaraan di jalan raya. Algoritma ini diterapkan dalam rangka untuk melacak pelat nomor kendaraan Indonesia. Parameter yang digunakan untuk pengujian, terdiri dari nilai ambang fungsi kesamaan dan jumlah iterasi maksimal, seperti diperlihatkan pada Tabel 3.2.

Tabel 3.2. Parameter Mean Shift

Parameter	Nilai
Nilai ambang fungsi kesamaan	0.16
Jumlah iterasi maksimal	100

Hasil simulasi ditunjukkan seperti terlihat pada gambar-gambar berikut. Untuk Video 1, diperoleh hasil-hasil pengujian seperti terlihat pada Gambar 3.12 dan Gambar 3.13. Bentuk keluaran fungsi kesamaan, rata-rata keluaran fungsi kesamaan dan plot *Switching Kernel* dari hasil data Video 1 diperlihatkan pada Gambar 3.12. Hasil proses pelacakan Video 1 menggunakan Fungsi Kernel: Seragam, Segitiga, Epanechnikov, Gaussian dan *Switching Kernel* berdasarkan hasil pelacakan pada frame ke-16 dan ke-32, ditampilkan pada Gambar 3.13.

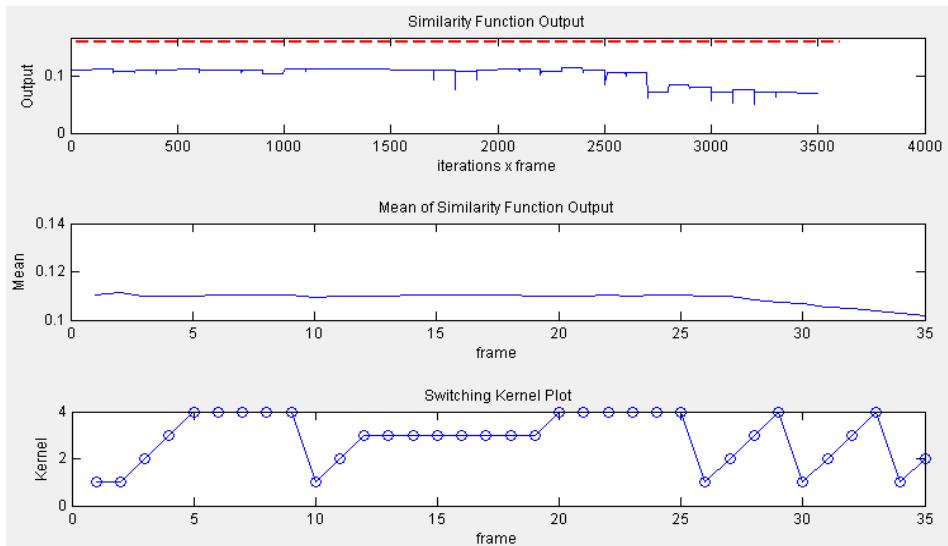


Gambar 3.12. Bentuk keluaran fungsi kesamaan, rata-rata keluaran fungsi kesamaan dan plot *Switching Kernel* dari hasil data Video 1

Data Video 2, hasil pengujinya ditunjukkan seperti pada Gambar 3.14 dan Gambar 3.15. Bentuk keluaran fungsi kesamaan, rata-rata keluaran fungsi kesamaan dan plot *Switching Kernel* dari hasil data Video 2, diperlihatkan pada Gambar 3.14. Hasil proses pelacakan Video 2 menggunakan Fungsi Kernel: Seragam, Segitiga, Epanechnikov, Gaussian dan *Switching Kernel* berdasarkan hasil pelacakan pada frame ke-16 dan ke-32, diperlihatkan pada Gambar 3.15.



Gambar 3.13. Hasil proses pelacakan Video 1 (dari atas ke bawah) menggunakan Fungsi Kernel: Seragam, Segitiga, Epanechnikov, Gaussian dan *Switching Kernel*. Hasil pelacakan pada frame ke-16 dan ke-32.

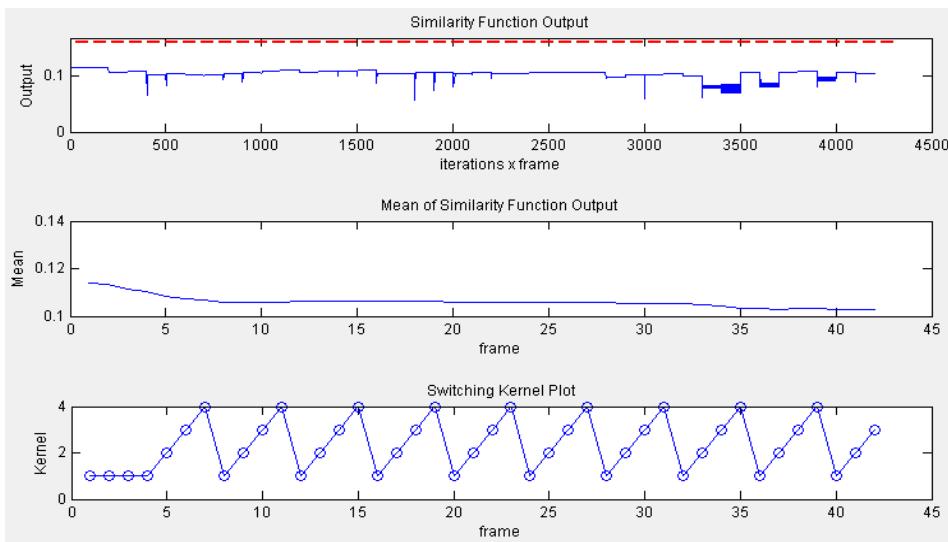


Gambar 3.14. Bentuk keluaran fungsi kesamaan, rata-rata keluaran fungsi kesamaan dan plot *Switching Kernel* dari hasil data Video 2



Gambar 3.15. Hasil proses pelacakan Video 2 (dari atas ke bawah) menggunakan Fungsi Kernel: Seragam, Segitiga, Epanechnikov, Gaussian dan *Switching Kernel*. Hasil pelacakan pada frame ke-16 dan ke-32.

Data Video ke-3, proses pelacakan pelat nomor kendaraan yang paling sukses di antara video lainnya. Warna mobil yang tampak hitam merata terlihat menguntungkan, karena memperkecil *clutter*. Meskipun pelat nomor kendaraan juga hitam, namun fitur warna pelat nomor masih bisa dibedakan. Hasil pengujian pelacakan pelat nomor kendaraan seperti ditampilkan pada Gambar 3.16 dan Gambar 3.17. Bentuk keluaran fungsi kesamaan, rata-rata keluaran fungsi kesamaan dan plot *Switching Kernel* dari hasil data Video 3, diperlihatkan pada Gambar 3.16. Hasil proses pelacakan Video 3 menggunakan Fungsi Kernel: Seragam, Segitiga, Epanechnikov, Gaussian dan *Switching Kernel* berdasarkan hasil pelacakan pada frame ke-20 dan ke-40, diperlihatkan pada Gambar 3.17.

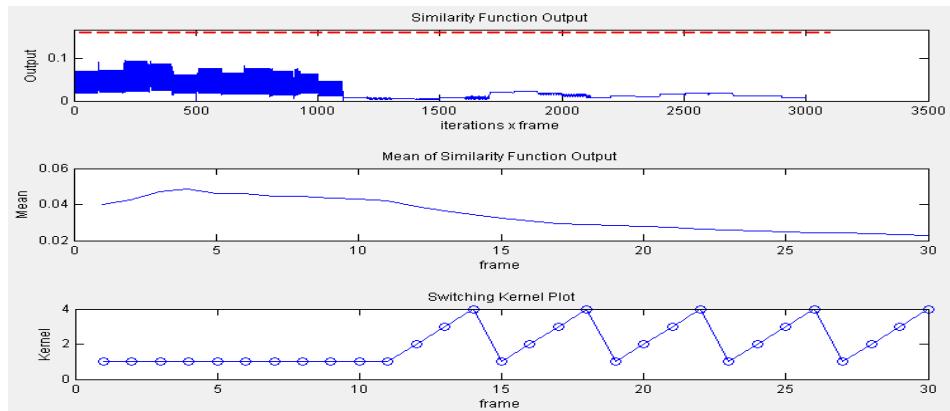


Gambar 3.16. Bentuk keluaran fungsi kesamaan, rata-rata keluaran fungsi kesamaan dan plot *Switching Kernel* dari hasil data Video 3



Gambar 3.17. Hasil proses pelacakan Video 3 (dari atas ke bawah) menggunakan Fungsi Kernel: Seragam, Segitiga, Epanechnikov, Gaussian dan *Switching Kernel*. Hasil pelacakan pada frame ke-20 dan ke-40.

Data Video ke-4, ada gambar tempelan stiker di kaca kendaraan bagian belakang dianggap oklusi mirip dengan pelat. Hal ini kadang-kadang dianggap sebagai pelat nomor, sehingga mempersulit proses pelacakan. Dari empat video yang digunakan untuk pengujian, hanya video ini yang menghasilkan proses pelacakan lebih buruk dari Mean Shift standar. Hasil pengujian pelacakan pelat nomor kendaraan ditampilkan seperti pada Gambar 3.18 dan 3.19. Bentuk keluaran fungsi kesamaan, rata-rata keluaran fungsi kesamaan dan plot *Switching Kernel* dari hasil data Video 4, diperlihatkan pada Gambar 3.18. Hasil proses pelacakan Video 4 menggunakan Fungsi Kernel: Seragam, Segitiga, Epanechnikov, Gaussian dan *Switching Kernel* berdasarkan hasil pelacakan pada frame ke-15 dan ke-30, diperlihatkan pada Gambar 3.19.

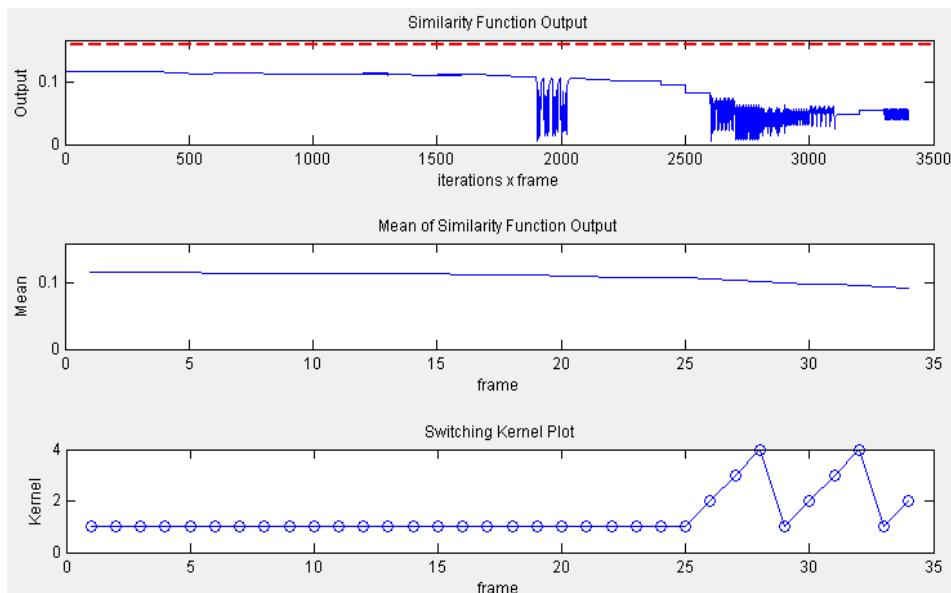


Gambar 3.18. Bentuk keluaran fungsi kesamaan, rata-rata keluaran fungsi kesamaan dan plot *Switching Kernel* dari hasil data Video 4

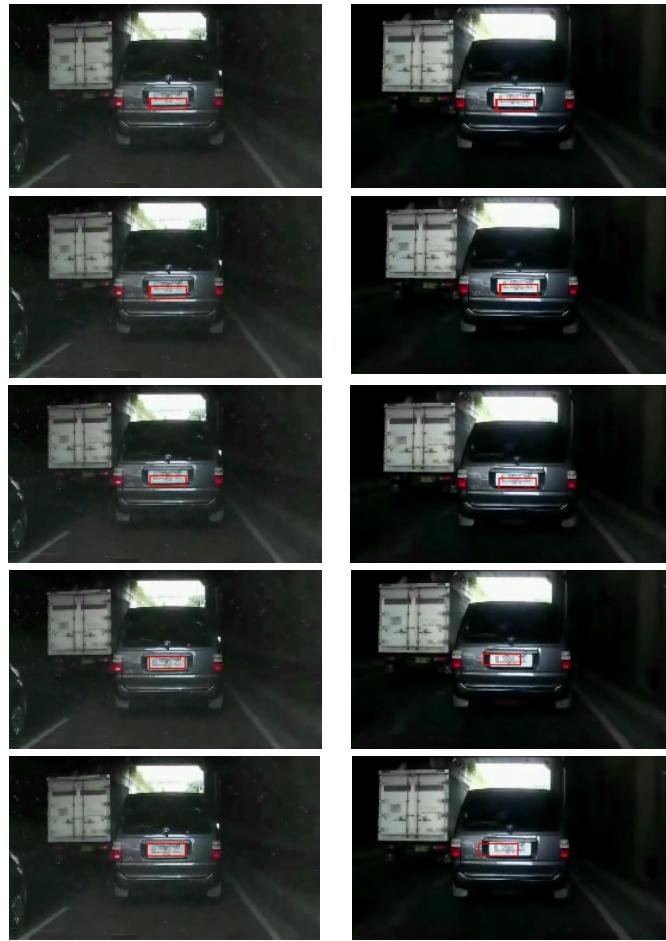
Data Video ke-5, perekaman video diambil pada saat kendaraan melaju di bawah jembatan. Dengan pencahayaan kurang dan cenderung gelap. Namun kondisi pelat nomor masih kelihatan sehingga masih dapat dibedakan. Dari hasil pengujian, metode BIM-Mean Shift dengan *Switching Kernel* menghasilkan performansi yang sama baiknya dengan Mean Shift standar dengan fungsi kernel Gaussian. Hasil pengujian pelacakan pelat nomor kendaraan ditampilkan seperti pada Gambar 3.20 dan 3.21. Bentuk keluaran fungsi kesamaan, rata-rata keluaran fungsi kesamaan dan plot *Switching Kernel* dari hasil data Video 5, diperlihatkan pada Gambar 3.20. Hasil proses pelacakan Video 5 menggunakan Fungsi Kernel: Seragam, Segitiga, Epanechnikov, Gaussian dan *Switching Kernel* berdasarkan hasil pelacakan pada frame ke-15 dan ke-30, diperlihatkan pada Gambar 3.21.



Gambar 3.19. Hasil proses pelacakan Video 4 (dari atas ke bawah) menggunakan Fungsi Kernel: Seragam, Segitiga, Epanechnikov, Gaussian dan *Switching Kernel*. Hasil pelacakan pada frame ke-15 dan ke-30.



Gambar 3.20. Bentuk keluaran fungsi kesamaan, rata-rata keluaran fungsi kesamaan dan plot *Switching Kernel* dari hasil data Video 5



Gambar 3.21. Hasil proses pelacakan Video 5 (dari atas ke bawah) menggunakan Fungsi Kernel: Seragam, Segitiga, Epanechnikov, Gaussian dan *Switching Kernel*. Hasil pelacakan pada frame ke-15 dan ke-30.

Selanjutnya, untuk penentuan akurasi pelacakan, secara empiris didefinisikan *Percentage Accuracy of Object Tracking* (PAOT) seperti pada Persamaan (3.37). Dihitung berdasarkan probabilitas, yaitu jumlah objek yang berhasil dilacak tiap frame, dibagi keseluruhan jumlah frame, dikalikan 100%.

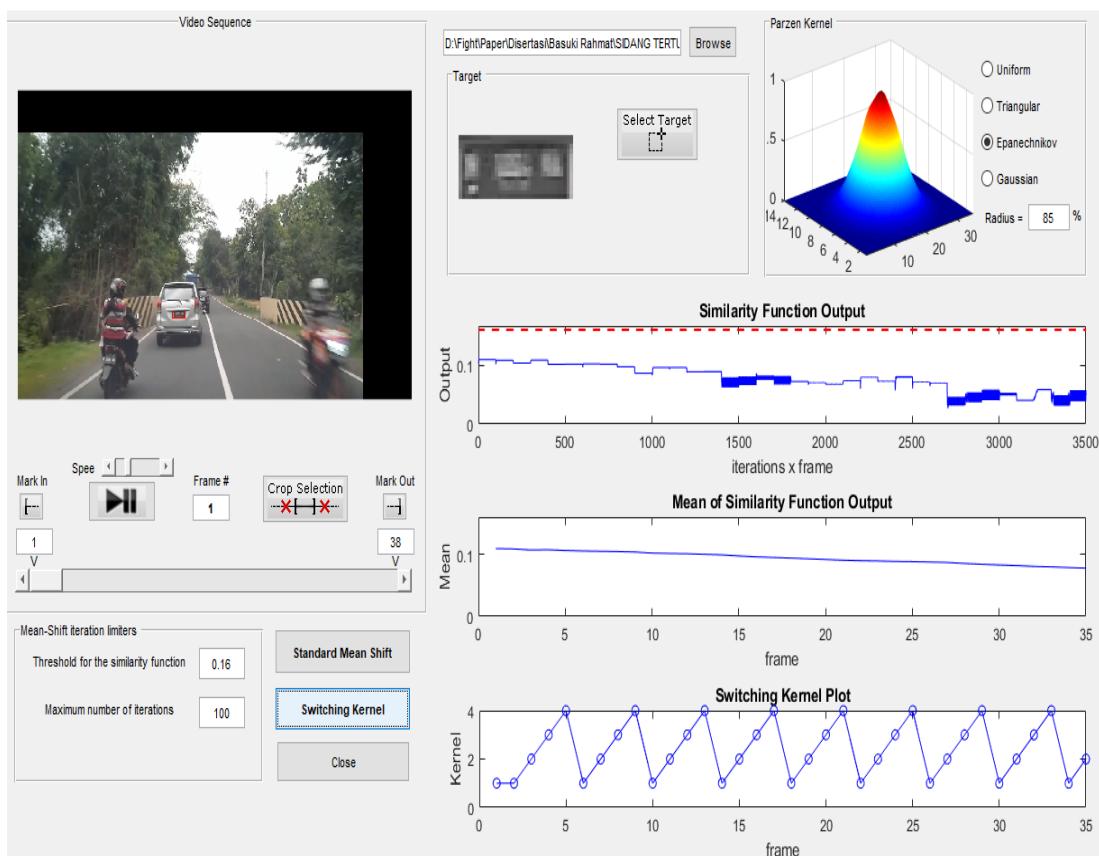
$$PAOT = \frac{\sum_{i=1}^Z O_i}{Z} \times 100 \% \quad (3.37)$$

Keterangan:

O_i : Objek yang berhasil dilacak pada *frame* ke-*i*.

Z : jumlah *frame*.

Contoh perhitungan persentase akurasi pelacakan dengan PAOT, dapat dilihat pada penjelasan Gambar 3.22 sampai dengan Gambar 3.24. Contoh hasil simulasi pelacakan diperlihatkan pada Gambar 3.22. Untuk menghitung jumlah objek yang berhasil dilacak, dilakukan secara manual, dengan cara pengamatan tiap frame. Contoh objek yang berhasil dilacak pada frame ke-18 diperlihatkan pada Gambar 3.23. Contoh objek yang gagal dilacak pada frame ke-30, diperlihatkan pada Gambar 3.24. Jumlah total frame adalah 38. Selanjutnya untuk mendapatkan persentase akurasi pelacakan, tinggal dihitung total objek yang berhasil dilacak tiap frame dibagi dengan jumlah total frame dikalikan dengan 100%. Misal objek yang berhasil dilacak tiap frame sebanyak 32 kali. Maka persentase akurasi pelacakannya adalah $32/38 \times 100\%$, yaitu 84.21%. Hasil pengujian simulasi pelacakan pelat nomor kendaraan secara lengkap dirangkum pada Tabel 3.3.



Gambar 3.22. Contoh hasil simulasi pelacakan



Gambar 3.23. Contoh objek yang berhasil dilacak pada frame ke-18



Gambar 3.24. Contoh objek yang gagal dilacak pada frame ke-30

Tabel 3.3. Akurasi pelacakan BIM-Mean Shift dengan *Switching Kernel* (%)

Video	Mean Shift Standar				BIM-Mean Shift dengan <i>Switching Kernel</i>
	a	b	c	d	
1	42.11	36.84	34.21	76.32	84.21
2	27.78	27.78	27.78	55.56	69.44
3	53.49	53.49	51.16	88.37	88.37
4	6.45	6.45	6.45	45.16	38.71
5	57.15	54.29	60.00	77.14	77.14
Rata-rata	37.40	35.77	35.92	68.51	71.57

Fungi Kernel:

a : Seragam

b : Segitiga

c : Epanechnikov

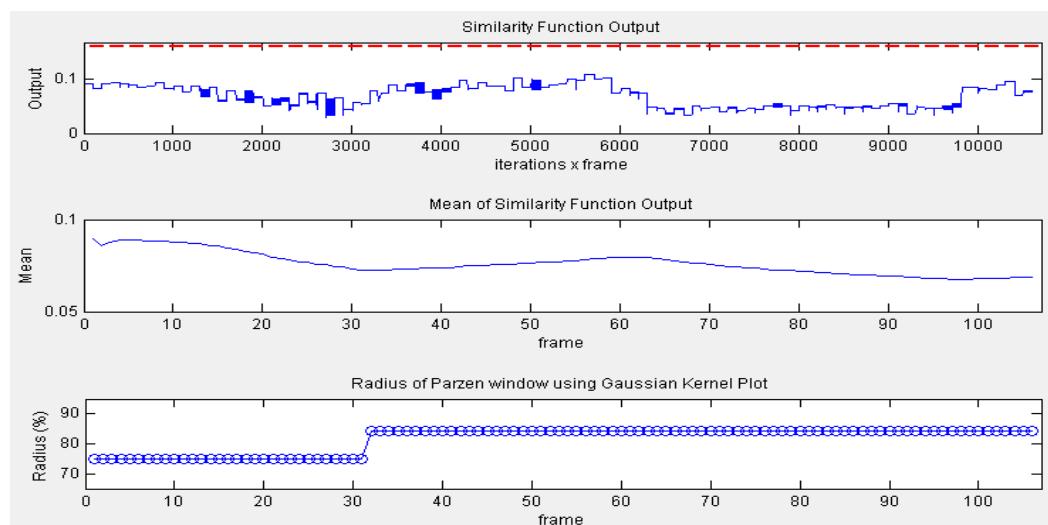
d : Gaussian

Dari hasil pengujian, Mean Shift standar dengan fungsi kernel Gaussian memiliki hasil rata-rata akurasi pelacakan yang lebih baik diantara Mean Shift standar lain. Dengan menghasilkan rata-rata akurasi pelacakan 68.51%, dibandingkan Mean Shift Standar dengan fungsi kernel Seragam, Segitiga, dan Epanechnikov masing-masing hanya 37.40%, 35.77% dan 35.92%. Dan metode yang diusulkan, yaitu BIM-Mean Shift dengan *Switching Kernel* menghasilkan rata-rata akurasi pelacakan yang lebih baik dibandingkan dengan Mean Shift standar termasuk Mean Shift standar yang menggunakan fungsi kernel Gaussian. Dengan menghasilkan rata-rata akurasi pelacakan 71.57%, dibandingkan Mean Shift Standar dengan fungsi kernel Gaussian yang hanya 68.51%.

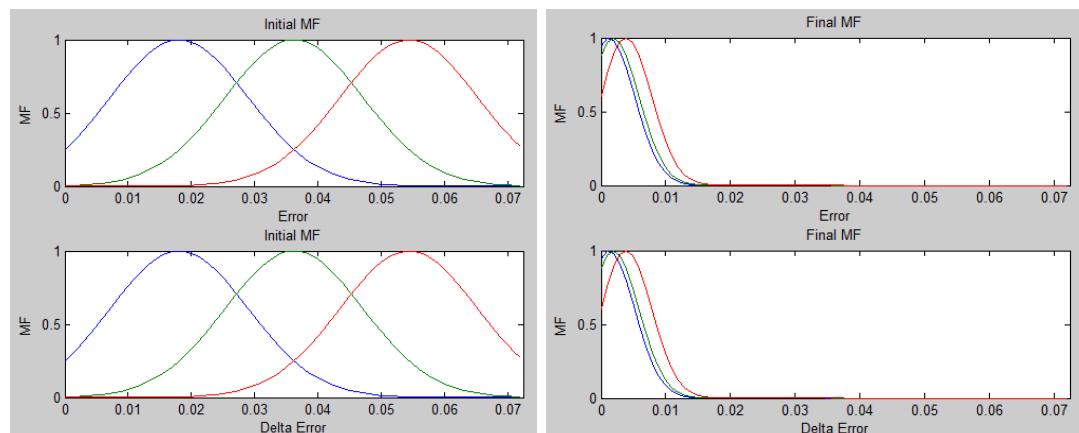
3.6.2. Pengujian BIM-Mean Shift dengan *Adaptive Fuzzy Gaussian Kernel*

Algoritma BIM-Mean Shift dengan *Adaptive Fuzzy Gaussian Kernel* diuji terhadap 4 rekaman video dari kendaraan di jalan raya. Algoritma ini diterapkan dalam rangka untuk melacak pelat nomor kendaraan Indonesia. Parameter yang digunakan untuk pengujian sama seperti pada BIM-Mean Shift dengan *Switching Kernel* yaitu menggunakan Tabel 3.2. Demikian juga untuk penentuan akurasi pelacakan, sama seperti pada BIM-Mean Shift dengan *Switching Kernel*, yaitu dengan menggunakan *Percentage Accuracy of Object Tracking* (PAOT) seperti pada Persamaan (3.37).

Gambar-gambar berikut ini memberikan gambaran hasil simulasasi. Data Video ke-1, hasil simulasinya seperti diperlihatkan pada Gambar 3.25 sampai dengan Gambar 3.27. Mean Shift standar menghasilkan persentase akurasi pelacakan objek sebesar 50.4673%, sedangkan BIM-Mean Shift dengan *Adaptive Fuzzy Gaussian Kernel* menghasilkan akurasi lebih baik sebesar 61.6822%. Dari sampel frame ke: 25, 50, 75, dan 100 seperti yang ditunjukkan pada Gambar 3.27, terlihat pada frame ke-25 dan frame ke-100 hasil pengujian menggunakan Mean Shift standar terjadi kegagalan proses pelacakan. Sementara BIM-Mean Shift dengan *Adaptive Fuzzy Gaussian Kernel*, hanya pada frame ke-100 terjadi kegagalan proses pelacakan. Ini menunjukkan bahwa BIM-Mean Shift dengan *Adaptive Fuzzy Gaussian Kernel* menghasilkan kinerja pelacakan objek yang lebih baik daripada menggunakan Mean Shift standar.



Gambar 3.25. Bentuk keluaran fungsi kesamaan, rata-rata keluaran fungsi kesamaan dan plot lebar pita atau radius kernel dari hasil data Video 1



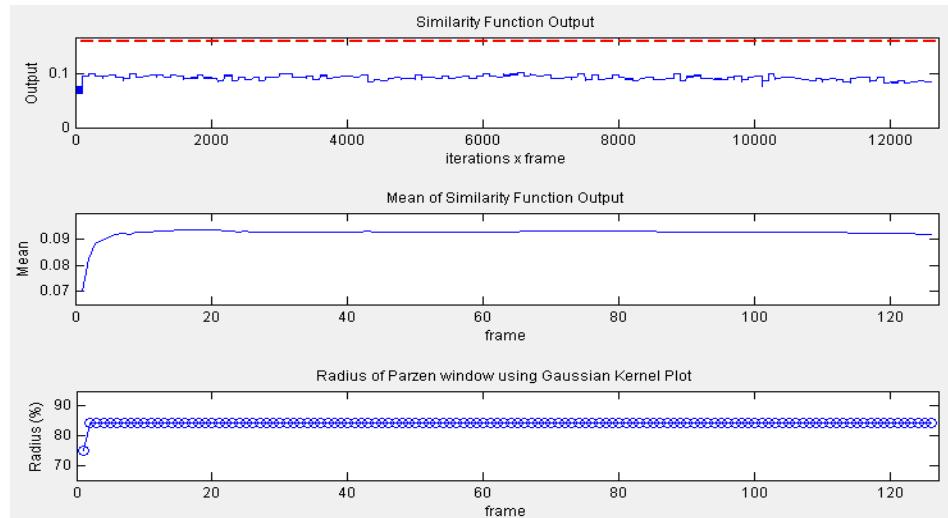
Gambar 3.26. Fungsi Keanggotaan Fuzzy awal dan akhir hasil data Video 1



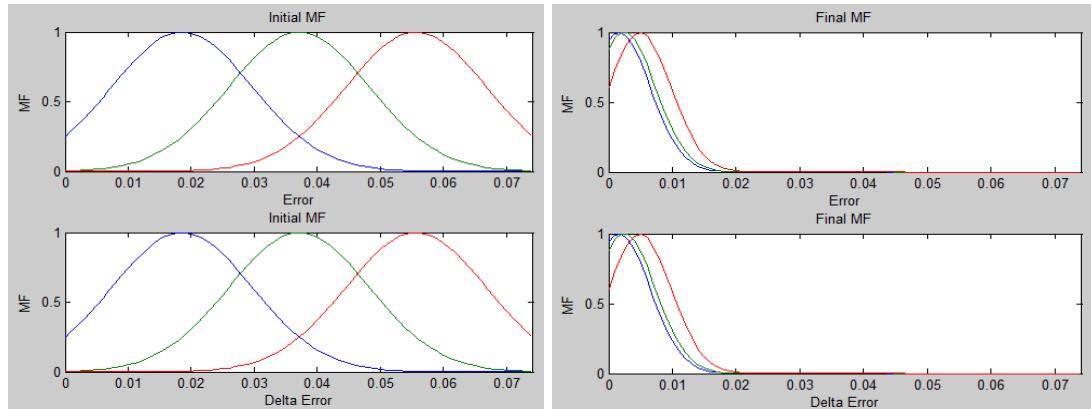
Gambar 3.27. Hasil proses pelacakan Video 1 sebelah kiri menggunakan *Static Gaussian Kernel* dan sebelah kanan menggunakan *Adaptive Fuzzy Gaussian Kernel*.

Hasil pelacakan (dari atas ke bawah) pada frame ke-25, 50, 75 dan ke-100.

Dari hasil pengujian menggunakan data Video ke-2, diperoleh hasil-hasil pengujian seperti diperlihatkan pada Gambar 3.28 sampai dengan Gambar 3.30. Mean Shift standar menghasilkan persentase akurasi pelacakan objek sebesar 76.3780%, sedangkan BIM-Mean Shift dengan *Adaptive Fuzzy Gaussian Kernel* menghasilkan akurasi lebih baik sebesar 80.3150%. Dari sampel frame seperti yang ditunjukkan pada Gambar 3.30, pada frame ke-90 hasil pengujian menggunakan Mean Shift standar terlihat keluar dari pelacakan dengan sangat jelas. Sementara BIM-Mean Shift dengan *Adaptive Fuzzy Gaussian Kernel* pada frame ke-120 terlihat tidak bisa melacak dengan sempurna, namun masih belum terlalu jauh dari objeknya. Ini menunjukkan bahwa BIM-Mean Shift dengan *Adaptive Fuzzy Gaussian Kernel* menghasilkan kinerja pelacakan objek yang lebih baik daripada menggunakan Mean Shift standar.

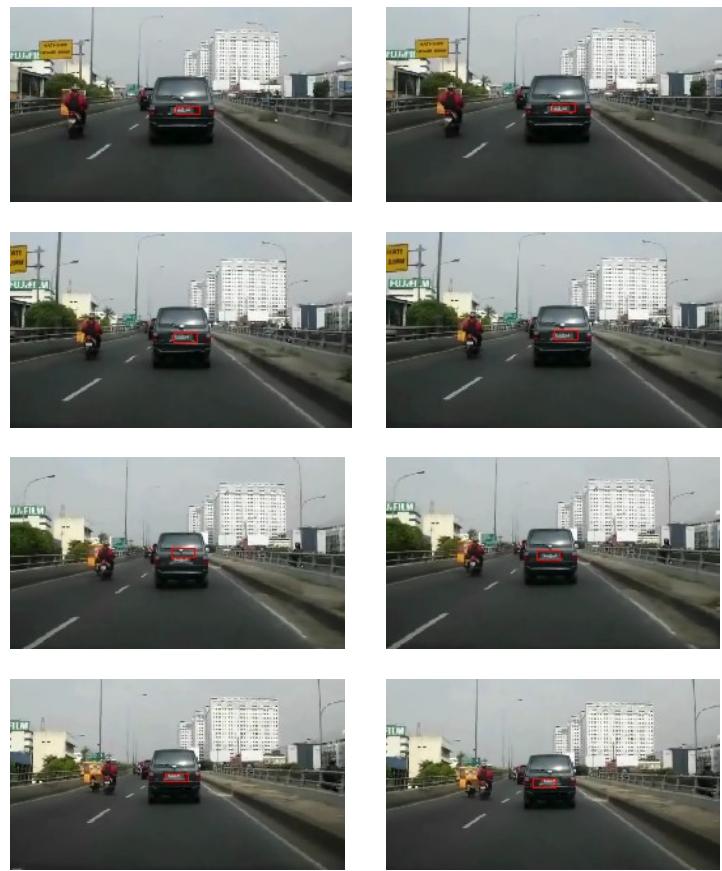


Gambar 3.28. Bentuk keluaran fungsi kesamaan, rata-rata keluaran fungsi kesamaan dan plot lebar pita atau radius kernel dari hasil data Video 2

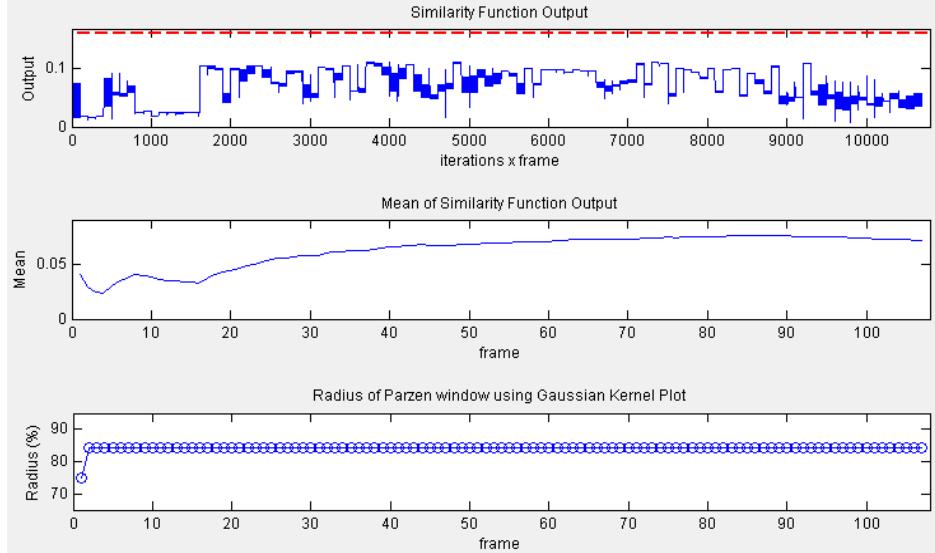


Gambar 3.29. Fungsi Keanggotaan Fuzzy awal dan akhir hasil data Video 2

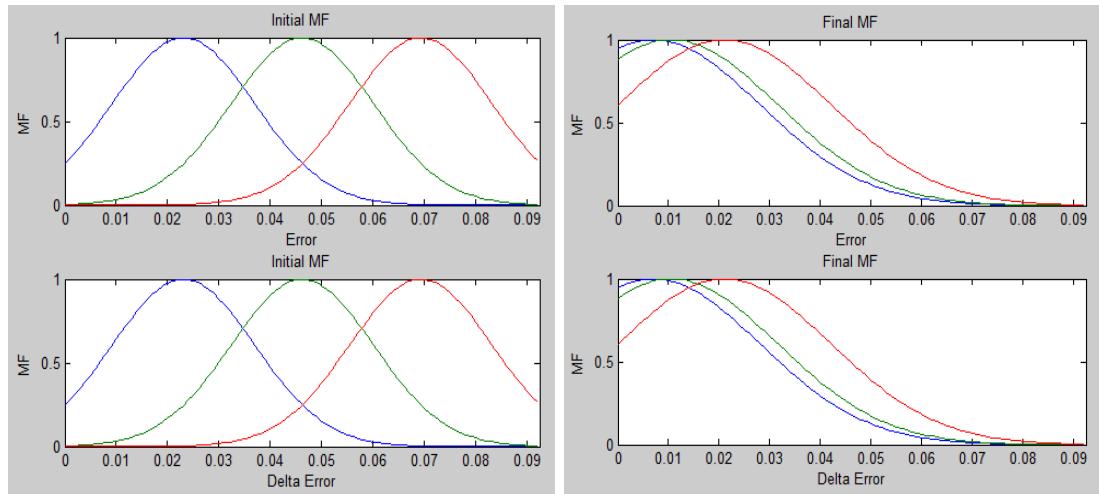
Dari hasil pengujian menggunakan data Video ke-3, diperoleh hasil-hasil pengujian seperti diperlihatkan pada Gambar 3.31 sampai dengan Gambar 3.33. Mean Shift standar menghasilkan persentase akurasi pelacakan objek sebesar 62.9630%, sedangkan BIM-Mean Shift dengan *Adaptive Fuzzy Gaussian Kernel* menghasilkan akurasi lebih baik sebesar 64.8148%. Dari hasil tersebut, tidak terlalu banyak perbedaan. Dari sampel frame ke: 27, 54, 81, dan 108, seperti yang ditunjukkan pada Gambar 3.33, hasil proses pelacakan masing-masing hampir sama. Hanya pada frame ke-108, dari hasil pengujian menggunakan Mean Shift standar, terlihat mulai keluar dari pelacakan. Ini menunjukkan bahwa BIM-Mean Shift dengan *Adaptive Fuzzy Gaussian Kernel* menghasilkan kinerja pelacakan objek yang lebih baik daripada menggunakan Mean Shift standar.



Gambar 3.30. Hasil proses pelacakan Video 2 sebelah kiri menggunakan *Static Gaussian Kernel* dan sebelah kanan menggunakan *Adaptive Fuzzy Gaussian Kernel*.
Hasil pelacakan (dari atas ke bawah) pada frame ke-30, 60, 90 dan ke-120.



Gambar 3.31. Bentuk keluaran fungsi kesamaan, rata-rata keluaran fungsi kesamaan dan plot lebar pita atau radius kernel dari hasil data Video 3



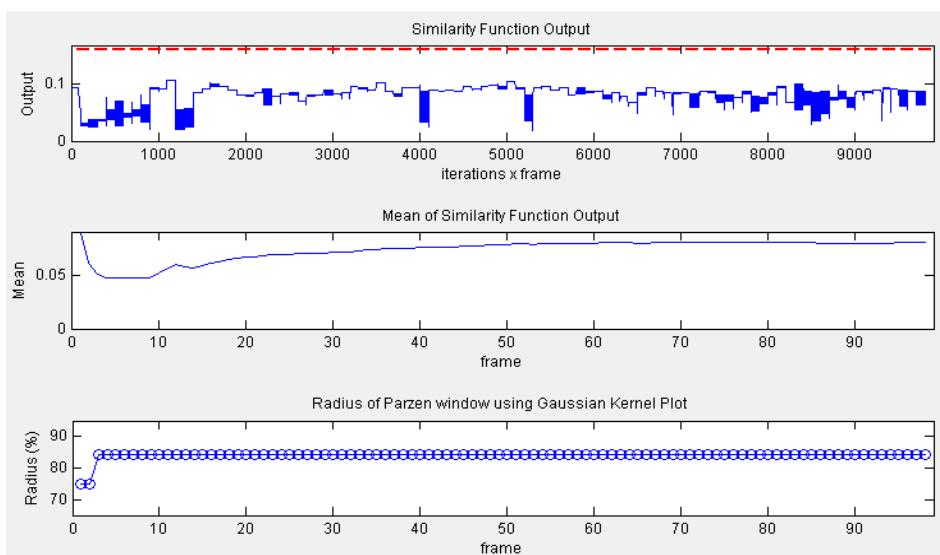
Gambar 3.32. Fungsi Keanggotaan Fuzzy awal dan akhir hasil data Video 3



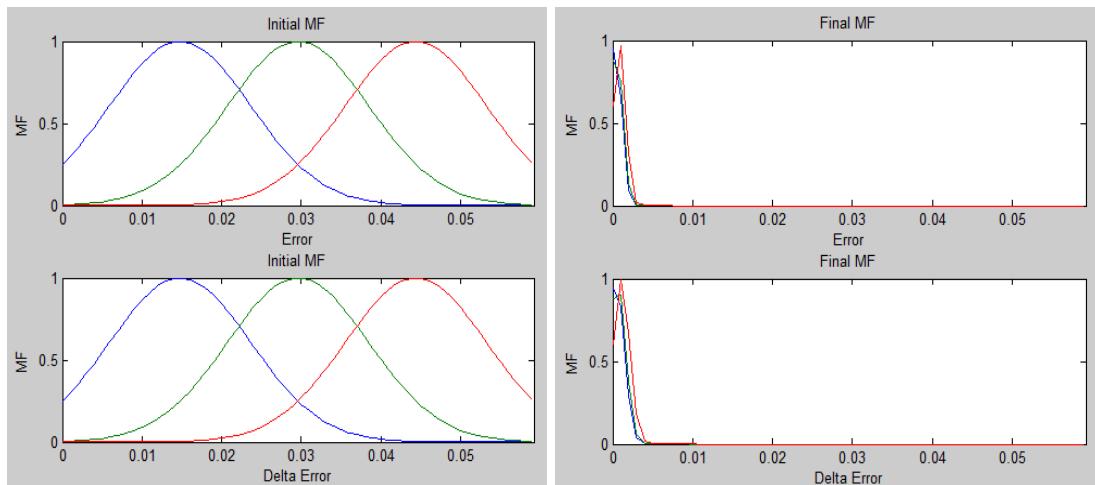
**Gambar 3.33. Hasil proses pelacakan Video 3 sebelah kiri menggunakan *Static Gaussian Kernel* dan sebelah kanan menggunakan *Adaptive Fuzzy Gaussian Kernel*.
Hasil pelacakan (dari atas ke bawah) pada frame ke-25, 50, 75 dan ke-100.**

Dari hasil pengujian menggunakan data Video ke-4, diperoleh hasil-hasil pengujian seperti diperlihatkan pada Gambar 3.34 sampai dengan Gambar 3.36. Mean

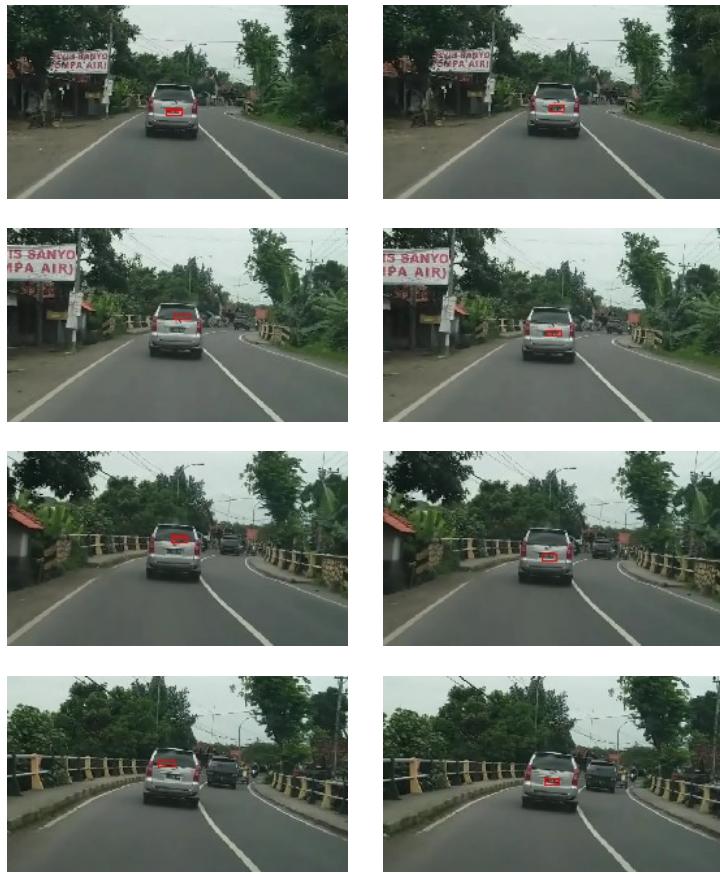
Shift standar menghasilkan persentase akurasi pelacakan objek sebesar 28.2828%, sedangkan BIM-Mean Shift dengan *Adaptive Fuzzy Gaussian Kernel* menghasilkan akurasi lebih baik sebesar 57.5758%. Dari sampel frame ke: 24, 48, 72, dan 96, seperti yang ditunjukkan pada Gambar 3.36, pada Mean Shift standar, terlihat semua kecuali pada frame ke-24, terjadi kegagalan proses pelacakan objek. Sementara BIM-Mean Shift dengan *Adaptive Fuzzy Gaussian Kernel*, terlihat pada semua frame, menghasilkan proses pelacakan objek dengan benar. Ini menunjukkan bahwa BIM-Mean Shift dengan *Adaptive Fuzzy Gaussian Kernel* menghasilkan kinerja pelacakan objek yang lebih baik daripada menggunakan Mean Shift standar.



Gambar 3.34. Bentuk keluaran fungsi kesamaan, rata-rata keluaran fungsi kesamaan dan plot lebar pita atau radius kernel dari hasil data Video 4



Gambar 3.35. Fungsi Keanggotaan Fuzzy awal dan akhir hasil data Video 4



Gambar 3.36. Hasil proses pelacakan Video 4 sebelah kiri menggunakan *Static Gaussian Kernel* dan sebelah kanan menggunakan *Adaptive Fuzzy Gaussian Kernel*.

Hasil pelacakan (dari atas ke bawah) pada frame ke-25, 50, 75 dan ke-100.

Seperti telah disebutkan sebelumnya, penentuan akurasi pelacakan, digunakan *Percentage Accuracy of Object Tracking* (PAOT) seperti pada Persamaan (3.37). Hasil pengujinya dirangkum pada Tabel 3.4.

Tabel 3.4. Akurasi pelacakan BIM-Mean Shift dengan *Adaptive Fuzzy Gaussian Kernel (%)*

Video	Mean Shift dengan <i>Static Gaussian Kernel</i>	BIM-Mean Shift dengan <i>Adaptive Fuzzy Gaussian Kernel</i>
1	50.4673	61.6822
2	76.3780	80.3150
3	62.9630	64.8148
4	28.2828	57.5758
Rata-rata	54.5228	66.0970

Dari hasil pengujian, BIM-Mean Shift dengan *Adaptive Fuzzy Gaussian Kernel* menghasilkan kinerja yang lebih baik. Dari 4 percobaan, secara “*head to head*” keseluruhan (100%) diperoleh hasil yang lebih baik daripada Mean Shift standar. Sedangkan jika dihitung berdasarkan rata-rata persentase akurasi pelacakan diperoleh 66.0970 % lebih baik daripada Mean Shift standar dengan *Static Gaussian Kernel* yang hanya 54.5228%.

3.7. Ringkasan Hasil

Dari hasil pengujian, BIM-Mean Shift baik dengan *Switching Kernel* maupun *Adaptive Fuzzy Gaussian Kernel*, masing-masing menghasilkan kinerja yang lebih baik daripada Mean Shift standar. Pelacakan pelat nomor kendaraan dengan BIM-Mean Shift dengan *Switching Kernel* memberikan rata-rata persentase akurasi pelacakan 71.57 % lebih baik daripada Mean Shift standar Gaussian yang hanya 68.51%. Sedangkan BIM-Mean Shift dengan *Adaptive Fuzzy Gaussian Kernel* memberikan rata-rata persentase akurasi pelacakan sebesar 66.0970%, lebih baik daripada Mean Shift standar dengan *Static Gaussian Kernel* yang hanya 54.5228%, namun dengan keseluruhan ujicoba 4 video (100%) secara “*head to head*” lebih unggul daripada Mean Shift standar.

Keterangan:

Publikasi yang dihasilkan dari topik ini, yaitu:

Rahmat, B., Joelianto, E., Purnama, I. K. E. and Purnomo, M. H. (2018) ‘An Improved Mean Shift Using Adaptive Fuzzy Gaussian Kernel for Indonesia Vehicle License Plate Tracking’, *IAENG International Journal of Computer Science*. International Association of Engineers.

Rahmat, B., Joelianto, E., Purnama, I. K. E. and Purnomo, M. H. (2018) ‘An Improved Mean Shift Performance Using Switching Kernels for Indonesia Vehicle License Plate Tracking Video’, *International Journal of Imaging and Robotics*. CESER Publications.

BAB 4

EKSTRAKSI KARAKTER PELAT NOMOR KENDARAAN

4.1. Fitur-fitur Ekstraksi Karakter Pelat Nomor Kendaraan

4.1.1. Fitur-fitur Segmentasi Karakter Pelat Nomor Kendaraan

Pelat nomor kendaraan yang sudah diisolasi kemudian dilakukan proses segmentasi untuk diekstraksi karakternya, selanjutnya akan dikenali. Sebuah pelat hasil ekstraksi dari tahap sebelumnya mungkin memiliki beberapa masalah, seperti kemiripan dan kecerahan yang tidak seragam. Algoritma segmentasi harus mengatasi semua permasalahan ini dalam langkah praprosesnya. Dalam beberapa penelitian sebelumnya disebutkan di literatur (Du *et al.*, 2013), penggunaan transformasi bilinear digunakan untuk memetakan kemiringan pelat nomor kendaraan yang diekstrak ke sebuah persegi panjang lurus. Penggunaan metode kuadrat terkecil (*least square*) digunakan untuk mengatasi kemiripan horisontal dan vertikal pada gambar pelat nomor kendaraan.

Dalam (Du *et al.*, 2013), metode pencocokan garis didasarkan pada kuadrat terkecil (*least square*) dengan *offset* tegak lurus untuk mengoreksi kemiringan pelat dalam arah horisontal. Koreksi kemiringan dalam arah vertikal dengan meminimalkan varian dari koordinat titik-titik proyeksi dilakukan. Segmentasi karakter dilakukan setelah koreksi horisontal dan karakter poin diproyeksikan sepanjang arah vertikal setelah transformasi geser.

Pemilihan nilai ambang batas (*threshold*) yang tidak tepat untuk proses binerisasi dapat menyebabkan karakter gabung. Karakter gabung ini membuat proses segmentasi sangat sulit dilakukan. Pelat nomor kendaraan dengan bingkai sekitarnya juga menyulitkan proses segmentasi, karena setelah proses binerisasi beberapa karakter dapat bergabung dengan bingkai. Peningkatan kualitas gambar sebelum proses binerisasi dapat membantu dalam pemilihan nilai ambang batas yang sesuai (Du *et al.*, 2013). Teknik yang biasa digunakan untuk peningkatan kualitas citra pelat nomor kendaraan adalah dengan cara penghapusan noise (*noise removal*), ekualisasi histogram (*histogram equalization*), dan peningkatan kontras (*contrast enhancement*) (Du *et al.*, 2013). Di penelitian sebelumnya juga dilakukan analisis gradien pada keseluruhan gambar untuk pendekripsi pelat nomor kendaraan. Kemudian gambar

pelat nomor kendaraan yang terdeteksi ditingkatkan kualitasnya dengan transformasi tingkat keabuan (*grey level transformation*).

Metode untuk peningkatan hanya karakter yang dituju dan pengurangan derau (*noise*) juga sudah dilakukan (Du *et al.*, 2013). Ukuran karakter dianggap sekitar 20 % dari ukuran pelat. Pertama, tingkat skala keabuan adalah skala untuk 0-100, maka piksel-piksel yang terbesar 20% dikalikan dengan 255. Hanya karakter yang dituju yang ditingkatkan, sementara derau (*noise*) piksel berkurang. Karena binerisasi dengan satu nilai ambang batas (*threshold*) global tidak selalu dapat diterima, maka metode binerisasi lokal adaptif biasanya digunakan.

Berikut ini, beberapa metode segmentasi karakter pelat nomor kendaraan yang ada didasarkan pada fitur-fitur yang digunakan (Du *et al.*, 2013).

- a. Segmentasi karakter pelat nomor kendaraan berdasarkan hubungan antar piksel (*License Plate Character Segmentation Using Pixel Connectivity*).
- b. Segmentasi karakter pelat nomor kendaraan berdasarkan profil proyeksi (*License Plate Character Segmentation Using Projection Profiles*).
- c. Segmentasi karakter pelat nomor kendaraan berdasarkan pengetahuan karakter sebelumnya (*License Plate Character Segmentation Using Prior Knowledge of Characters*).
- d. Segmentasi karakter pelat nomor kendaraan berdasarkan kontur karakter (*License Plate Character Segmentation Using Character Contours*).
- e. Segmentasi karakter pelat nomor kendaraan berdasarkan kombinasi fitur (*License Plate Character Segmentation Using Combined Features*).

Dari hasil-hasil penelitian sebelumnya berikut ini ringkasan perbandingan dari beberapa metode segmentasi karakter pelat nomor kendaraan yang ada didasarkan pada fitur-fitur yang digunakan. Perbandingan disajikan dalam penilaian kelebihan dan kekurangan dari metode masing-masing, seperti terlihat pada Tabel 4.1 (Du *et al.*, 2013).

4.1.2. Fitur-fitur Pengenalan Karakter Pelat Nomor Kendaraan

Hasil ekstraksi karakter yang kemudian dikenali dan sebagai keluaran sistem pelacakan dan pengenalan pelat nomor kendaraan berbasis video adalah karakter pelat nomor kendaraan. Pengenalan karakter dalam sistem ini mungkin memiliki beberapa kesulitan. Karena faktor *zoom* kamera, karakter yang diekstraksi tidak memiliki ukuran

yang sama dan ketebalan yang sama (Du *et al.*, 2013). Mengubah ukuran karakter menjadi satu ukuran sebelum dikenali membantu mengatasi masalah ini. Karakter huruf yang tidak sama sepanjang waktu karena pelat nomor berbagai negara menggunakan jenis huruf yang berbeda. Hasil ekstraksi karakter ada kemungkinan memiliki beberapa derau (*noise*) atau mereka mungkin rusak. Hasil ekstraksi karakter juga dapat dimiringkan (Du *et al.*, 2013).

Tabel 4.1. Perbandingan metode segmentasi karakter pelat nomor kendaraan

Metode	Kelebihan	Kekurangan
Menggunakan koneksi piksel (<i>pixel connectivity</i>)	Sederhana dan lugas, tegar terhadap rotasi pelat nomor.	Gagal mengekstrak semua karakter saat ada karakter yang tergabung atau rusak.
Menggunakan profil-profil proyeksi (<i>projection profiles</i>)	Independen terhadap posisi karakter, bisa menghadapi beberapa rotasi.	Derau (<i>noise</i>) mempengaruhi nilai proyeksi, memerlukan pengetahuan sebelumnya tentang jumlah karakter pelat.
Menggunakan pengetahuan karakter sebelumnya (<i>prior knowledge of characters</i>)	Sederhana.	Terbatas dengan pengetahuan sebelumnya, setiap perubahan dapat menyebabkan kesalahan.
Menggunakan kontur karakter (<i>character contours</i>)	Bisa mendapatkan batas karakter yang pasti.	Lambat dan bisa menghasilkan kontur yang tidak lengkap atau terdistorsi.
Menggunakan fitur gabungan (<i>combined features</i>)	Lebih andal.	Komputasional kompleks.

Berikut ini, beberapa metode pengenalan karakter pelat nomor kendaraan yang ada didasarkan pada fitur-fitur yang digunakan (Du *et al.*, 2013).

- a. Pengenalan karakter berdasarkan data mentah (*Character Recognition Using Raw Data*).

- b. Pengenalan karakter berdasarkan ekstraksi fitur (*Character Recognition Using Extracted Features*).

Dari hasil-hasil penelitian sebelumnya berikut ini ringkasan perbandingan dari beberapa metode pengenalan karakter pelat nomor kendaraan yang ada didasarkan pada fitur-fitur yang digunakan. Perbandingan disajikan dalam penilaian kelebihan dan kekurangan dari metode masing-masing, seperti terlihat pada Tabel 4.2 (Du *et al.*, 2013).

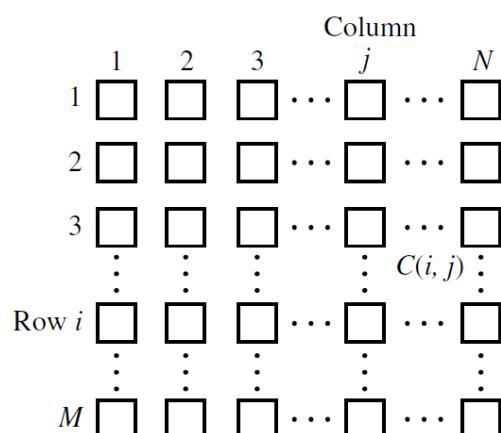
4.2. Segmentasi Karakter Pelat Nomor Kendaraan

4.2.1. *Cellular Neural Network (CNN)*

Salah satu metode yang bisa digunakan untuk keperluan segmentasi karakter pelat nomor kendaraan adalah *Cellular Neural Network* (CNN). CNN adalah sebuah paradigma komputasi paralel masif yang didefinisikan kedalam *array* biasa dimensi-n dari elemen-elemen, yang disebut sel (Chua and Yang, 1988). CNN dapat didefinisikan sebagai *array* 2D atau 3D yang secara lokal terhubung ke sistem dinamis nonlinear yang disebut sel, yang dinamikanya secara fungsional ditentukan oleh sekelompok kecil parameter yang mengontrol kekuatan interkoneksi sel (Chua and Yang, 1988). Beberapa notasi dasar dan definisi-definisi yang penting dari CNN diuraikan sebagai berikut.

Definisi 1: Arsitektur CNN standar.

Arsitektur standar CNN terdiri *array* persegi panjang berukuran MxN yang disebut sel ($C(i, j)$) dengan koordinat Kartesian (i, j), $i = 1, 2, \dots, M$, dan $j = 1, 2, \dots, N$ seperti terlihat pada Gambar 4.1 (Chua and Roska, 2004).



Gambar 4.1. Arsitektur CNN standar

Tabel 4.2. Perbandingan metode pengenalan karakter pelat nomor kendaraan

Metode	Kelebihan	Kekurangan
Mengguna-kan nilai piksel (<i>pixel values</i>)	Pencocokan template (<i>Template matching</i>)	Sederhana dan lugas.
	Beberapa template untuk setiap karakter	Mampu mengenali karakter miring. Waktu proses lebih lama.
Mengguna-kan ekstraksi fitur (<i>extracted features</i>)	Proyeksi horisontal dan vertikal	Mampu mengekstrak fitur penting, tegar terhadap distorsi, pengenalan cepat karena jumlah fitur lebih kecil dari piksel. Ekstraksi fitur membutuhkan waktu, fitur tidak tegar (<i>nonrobust</i>) akan menurunkan pengenalan-nya.
	<i>Hotelling transform</i>	
	Jumlah piksel hitam di setiap blok 3×3 piksel	
	Hitung jumlah elemen yang memiliki kecenderungan tingkat tertentu	
	Jumlah transisi dari karakter ke latar belakang dan jarak antar keduanya	
	Pencuplikan kontur karakter disekitarnya	
	Filter Gabor	
	Deteksi tepi Kirsch	
	Mengkonversi arah goresan karakter menjadi satu kode	
	Nilai piksel dari 11 sub blok	

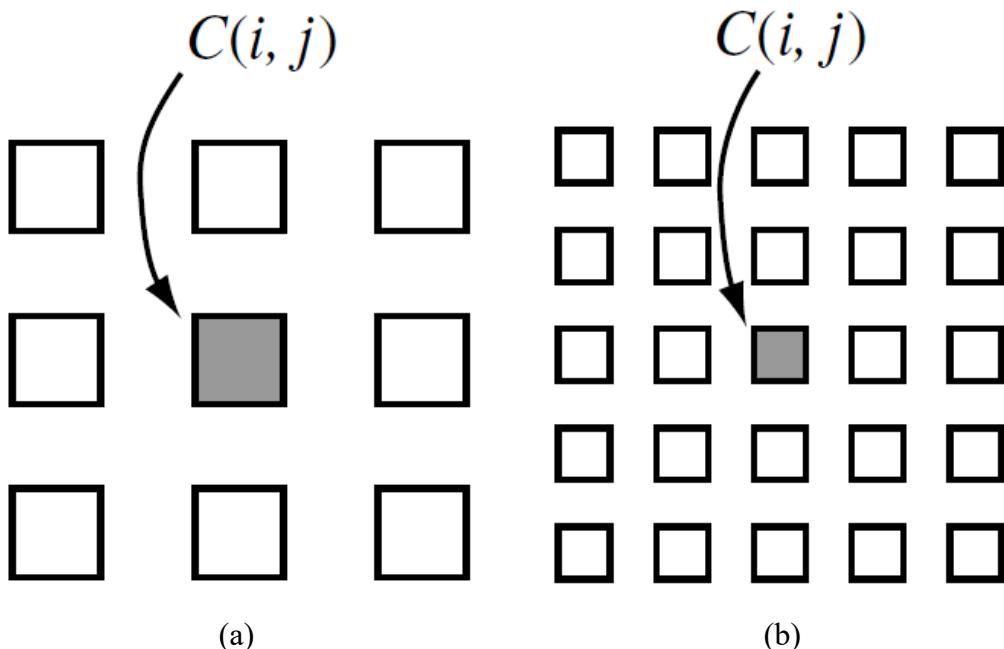
Definisi 2: Lingkup pengaruh sel $C(i, j)$.

Lingkup pengaruh, $S_r(i, j)$, sejauh radius r dari sel $C(i, j)$ didefinisikan sebagai himpunan dari semua sel di lingkungan yang memenuhi sifat berikut (Chua and Roska, 2004):

$$S_r(i, j) = \left\{ C(k, l) \mid \max_{1 \leq k \leq M, 1 \leq l \leq N} \{|k - i|, |l - j|\} \leq r \right\} \quad (4.1)$$

dengan radius, r integer positif.

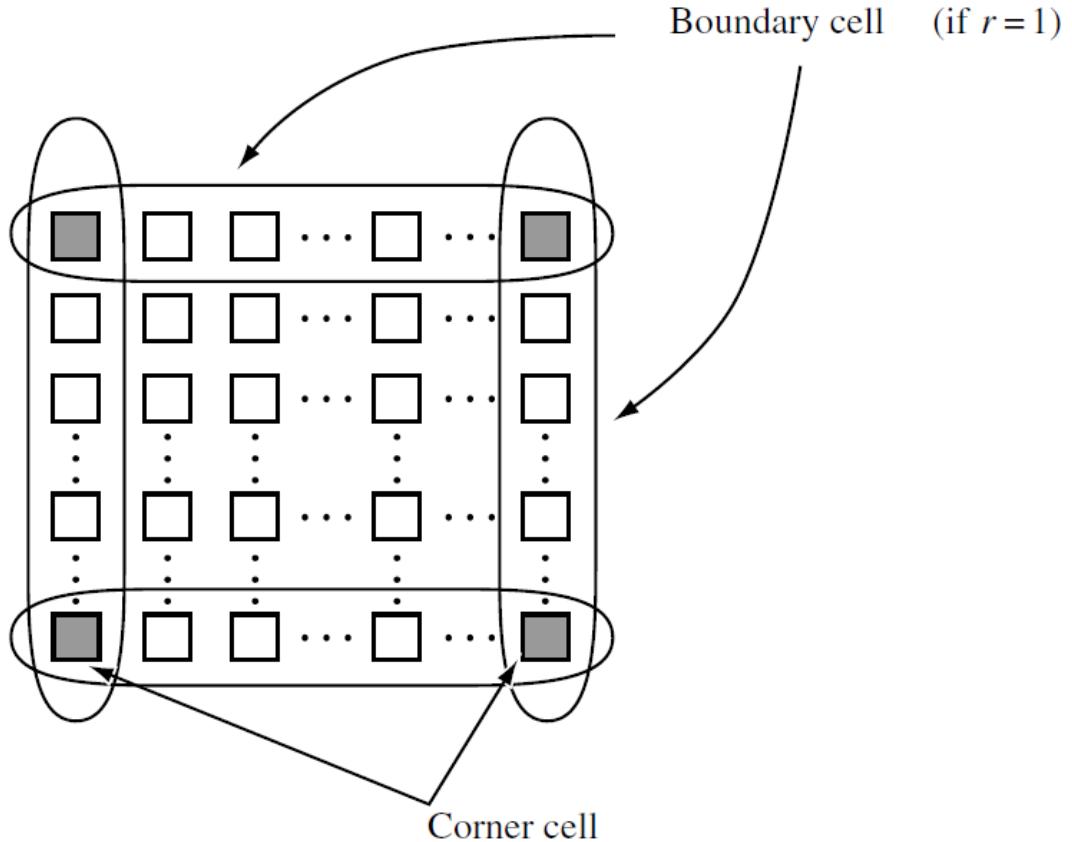
Kita kadang akan mengacu pada $S_r(i, j)$ sebagai lingkungan $(2r+1) \times (2r+1)$. Sebagai contoh pada Gambar 4.2(a) menunjukkan lingkungan $r = 1$ (3×3). Gambar 4.2(b) menunjukkan lingkungan $r = 2$ (5×5).



Gambar 4.2. (a) lingkungan $r = 1$ (3×3). (b) lingkungan $r = 2$ (5×5)

Definisi 3: Sel biasa dan sel perbatasan.

Sel $C(i, j)$ dinamakan sel biasa (*regular cell*) berkaitan dengan $S_r(i, j)$ jika dan hanya jika semua sel-sel lingkungan (tetangga) $C(k, l) \in S_r(i, j)$ ada. Dan sebaliknya $C(i, j)$ dinamakan sel perbatasan (*boundary cell*), seperti diperlihatkan pada Gambar 4.3. Sel-sel diluar perbatasan disebut sel tepi (*edge cell*). Tidak semua sel perbatasan disebut sel tepi, hanya sel yang memiliki nilai $r > 1$ saja.



Gambar 4.3. Sel perbatasan

Definisi 4: CNN standar.

Kelas CNN standar berukuran $1 M \times N$ didefinisikan oleh *array* persegi panjang dari sel $M \times N$, $C(i, j)$ yang berlokasi di tempat (i, j) , $i = 1, 2, \dots, M$, $j = 1, 2, \dots, N$. Masing-masing sel $C(i, j)$ didefinisikan secara matematik oleh persamaan-persamaan 4.2:

- 1). Persamaan keadaan (*State equation*) (Chua and Roska, 2004):

$$\dot{x}_{ij} = -x_{ij} + \sum_{C(k,l) \in S_r(i,j)} A(i, j; k, l) y_{kl} + \sum_{C(k,l) \in S_r(i,j)} B(i, j; k, l) u_{kl} + z_{ij} \quad (4.2)$$

Keterangan:

$x_{ij}, y_{kl}, u_{kl}, z_{ij}$: keadaan (*state*), keluaran, masukan, nilai ambang dari sel CNN.

\Re : ruang keadaan.

$A(i, j ; k, l)$: *template* umpanbalik.

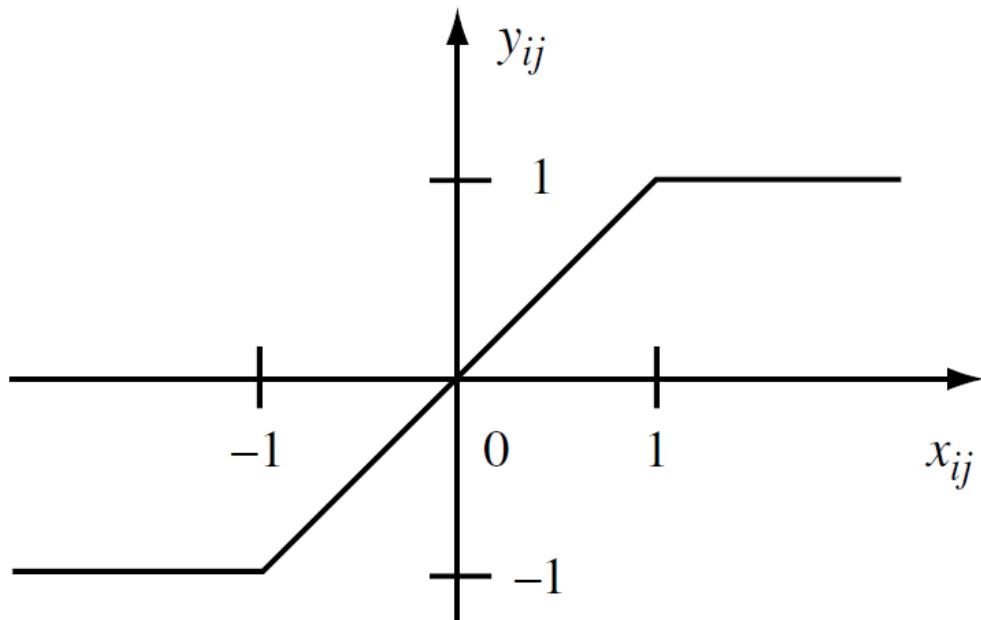
$B(i, j ; k, l)$: *template* kontrol.

untuk $x_{ij} \in \Re$, $y_{kl} \in \Re$, $u_{kl} \in \Re$, $z_{ij} \in \Re$ masing-masing dinamakan dengan keadaan (*state*), keluaran (*output*), masukan (*input*), dan nilai ambang (*threshold*) dari sel $C(i, j)$. Sedangkan matriks $A(i, j ; k, l)$ dan $B(i, j ; k, l)$ disebut dengan operator umpanbalik (*feedback operator*) dan operator sinaptik masukan (*input synaptic operator*). Kadang disebut *template* umpanbalik (*feedback template*) dan *template* kontrol (*control template*).

- 2). Persamaan keluaran (*Output equation*) (Chua and Roska, 2004):

$$y_{ij} = f(x_{ij}) = \frac{1}{2} |x_{ij} + 1| - \frac{1}{2} |x_{ij} - 1| \quad (4.3)$$

Ini dinamakan nonlinearitas standar, fungsinya seperti diperlihatkan pada Gambar 4.4.



Gambar 4.4. Nonlinearitas standar

- 3). Kondisi perbatasan (*Boundary conditions*)

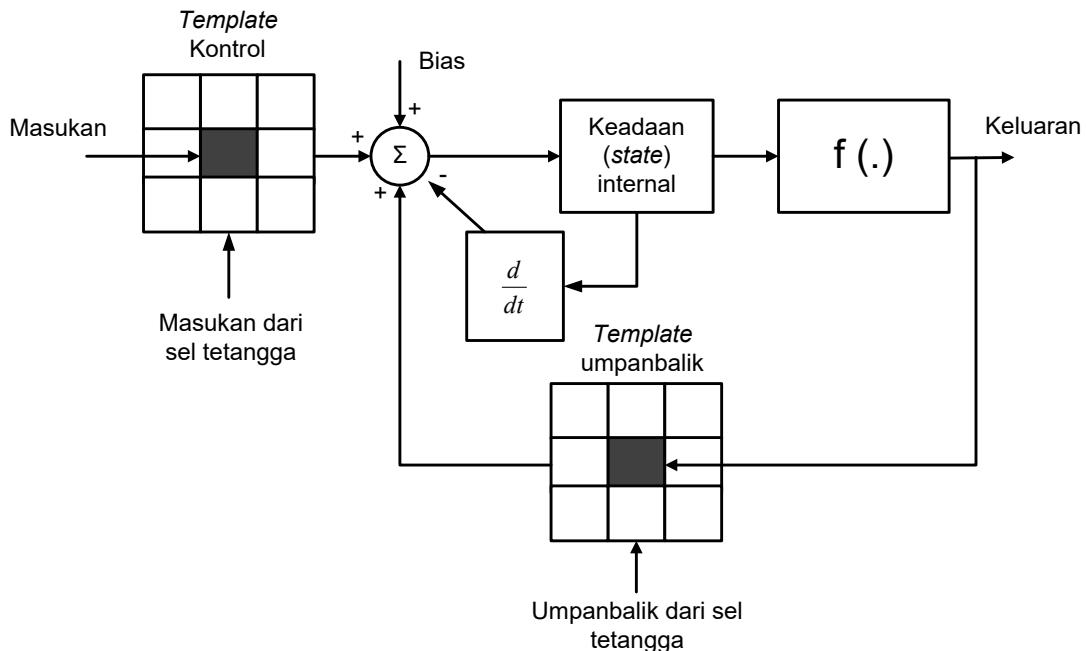
Kondisi perbatasan adalah penentuan y_{kl} dan u_{kl} untuk sel-sel milik Sr (i, j) dari sel tepi tetapi berada di luar array $M \times N$.

- 4). Keadaan awal (*Initial state*).

$$x_{ij}(0), \quad i=1, \dots, M, \quad j=1, \dots, N \quad (4.4)$$

4.2.2. Arsitektur CNN

Arsitektur CNN digambarkan seperti blok diagram CNN seperti diperlihatkan pada Gambar 4.5 (Selvathi, Selvaraj and Selvi, 2010).



Gambar 4.5. Blok diagram CNN

Dari blok diagram CNN dapat dilihat terdapat 2 (dua) matriks penting yaitu matriks $A(i, j ; k, l)$ dan $B(i, j ; k, l)$ yang disebut *template umpanbalik (feedback template)* dan *template kontrol (control template)*. Dalam banyak penelitian yang menggunakan metode CNN untuk menyelesaikan permasalahan, perhatian terhadap kedua matriks ini menjadi penting. Kadang nilai-nilai elemen-elemen dari kedua matriks tersebut dibiarkan bebas. Contohnya dapat dibaca di referensi berikut: (Fasih, Chamberlain Chedjou and Kyamakya, 2009), (Fasih, Chedjou and Kyamakya, 2008), (Khan *et al.*, 2009).

Misalnya untuk ukuran matriks 3x3, maka nilai elemen-elemen dari kedua matriks tersebut adalah:

$$A = \begin{bmatrix} a1 & a2 & a3 \\ a4 & a5 & a6 \\ a7 & a8 & a9 \end{bmatrix}, \quad B = \begin{bmatrix} b1 & b2 & b3 \\ b4 & b5 & b6 \\ b7 & b8 & b9 \end{bmatrix} \quad (4.5)$$

dengan nilai elemen-elemen $a1, \dots, a9$, dan $b1, \dots, b9$ nilainya bebas.

Sedangkan di penelitian-penelitian lain, nilai elemen-elemen dari matriks $A(i, j ; k, l)$ dan $B(i, j ; k, l)$ disusun dalam bentuk *cloning template*. Antara lain dapat dibaca

di referensi berikut (Basturk and Gunay, 2009), (Selvathi, Selvaraj and Selvi, 2010), (Crounse and Chua, 1995), (Parmaksizoglu, Gunay and Alci, 2011). Penjabarannya sebagai berikut, misalnya untuk matriks berukuran 3x3.

Nilai masing-masing matriks sesuai persamaan berikut (Basturk and Gunay, 2009):

$$\begin{aligned} A(i, j; k, l) &= A(k, l; i, j) \\ B(i, j; k, l) &= B(k, l; i, j) \end{aligned}, \quad |x_{ij}(0)| \leq 1, \quad |u_{ij}| \leq 1 \quad (4.6)$$

Sehingga CNN stabil secara lengkap dapat dibentuk dengan cara pemilihan *cloning template* berikut (Basturk and Gunay, 2009):

$$\begin{aligned} a_1 &= a_9, \quad a_2 = a_8, \quad a_3 = a_7, \quad a_4 = a_6, \\ b_1 &= b_9, \quad b_2 = b_8, \quad b_3 = b_7, \quad b_4 = b_6. \end{aligned} \quad (4.7)$$

sehingga bentuk matriks *template* umpanbalik (*feedback template*) dan *template* kontrol (*control template*) adalah sebagai berikut (Basturk and Gunay, 2009):

$$A = \begin{bmatrix} a1 & a2 & a3 \\ a4 & a5 & a6 \\ a7 & a8 & a9 \end{bmatrix} = \begin{bmatrix} a1 & a2 & a3 \\ a4 & a5 & a4 \\ a3 & a2 & a1 \end{bmatrix} \quad (4.8)$$

$$B = \begin{bmatrix} b1 & b2 & b3 \\ b4 & b5 & b6 \\ b7 & b8 & b9 \end{bmatrix} = \begin{bmatrix} b1 & b2 & b3 \\ b4 & b5 & b4 \\ b3 & b2 & b1 \end{bmatrix} \quad (4.9)$$

4.2.3. Hybrid CNN

Hybrid CNN atau *HCNN* adalah metode penggabungan metode CNN dengan metode algoritma cerdas lainnya. Peluang penggabungan CNN dengan algoritma cerdas lain adalah bagaimana melakukan proses penalaan (*tuning*) terhadap kedua matriks CNN, yaitu matriks *template* umpanbalik (*feedback template*) dan *template* kontrol (*control template*). Telah banyak penelitian sebelumnya yang menggabungkan metode CNN dengan metode algoritma cerdas lainnya untuk menyelesaikan berbagai persoalan, diantaranya dapat disebutkan sebagai berikut:

- Penggabungan CNN dengan Algoritma Genetika (*Genetic Algorithm*) (GA) (Khan *et al.*, 2009), (Fasih, Chedjou and Kyamakya, 2008), (Kozek, Roska and Chua, 1993), (Fasih, Chamberlain Chedjou and Kyamakya, 2009), (Parmaksizoglu, Gunay and Alci, 2011), (Balya and Gal, 2006), (Meng and Zhao, 2005), (Taraglio and Zanella, 1996), (Lopez, Vilarino and Cabello, 2000).

- Penggabungan CNN dengan Algoritma *Simulated Annealing* (SA) (Selvathi, Selvaraj and Selvi, 2010), (Feiden and Tetzlaff, 2001), (Bang, Sheu and Wu, 1996), (Bang, Sheu and Chou, 1996).
- Penggabungan CNN dengan Algoritma *Particle Swarm Optimization* (PSO) (Jianlai, Chunling and Chao, 2009), (Giaquinto and Fornarelli, 2009), (Su, Lin and Liu, 2008), (Su, Wang and Liu, 2007).
- Penggabungan CNN dengan Algoritma *Differential Evolution* (DE) (Basturk and Gunay, 2009).
- Penggabungan CNN dengan Algoritma *Adaptive Fuzzy System* (AFS) (Rahmat *et al.*, 2013).
- Penggabungan CNN dengan Algoritma *Adaptive Fuzzy* (AF) dan CNN dengan Neuro-Fuzzy struktur *Adaptive Neuro-Fuzzy Inference System* (ANFIS), yang diberi nama BIM-HCNN yang digunakan untuk proses segmentasi karakter pelat nomor kendaraan, diuraikan pada Subbab 4.2.4 (Rahmat *et al.*, 2016).

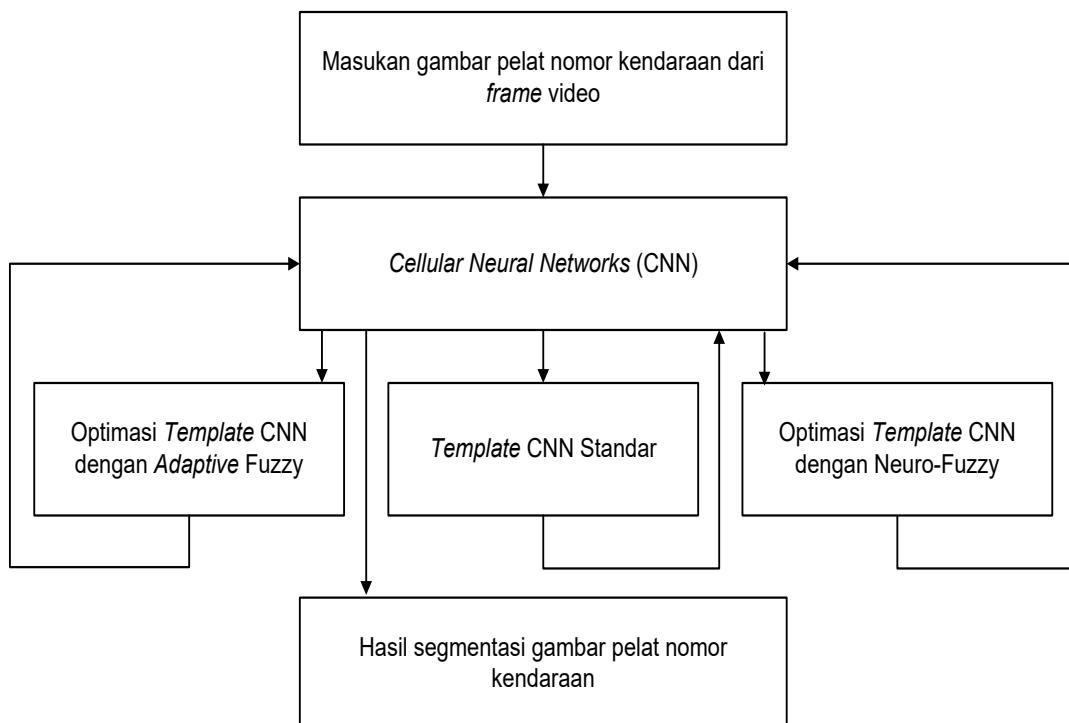
4.2.4. Segmentasi Karakter Pelat Nomor Kendaraan dengan BIM-HCNN

Segmentasi karakter pelat nomor kendaraan dengan BIM-HCNN merupakan penggabungan CNN standar dengan *Adaptive Fuzzy* dan Neuro-Fuzzy struktur ANFIS. Langkah-langkah komputasinya sebagai berikut (Rahmat *et al.*, 2016):

- (1) Gambar atau frame video yang berisi gambar pelat nomor kendaraan yang akan disegmentasi dan gambar pelat nomor ideal dijadikan masukan CNN.
- (2) Nilai *template* CNN dioptimasi dengan *template* CNN standar.
- (3) CNN dijalankan dengan gambar pelat masukan dari langkah 1 dan nilai-nilai *template* dari langkah 2.
- (4) Keluaran dari langkah 3 sebagai hasil segmentasi karakter pelat nomor kendaraan disimpan.
- (5) Nilai *template* dioptimasi dengan algoritma *Adaptive Fuzzy*.
- (6) CNN dijalankan dengan gambar masukan yang sama seperti pada langkah 1, tetapi dengan nilai-nilai template baru yang diperoleh dari langkah 5.
- (7) Keluaran dari langkah 6 sebagai hasil segmentasi karakter pelat nomor kendaraan disimpan.
- (8) Nilai *template* dioptimasi dengan algoritma Neuro-Fuzzy.

- (9) CNN dijalankan dengan gambar masukan yang sama seperti pada langkah 1, tetapi dengan nilai-nilai template baru yang diperoleh dari langkah 8.
- (10) Keluaran dari langkah 9 sebagai hasil segmentasi karakter pelat nomor kendaraan disimpan.
- (11) Evaluasi dilakukan antara hasil yang diperoleh dari langkah 4, 7 dan 10.

Blok diagram dari BIM-HCNN untuk segmentasi karakter pelat nomor kendaraan ini diperlihatkan pada Gambar 4.6 dan Gambar 4.7.

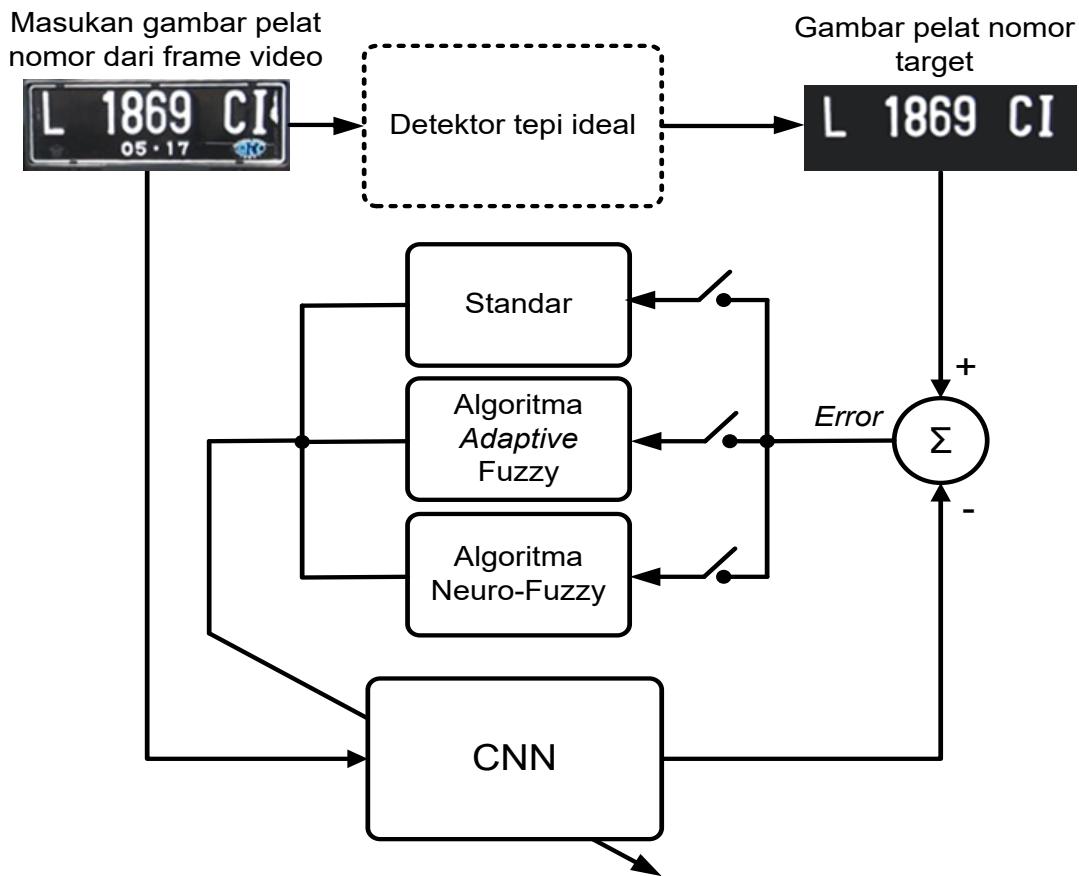


Gambar 4.6. Blok diagram HCNN

4.2.4.1. Optimasi *Template CNN* dengan *Adaptive Fuzzy*

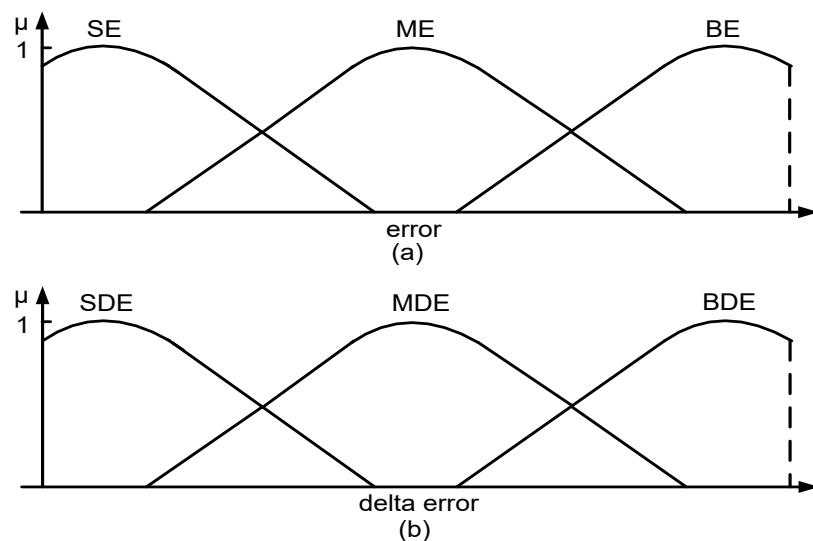
Algoritma *Adaptive Fuzzy* digunakan untuk melakukan proses penalaan *template CNN* agar diperoleh nilai parameter matriks A and B sebagai *template* umpanbalik dan *template* kontrol. Untuk didapatkan nilai *template CNN*, matriks A and B dalam bentuk standar dinyatakan sebagai berikut:

$$A = \begin{bmatrix} a1 & a2 & a3 \\ a4 & a5 & a6 \\ a7 & a8 & a9 \end{bmatrix}, \quad B = \begin{bmatrix} b1 & b2 & b3 \\ b4 & b5 & b6 \\ b7 & b8 & b9 \end{bmatrix} \quad (4.10)$$



Gambar 4.7. Blok diagram BIM-HCNN untuk segmentasi karakter pelat nomor kendaraan

Nilai *template* CNN dioptimasi dengan cara memanfaatkan nilai-nilai parameter fungsi keanggotaan fuzzy seperti diperlihatkan pada Gambar 4.8.



Gambar 4.8. (a) Fungsi Keanggotaan Error. (b) Fungsi Keanggotaan Delta Error

Dengan menggunakan istilah-istilah: *small error* (SE), *medium error* (ME), *big error* (BE), *small delta error* (SDE), *medium delta error* (MDE) dan *big delta error* (BDE), dan digunakan fungsi bell sebagai Fungsi Keanggotaan Fuzzy, maka bentuk fungsi bell dinyatakan sebagai berikut:

$$\begin{aligned}\mu_{SE} &= bell(error; a1, a2, a3) & \mu_{SDE} &= bell(delta\ error; b1, b2, b3) \\ \mu_{ME} &= bell(error; a4, a5, a6) & \mu_{MDE} &= bell(delta\ error; b4, b5, b6) \\ \mu_{BE} &= bell(error; a7, a8, a9) & \mu_{BDE} &= bell(delta\ error; b7, b8, b9)\end{aligned}\quad (4.11)$$

Persamaan fungsi keanggotaan bell dengan parameter-parameter fungsi keanggotaan a_i, b_i, c_i dan n_e sebagai masukan *error* atau *delta error* dinyatakan dalam persamaan berikut:

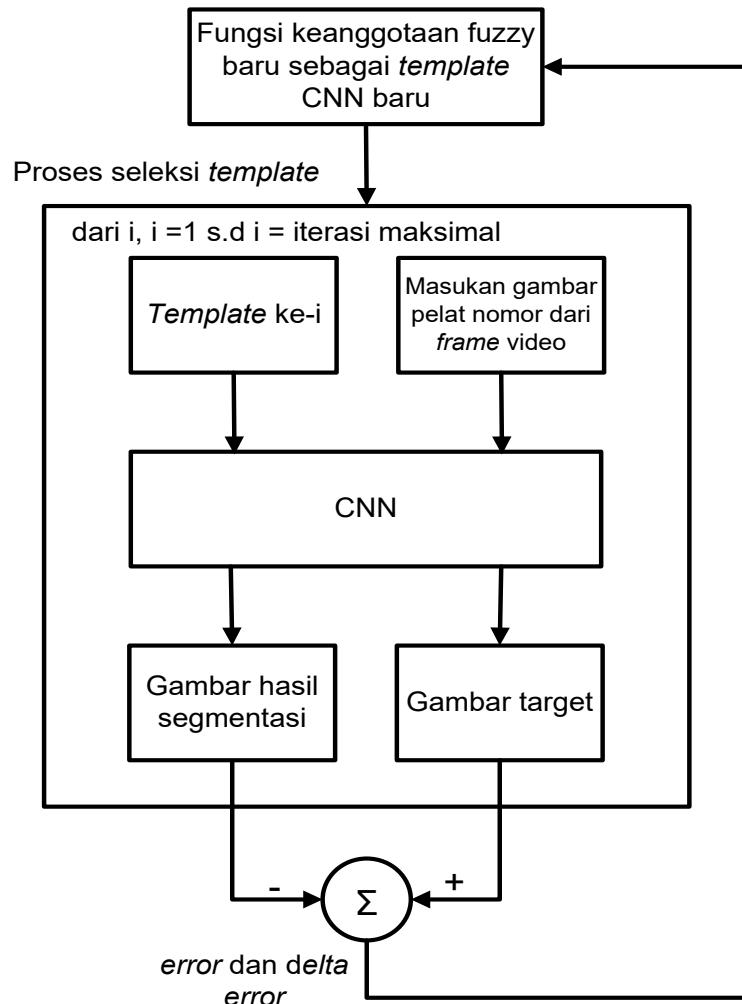
$$bell(n_e; a_i, b_i, c_i) = \frac{1}{1 + \left| \frac{(x - c_i)}{a_i} \right|^2 b_i} \quad (4.12)$$

Selanjutnya untuk mendapatkan variasi dari fungsi keanggotaan fuzzy yang berarti variasi dari *template CNN*, dilakukan proses secara iteratif seperti diperlihatkan pada Gambar 4.9.

Keluaran CNN dibandingkan dengan gambar target yang dianggap sebagai pelat nomor kendaraan ideal yang diinginkan. Dari perbandingan ini, diperoleh selisih (*error*). *Error* dan *delta error* kemudian digunakan sebagai fungsi keanggotaan *Adaptive Fuzzy* didefinisikan sebagai variasi *template CNN* yang baru. Selanjutnya, keluaran CNN dibandingkan dengan gambar target dan memberikan selisih (*error*) lagi. Proses ini diulang terus menerus sampai RMSE yang diinginkan tercapai atau iterasi dihentikan.

Variasi parameter fungsi keanggotaan *Adaptive Fuzzy* sebagai variasi *template CNN* yang baru dinyatakan dalam persamaan berikut:

$$\begin{aligned}A(i+1) &= \begin{bmatrix} |a1(i) - \overline{error}(i)| & |a2(i)| & |a3(i) - \overline{error}(i)| \\ |a4(i) - \overline{error}(i)| & |a5(i)| & |a6(i) - \overline{error}(i)| \\ |a7(i) - \overline{error}(i)| & |a8(i)| & |a9(i) - \overline{error}(i)| \end{bmatrix} \\ B(i+1) &= \begin{bmatrix} |b1(i) - \overline{delta\ error}(i)| & |b2(i)| & |b3(i) - \overline{delta\ error}(i)| \\ |b4(i) - \overline{delta\ error}(i)| & |b5(i)| & |b6(i) - \overline{delta\ error}(i)| \\ |b7(i) - \overline{delta\ error}(i)| & |b8(i)| & |b9(i) - \overline{delta\ error}(i)| \end{bmatrix}\end{aligned}\quad (4.13)$$



Gambar 4.9. Pembangkitan variasi template CNN menggunakan Adaptive Fuzzy

Keterangan, \overline{error} dan $\overline{\text{delta error}}$ sebagai rata-rata $error$ dan delta error , dan i sebagai tahap iterasi (iterasi ke- i).

Untuk mendapatkan parameter Fungsi Keanggotaan *Adaptive Fuzzy* terbaik yang berarti template CNN terbaik, hal itu dapat dilakukan dengan mengulangi percobaan beberapa kali. Setelah itu, parameter terbaik yang memenuhi kriteria yang dipilih. Atau, selama proses tersebut, perubahan parameter diproses menggunakan aturan konvergensi kesalahan yang didefinisikan sebagai *root mean square error* (RMSE). Konvergensi kesalahan didefinisikan oleh persamaan:

$$P_{i+1} = \begin{cases} P_{\text{baru}}, & \text{jika } RMSE(P_{i+1}) \leq RMSE(P_i), \\ P_{\text{lama}}, & \text{lainnya.} \end{cases} \quad (4.14)$$

P_i parameter Fungsi Keanggotaan Fuzzy Bell, yaitu a_i, b_i, c_i dan (i adalah iterasi).

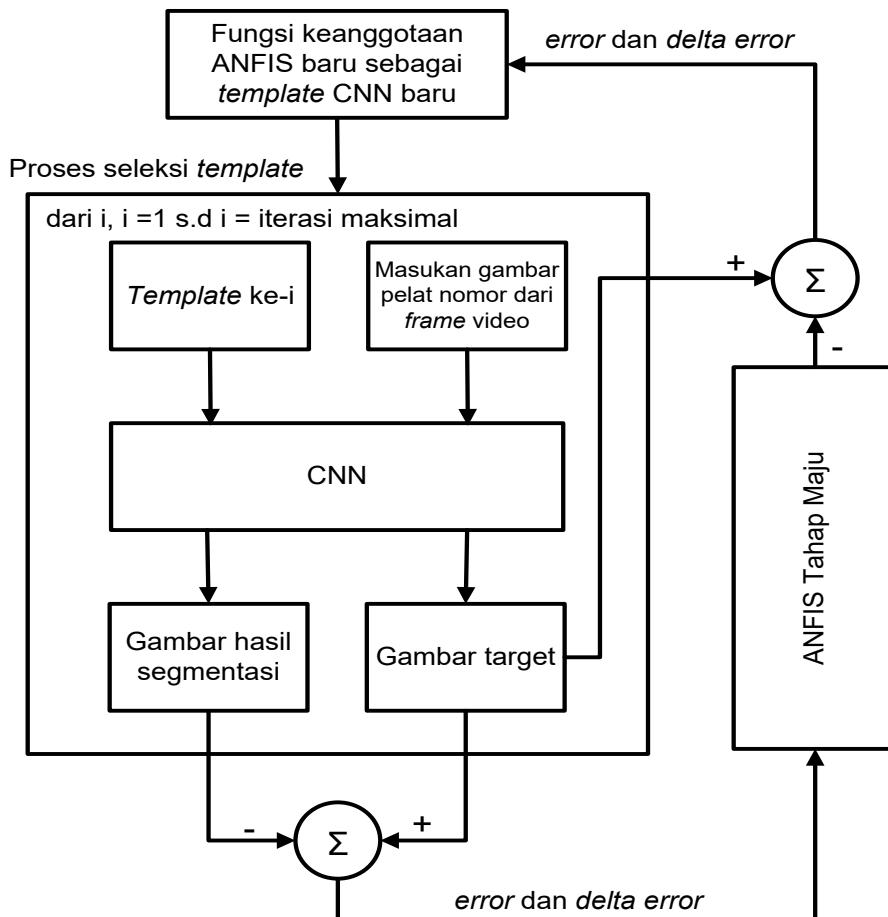
P_{baru} sebagai variasi baru dari P_{i+1} dan P_{lama} berarti $P_{i+1} = P_i$.

4.2.4.2. Optimasi *Template CNN* dengan Neuro-Fuzzy

Sistem Neuro-Fuzzy yang digunakan untuk optimasi *template CNN* adalah *Adaptive-Network-Based Fuzzy Inference System* atau *Adaptive Neuro-Fuzzy Inference System* (ANFIS) yang dijabarkan oleh (Jang, 1993), (Tjahjono *et al.*, 2014). Dalam proses ini, ANFIS diimplementasikan dalam rangka untuk menemukan nilai yang didefinisikan sebagai *template* umpanbalik dan *template* kontrol. Nilai *template CNN* menggunakan matriks standar seperti yang digunakan *Adaptive Fuzzy* yaitu Persamaan (4.10):

$$A = \begin{bmatrix} a1 & a2 & a3 \\ a4 & a5 & a6 \\ a7 & a8 & a9 \end{bmatrix}, \quad B = \begin{bmatrix} b1 & b2 & b3 \\ b4 & b5 & b6 \\ b7 & b8 & b9 \end{bmatrix}$$

Untuk mendapatkan berbagai variasi fungsi keanggotaan ANFIS yang berarti variasi *template CNN*, proses ini ditunjukkan pada Gambar 4.10. Proses ini mirip optimasi dengan *Adaptive Fuzzy* dengan tambahan proses ANFIS Tahap Maju (*Forward Pass*).



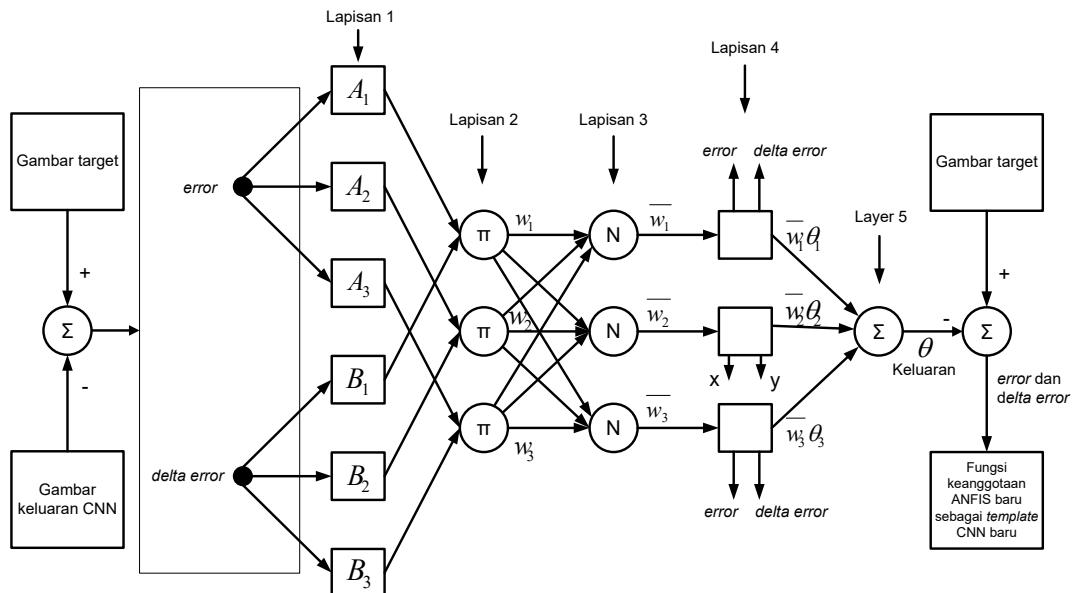
Gambar 4.10. Pembangkitan variasi template CNN menggunakan ANFIS

Error dan *delta error* didefinisikan sebagai masukan dari ANFIS dengan struktur ANFIS seperti pada Gambar 4.11. Langkah-langkah dari algoritma Neuro-Fuzzy untuk memproses penalaan *template CNN* dengan menggunakan ANFIS Tahap Maju dijelaskan sebagai berikut.

Lapisan 1:

Dalam lapisan ini, fungsi bell digunakan sebagai fungsi keanggotaan yang diberikan oleh Persamaan (4.15):

$$\begin{aligned} A_1 &= \text{bell}(\text{error}; a1, a2, a3) & B_1 &= \text{bell}(\text{delta error}; b1, b2, b3) \\ A_2 &= \text{bell}(\text{error}; a4, a5, a6) & B_2 &= \text{bell}(\text{delta error}; b4, b5, b6) \\ A_3 &= \text{bell}(\text{error}; a7, a8, a9) & B_3 &= \text{bell}(\text{delta error}; b7, b8, b9) \end{aligned} \quad (4.15)$$



Gambar 4.11. Struktur ANFIS

Lapisan 2:

Dalam lapisan ini, logika fuzzy AND digunakan dalam fungsi simpul. Sehingga keluaran dari lapisan ini mengikuti Persamaan (4.16).

$$\begin{aligned} w_1 &= \min(A_1, B_1) \\ w_2 &= \min(A_2, B_2) \\ w_3 &= \min(A_3, B_3) \end{aligned} \quad (4.16)$$

Lapisan 3:

Sinyal input dari lapisan ini dinormalisasi, proses normalisasi diberikan oleh Persamaan (4.17).

$$\begin{aligned}\overline{w_1} &= \frac{w_1}{w_1 + w_2 + w_3} \\ \overline{w_2} &= \frac{w_2}{w_1 + w_2 + w_3} \\ \overline{w_3} &= \frac{w_3}{w_1 + w_2 + w_3}\end{aligned}\tag{4.17}$$

Lapisan 4:

Dengan pengaturan sinyal-sinyal yang masuk, dengan *error* dinyatakan sebagai *er* dan *delta error* sebagai *dr* diperoleh matriks seperti pada Persamaan (4.18).

$$\xi = [(\overline{w_1} er) (\overline{w_1} dr) \overline{w_1} (\overline{w_2} er) (\overline{w_2} dr) \overline{w_2} (\overline{w_3} er) (\overline{w_3} dr) \overline{w_3}] \tag{4.18}$$

Dengan menggunakan metode *Least Square Estimate* (LSE), parameter konsekuensi $\alpha = [\alpha_1 \dots \alpha_9]$ diperoleh dengan menggunakan Persamaan (4.19).

$$\alpha = [\xi^T \xi]^{-1} \xi^T \tau \tag{4.19}$$

τ adalah keluaran ANFIS yang diinginkan. Parameter konsekuensi $\alpha = [\alpha_1 \dots \alpha_9]$ selanjutnya digunakan untuk menghitung θ_1 , θ_2 dan θ_3 dengan menggunakan Persamaan (4.20).

$$\begin{aligned}\theta_1 &= \alpha_1 er + \alpha_2 dr + \alpha_3 \\ \theta_2 &= \alpha_4 er + \alpha_5 dr + \alpha_6 \\ \theta_3 &= \alpha_7 er + \alpha_8 dr + \alpha_9\end{aligned}\tag{4.20}$$

Setelah itu, keluaran dari simpul pada lapisan 4 diperoleh dengan komputasi $\overline{w_1} \theta_1$, $\overline{w_2} \theta_2$ dan $\overline{w_3} \theta_3$.

Lapisan 5:

Akhirnya, keluaran dari lapisan ini menjadi keluaran ANFIS yang merupakan penjumlahan dari sinyal keluaran lapisan 4 yang diberikan oleh Persamaan (4.21).

$$\theta = \overline{w_1} \theta_1 + \overline{w_2} \theta_2 + \overline{w_3} \theta_3 \tag{4.21}$$

Proses selanjutnya adalah membandingkan keluaran dari ANFIS dengan gambar target yang dianggap pelat nomor ideal yang diinginkan. Dari perbandingan ini, diperoleh selisih (*error*). *Error* dan *delta error* kemudian digunakan sebagai

Fungsi Keanggotaan ANFIS sebagai variasi baru *template* CNN. Selanjutnya, keluaran CNN dibandingkan dengan gambar target dan dihasilkan *error* lagi. *Error* dan *delta error* dijadikan masukan ANFIS dan kemudian diproses dengan menggunakan ANFIS Tahap Maju untuk menghasilkan keluaran ANFIS. Keluaran ANFIS kemudian dibandingkan dengan target diperoleh *error* lagi yang digunakan untuk membangkitkan variasi *template* CNN yang baru. Proses ini diulang terus menerus sampai nilai RMSE diterima atau iterasi dihentikan.

Persamaan parameter Fungsi Keanggotaan ANFIS sebagai variasi *template* CNN yang baru diperbarui dengan menggunakan definisi yang sama seperti parameter Fungsi Keanggotaan *Adaptive Fuzzy* dalam Persamaan (4.13). Untuk mendapatkan parameter Fungsi Keanggotaan ANFIS terbaik yang berarti *template* CNN terbaik, itu dilakukan dengan mengulangi percobaan beberapa kali dan parameter terbaik yang memenuhi kriteria dipilih. Atau, selama proses tersebut, perubahan parameter diproses menggunakan aturan konvergensi kesalahan yang didefinisikan sebagai RMSE seperti pada Persamaan (4.14).

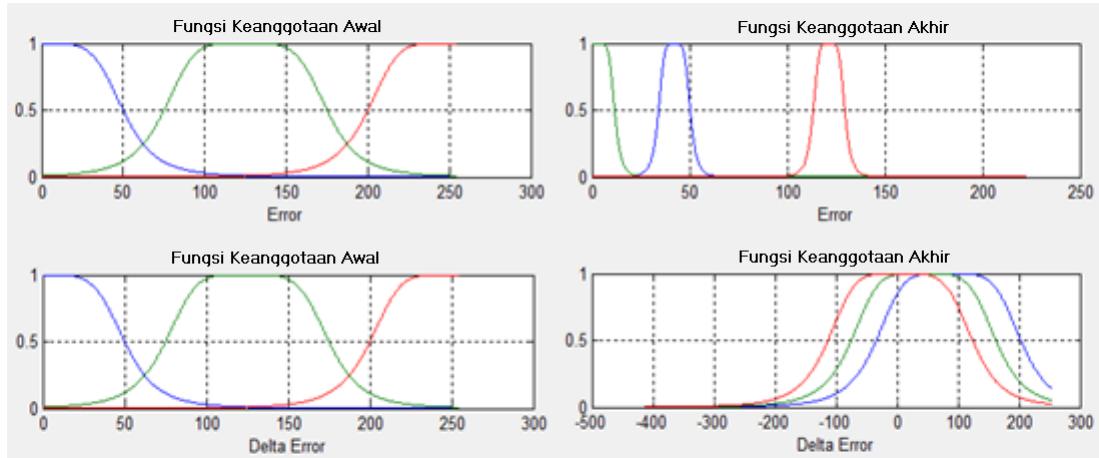
4.2.5. Pengujian Segmentasi Karakter Pelat Nomor Kendaraan dengan BIM-HCNN

Metode BIM-HCNN diuji untuk segmentasi karakter pelat nomor kendaraan Indonesia. Hasil dari proses segmentasi diharapkan mendekati target gambar pelat nomor ideal. Percobaan telah dilakukan beberapa kali untuk memproses segmentasi karakter pelat nomor kendaraan Indonesia. Hasil uji coba berupa *template* umpanbalik dan *template* kontrol hasil optimasi, dan bias (*I*) disajikan dalam Tabel 4.3:

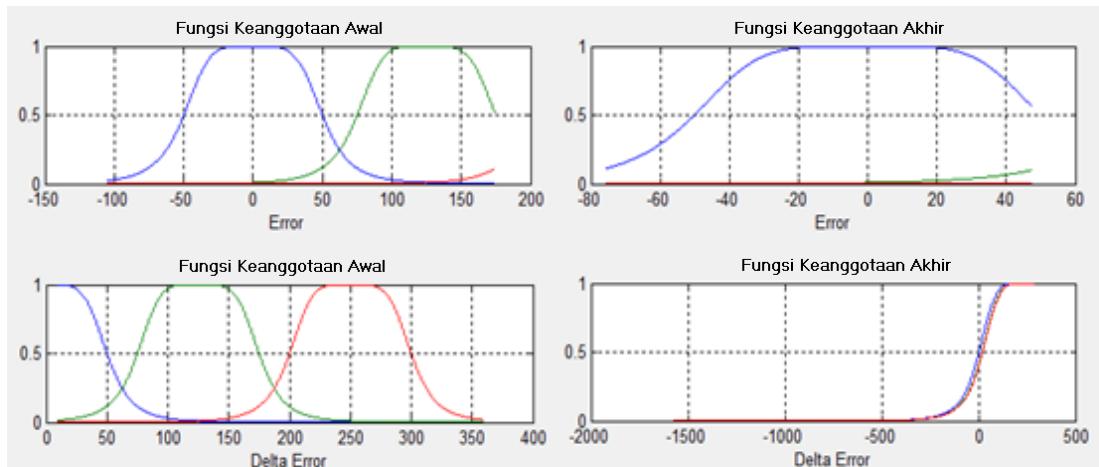
Tabel 4.3. Nilai template hasil optimasi

Template berbasis optimasi fuzzy	Template berbasis optimasi ANFIS
$A = \begin{bmatrix} 8.11 & 2.50 & 42.25 \\ 8.11 & 2.50 & 3.53 \\ 8.11 & 2.50 & 121.47 \end{bmatrix}$ $B = \begin{bmatrix} 118.54 & 2.50 & 84.41 \\ 118.54 & 2.50 & 43.54 \\ 118.54 & 2.50 & 5.18 \end{bmatrix}$ $I = 0$	$A = \begin{bmatrix} 50.00 & 2.50 & 0.01 \\ 50.00 & 2.50 & 125.00 \\ 50.00 & 2.50 & 250.00 \end{bmatrix}$ $B = \begin{bmatrix} 229.90 & 2.50 & 232.64 \\ 229.90 & 2.50 & 249.12 \\ 229.90 & 2.50 & 251.26 \end{bmatrix}$ $I = 0$

Fungsi Keanggotaan Fuzzy awal dan akhir yang diperoleh dari hasil proses BIM-HCNN dengan algoritma *Adaptive Fuzzy* sebagai fungsi *error* dan *delta error* sebelum dan setelah proses optimasi ditunjukkan pada Gambar 4.12. Untuk BIM-HCNN dengan ANFIS, Fungsi Keanggotaan Fuzzy awal dan akhir ditunjukkan pada Gambar 4.13.



Gambar 4.12. Fungsi Keanggotaan Fuzzy awal dan akhir BIM-HCNN dengan Adaptive Fuzzy



Gambar 4.13. Fungsi Keanggotaan Fuzzy awal dan akhir BIM-HCNN dengan ANFIS

Dari Gambar 4.12, warna biru, hijau dan merah masing-masing menunjukkan grafik Fungsi Keanggotaan fuzzy *small error* (SE), *medium error* (ME), dan *big error* (BE). Tampak sebelum dan sesudah proses optimasi *template CNN* dengan *adaptive fuzzy* mengalami perubahan bentuk, hal ini menunjukkan terjadi perubahan parameter Fungsi Keanggotaan fuzzy. Dengan kata lain proses optimasi berjalan (penalaan *template CNN* berjalan) sesuai yang diharapkan.

Dari Gambar 4.13, warna biru, hijau dan merah masing-masing menunjukkan grafik Fungsi Keanggotaan fuzzy *small delta error* (SDE), *medium delta error* (MDE) dan *big delta error* (BDE). Tampak sebelum dan sesudah proses optimasi *template* CNN dengan neuro-fuzzy mengalami perubahan bentuk, hal ini menunjukkan terjadi perubahan parameter Fungsi Keanggotaan fuzzy. Dengan kata lain proses optimasi berjalan sesuai yang diharapkan.

Hasil beberapa percobaan menggunakan gambar pelat nomor kendaraan yang berbeda ditunjukkan pada Gambar 4.14. Tiga pengukuran kualitas gambar, seperti *Peak signal-to-noise ratio* (PSNR), *Mean Squared Error* (MSE) dan *Root Mean Squared Error* (RMSE) digunakan untuk mengukur perbedaan antara dua gambar. Hasil perhitungan dengan menggunakan PSNR, MSE, dan pengukuran kualitas gambar RMSE ditunjukkan pada Tabel 4.4. Untuk menghitung MSE dan PSNR, menggunakan Persamaan (4.22) (Almohammad and Ghinea, 2010):

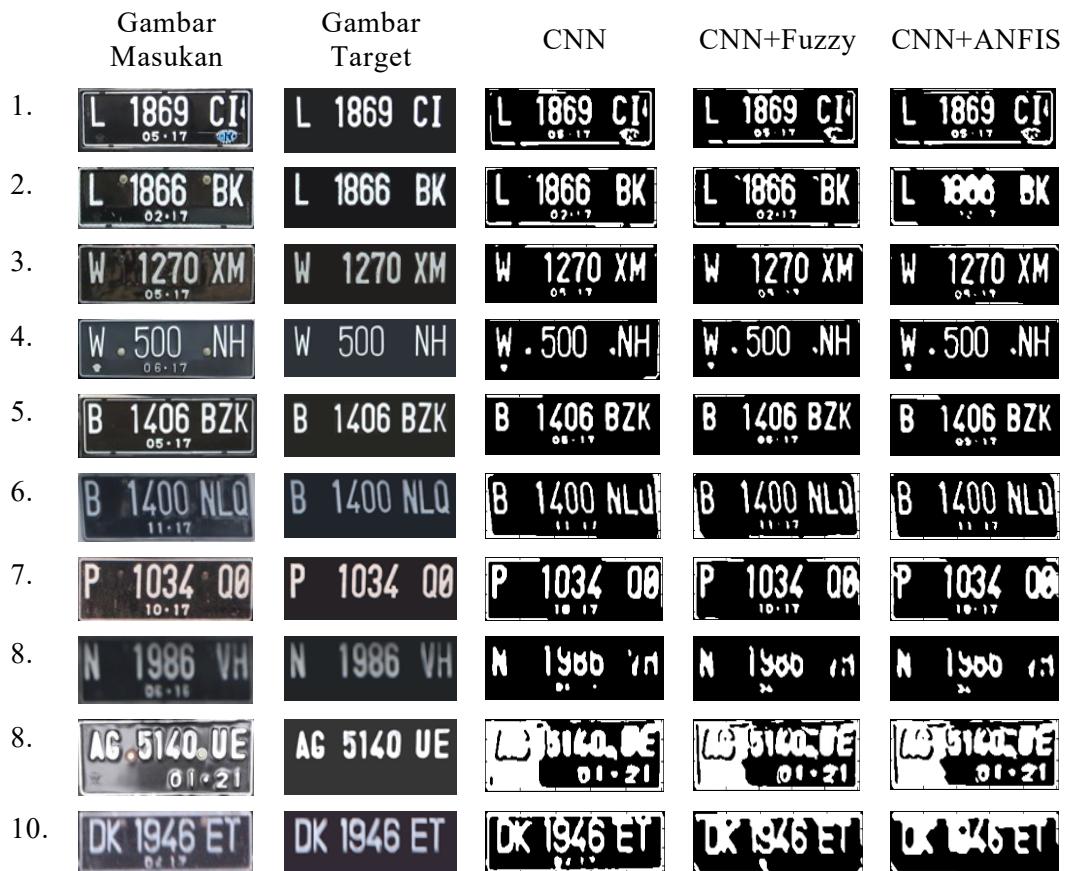
$$MSE = \left(\frac{1}{MN} \right) \sum_{i=1}^M \sum_{j=1}^N (\rho_{ij} - \overline{\rho}_{ij})^2 \quad (4.22)$$

$$PSNR = 10 \cdot \log_{10} \frac{\psi^2}{MSE} \text{ dB}$$

Keterangan:

- ρ_{ij} : baris ke-*i* dan kolom ke-*j* piksel dari gambar target,
- $\overline{\rho}_{ij}$: baris ke-*i* dan kolom ke-*j* piksel dari gambar hasil segmentasi,
- M* dan *N* : tinggi dan lebar gambar,
- ψ : rentang dinamis nilai piksel, atau nilai maksimum piksel (setara dengan (255) untuk gambar 8-bit).

Hasil simulasi yang diharapkan adalah nilai-nilai kecil untuk MSE dan RMSE dan besar untuk PSNR. Ini berarti perbedaan antara piksel di area target mendekati nol. Hasil pengujian secara lengkap disajikan pada Gambar 4.14 dan Tabel 4.4.



Gambar 4.14. Hasil segmentasi karakter pelat nomor kendaraan Indonesia

Tabel 4.4. Hasil pengukuran kualitas gambar

No	Pelat Nomor	CNN			CNN+Adaptive Fuzzy			CNN+ANFIS		
		PSNR	MSE	RMSE	PSNR	MSE	RMSE	PSNR	MSE	RMSE
1	L 1869 CI	10.01	191.25	13.83	10.13	193.01	13.89	10.05	190.61	13.81
2	L 1866 BK	10.07	186.48	13.66	10.25	186.16	13.64	10.39	190.85	13.81
3	W 1270 XM	11.25	207.34	14.40	11.74	207.91	14.42	11.34	205.76	14.34
4	W 500 NH	11.61	215.87	14.69	11.72	214.59	14.65	11.68	214.56	14.65
5	B 1406 BZK	12.26	208.94	14.45	12.39	210.82	14.52	12.26	207.31	14.40
6	B 1400 NLQ	9.99	200.54	14.16	10.07	199.36	14.12	10.01	199.01	14.11
7	P 1034 QO	10.33	189.97	13.78	10.67	193.42	13.91	10.55	190.50	13.80
8	N 1986 VH	12.34	221.06	14.87	12.45	223.08	14.94	12.47	223.17	14.94
9	AG 5140 UE	5.77	111.27	10.55	5.77	110.21	10.50	5.73	108.65	10.42
10	DK 1946 ET	7.69	153.15	12.38	7.86	153.87	12.40	7.94	152.18	12.34
	Rata-rata	10.13	188.59	13.68	10.31	189.24	13.70	10.24	188.26	13.66

Seperi dapat dilihat pada Tabel 4.4, *template* CNN yang dioptimasi oleh algoritma *Adaptive Fuzzy* menghasilkan nilai rata-rata PSNR lebih tinggi daripada CNN standar. Namun menghasilkan nilai rata-rata MSE dan RMSE yang kurang baik dibandingkan dengan CNN standar. Sedangkan template CNN yang dioptimasi oleh algoritma ANFIS menghasilkan nilai rata-rata PSNR lebih tinggi daripada CNN standar dan menghasilkan nilai rata-rata MSE dan RMSE yang lebih baik (lebih kecil) dibandingkan dengan CNN standar. Dengan demikian BIM-HCNN dengan ANFIS lebih unggul dibandingkan CNN standar maupun BIM-HCNN dengan *Adaptive Fuzzy*.

Jika dianalisis perbandingan secara “*head to head*”, *template* CNN yang dioptimasi oleh algoritma *Adaptive Fuzzy*, menghasilkan nilai PSNR lebih tinggi daripada CNN standar pada pelat nomor kendaraan ke: 1, 2, 3, 4, 5, 6, 7, 8, 10 (sembilan dari sepuluh atau 90%). Namun, menghasilkan nilai MSE dan RMSE yang lebih kecil dari CNN standar hanya pada pelat nomor kendaraan ke: 2, 4, 6, 9 (empat dari sepuluh atau 40%). Sedangkan untuk *template* CNN yang dioptimasi oleh algoritma ANFIS, menghasilkan nilai PSNR lebih tinggi daripada CNN standar pada pelat nomor kendaraan ke: 1, 2, 3, 4, 6, 7, 8 , 10 (delapan dari sepuluh atau 80%). Dan menghasilkan nilai MSE dan RMSE yang lebih kecil dari CNN standar pada pelat nomor kendaraan ke: 1, 3, 4, 5, 6, 9, 10 (tujuh dari sepuluh atau 70%).

4.2.6. Ringkasan Hasil

Secara ringkas, dari hasil perhitungan PSNR, *template* CNN yang dioptimasi oleh algoritma ANFIS menghasilkan kinerja 80% lebih baik daripada CNN standar, dan nilai MSE dan RMSE 70% lebih baik daripada CNN standar. Sedangkan template CNN yang dioptimasi oleh algoritma *Adaptive Fuzzy* mencapai PSNR 90% lebih baik daripada CNN standar, tetapi menghasilkan nilai MSE dan RMSE 40% lebih buruk daripada CNN standar. Dengan demikian BIM-HCNN dengan ANFIS lebih direkomendasikan daripada CNN standar maupun BIM-HCNN dengan *Adaptive Fuzzy*.

Keterangan:

Publikasi yang dihasilkan dari topik ini, yaitu:

Rahmat, B., Joelianto, E., Purnama, I. K. E. and Purnomo, M. H. (2016) ‘Vehicle License Plate Image Segmentation System Using Cellular Neural Network Optimized by Adaptive Fuzzy and Neuro-Fuzzy Algorithms’, *International Journal of Multimedia and Ubiquitous Engineering*. SERSC: Science & Engineering Research Support soCiety, 11(12), pp. 383–400.

4.3. Pengenalan Karakter Pelat Nomor Kendaraan

4.3.1. *Extreme Learning Machine (ELM)*

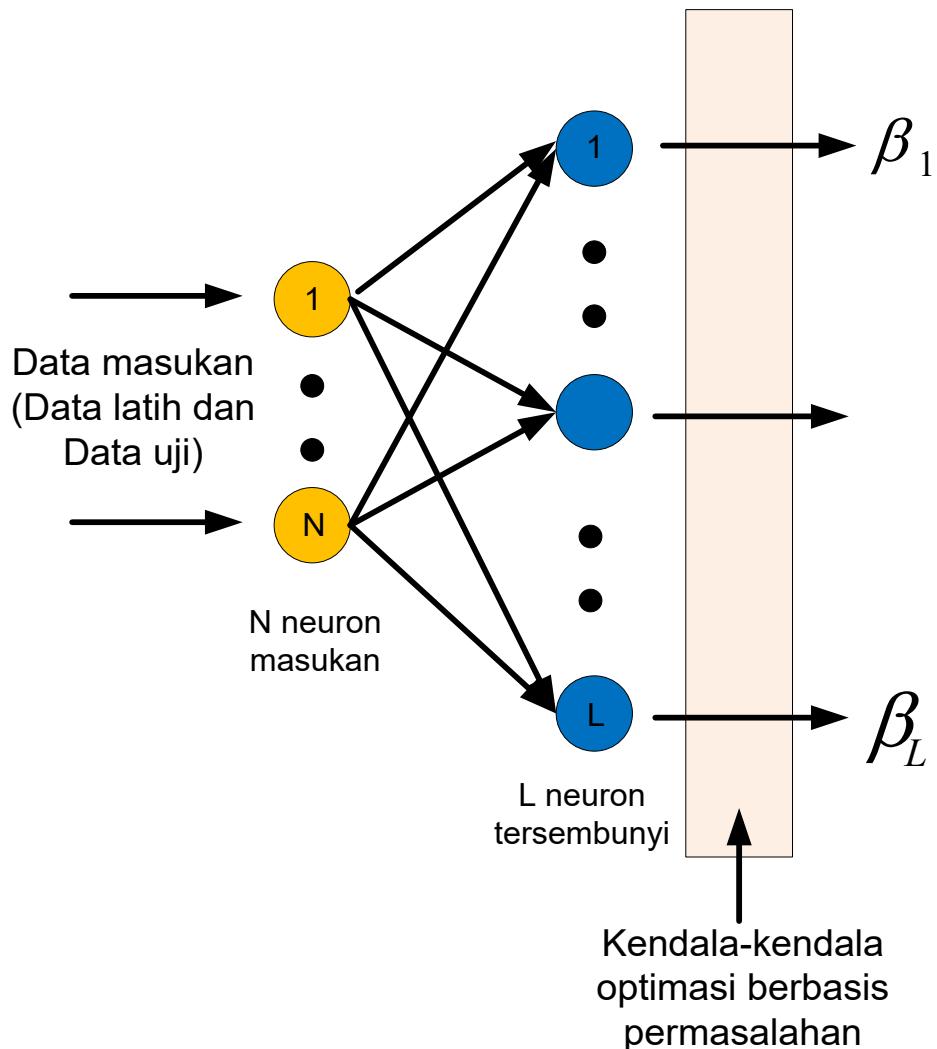
Salah satu metode yang relatif baru yang bisa digunakan untuk pengenalan karakter pelat nomor kendaraan adalah *Extreme Learning Machine (ELM)*. ELM merupakan metode pembelajaran baru dari Jaringan Syaraf Tiruan (JST). Metode ini pertama kali diperkenalkan oleh Huang (Huang, Zhu and Siew, 2006). ELM merupakan JST *feedforward* dengan *single hidden layer* atau biasa disebut dengan *Single Hidden Layer Feedforward Neural Networks (SLFNs)*.

Metode pembelajaran ELM dibuat untuk mengatasi kelemahan-kelemahan dari JST *feedforward* terutama dalam hal *learning speed*. Dua alasan mengapa JST *feedforward* mempunyai *learning speed* rendah, yaitu (Huang, Zhu and Siew, 2006):

1. Menggunakan *slow gradient based learning algorithm* untuk melakukan *training*.
2. Semua parameter pada jaringan ditentukan secara iteratif dengan menggunakan metode pembelajaran tersebut

Pada pembelajaran dengan menggunakan *Conventional gradient based learning algorithm* seperti *backpropagation* (BP) dan varian Lavenberg Marquadt (LM) semua parameter pada JST feedforward harus ditentukan secara manual. Parameter yang dimaksud adalah *input weight* dan *hidden bias*. Parameter-parameter tersebut juga saling berhubungan antara layer yang satu dengan yang lain, sehingga membutuhkan *learning speed* yang lama dan sering terjebak pada *local minima* (Huang *et al.*, 2008).

Sedangkan pada ELM parameter-parameter seperti *input weight* dan *hidden bias* dipilih secara acak, sehingga ELM memiliki *learning speed* yang cepat dan mampu menghasilkan *good generalization performance*. Gambar 4.15 merupakan struktur dari ELM (Demertzis and Iliadis, 2016).



Gambar 4.15. Struktur ELM

Metode ELM mempunyai model matematis yang berbeda dari JST *feedforward*. Model matematis dari ELM lebih sederhana dan efektif. Dengan data latih sembarang (v_i, t_i) sebanyak N , dimana $v_i = [v_{i1}, v_{i2}, \dots, v_{in}]^T \in R^n$ sebagai vektor masukan jaringan, dan $t_i = [t_{i1}, t_{i2}, \dots, t_{im}]^T \in R^m$ sebagai target yang diinginkan, SLFNs standar dengan neuron tersembunyi sebanyak \tilde{N} dan fungsi aktivasi $\phi(v)$ maka dapat dimodelkan seperti pada Persamaan (4.23) (Huang, Zhu and Siew, 2004).

$$\sum_{i=1}^N \beta_i \phi(w_i \cdot v_j + b_i) = Out_j, \quad j=1, \dots, N \quad (4.23)$$

Dimana $w_i = [w_{i1}, w_{i2}, \dots, w_{in}]^T$ adalah vektor bobot interkoneksi antara neuron tersembunyi sebanyak i dan neuron masukan, $\beta_i = [\beta_{i1}, \beta_{i2}, \dots, \beta_{im}]^T$ adalah vektor bobot interkoneksi antara neuron tersembunyi sebanyak i dan neuron keluaran, dan b_i adalah nilai ambang batas (*threshold*) dari neuron tersembunyi sebanyak i . Sedangkan $w_i \cdot v_j$ menyatakan perkalian matriks (*inner product*) antara w_i dan v_j . Keluaran jaringan SLFNs dipilih mendekati linear.

SLFNs standar dengan N neuron tersembunyi dengan fungsi aktivasi $\phi(v)$ terhadap sejumlah N sampel data latih, tujuannya adalah mendekati *error* nol, atau dengan kata lain keluaran jaringan dikurangi target sama dengan nol, atau $\sum_{j=1}^N \|Out_j - t_j\| = 0$. Jika nilai β_i, w_i dan b_i ada, maka Persamaan (4.23) dapat dinyatakan kedalam Persamaan (4.24) (Huang, Zhu and Siew, 2004).

$$\sum_{i=1}^N \beta_i \phi(w_i \cdot v_j + b_i) = t_j, \quad j=1, \dots, N \quad (4.24)$$

Persamaan (4.24) dapat dituliskan secara sederhana seperti pada Persamaan (4.25) (Huang, Zhu and Siew, 2004).

$$H \beta = T \quad (4.25)$$

Dimana

$$\begin{aligned} H(w_1, \dots, w_N, b_1, \dots, b_N, v_1, \dots, v_N) \\ = & \begin{bmatrix} \phi(w_1 \cdot v_1 + b_1) & \Lambda & \phi(w_N \cdot v_1 + b_N) \\ M & \Lambda & M \\ \phi(w_1 \cdot v_N + b_1) & \Lambda & \phi(w_N \cdot v_N + b_N) \end{bmatrix}_{N \times N} \end{aligned} \quad (4.26)$$

$$\beta = \begin{bmatrix} \beta_1^T \\ M \\ \beta_N^T \end{bmatrix}_{N \times m} \quad \text{dan} \quad T = \begin{bmatrix} t_1^T \\ M \\ t_N^T \end{bmatrix}_{N \times m} \quad (4.27)$$

H adalah matriks keluaran dari lapisan tersembunyi dari jaringan SLFNs atau ELM. Kolom sejumlah i dari H adalah vektor keluaran sejumlah i neuron-neuron tersembunyi akibat masukan v_1, v_2, \dots, v_N .

Secara garis besar tahapan metode ELM dibagi menjadi tiga tahap yaitu pemrosesan awal (*preprocessing*), pelatihan (*training*), dan pengujian (*testing*). Dalam pengenalan karakter pelat nomor kendaraan, digunakan untuk mengenali huruf dan angka yang terdapat pada pelat nomor kendaraan. Proses pelatihan dilakukan setelah dilakukan proses segmentasi karakter pelat nomor kendaraan. Demikian juga pengujian (*testing*), namun diterapkan pada pelat nomor kendaraan lain diluar yang dilakukan pelatihan.

4.3.2. Pengenalan Karakter Pelat Nomor Kendaraan dengan ELM

Pengenalan karakter pelat nomor kendaraan dengan ELM dirancang seperti pada Gambar 4.16. Untuk penentuan akurasi pengenalan karakter tiap pelat nomor kendaraan, dilakukan dengan cara menghitung jumlah karakter yang berhasil dikenali dibagi dengan jumlah seluruh karakter yang terdapat pada pelat nomor kendaraan, dikalikan dengan 100%. Dengan N jumlah karakter pada pelat nomor kendaraan yang diuji, maka perhitungan akurasi didefinisikan seperti pada Persamaan (4.28).

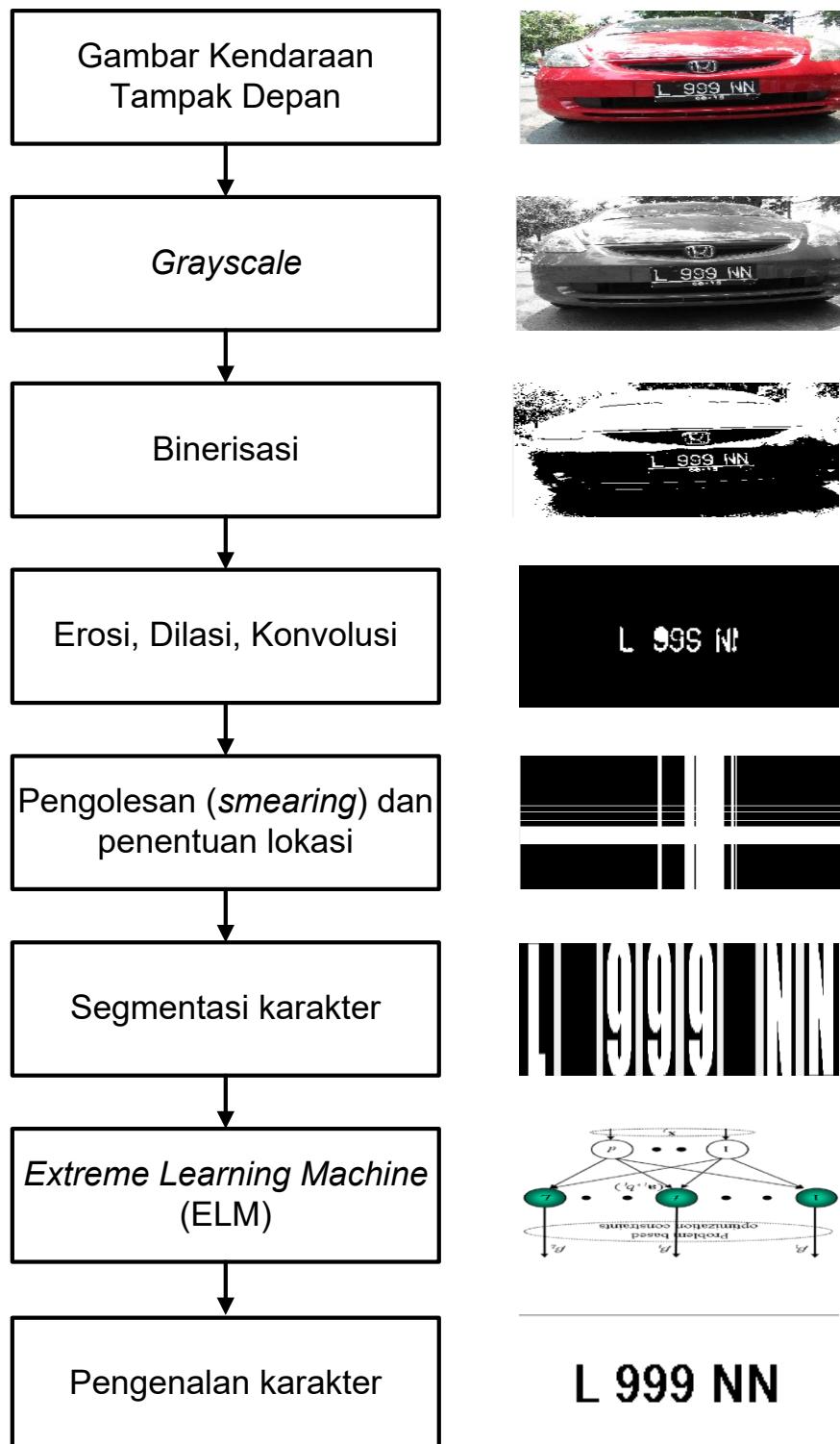
$$\text{Akurasi (\%)} = \frac{\sum_{i=1}^N \text{Karakter yang dikenali (i)}}{N} \times 100\% \quad (4.28)$$

4.3.3. Pengujian Pengenalan Karakter Pelat Nomor Kendaraan dengan ELM

Setelah mengimplementasikan metode ELM yang diusulkan, tahap selanjutnya adalah melakukan uji coba. Uji coba dilakukan untuk melihat unjuk kerja yang dihasilkan oleh metode ELM terhadap pengenalan pelat nomor kendaraan. Data masukan pada uji coba ini berupa gambar kendaraan tampak depan. Dengan menerapkan proses *grayscale*, binerisasi, erosi, dilasi dan konvolusi, serta pengolesan (*smearing*) dan penentuan lokasi, serta segmentasi karakter, maka diperoleh keluaran berupa karakter hasil proses segmentasi. Karakter hasil proses segmentasi inilah yang selanjutnya akan dikenali menggunakan algoritma ELM.

Data latih yang digunakan dalam penelitian ini adalah gambar angka dan huruf yang merupakan karakter pelat nomor kendaraan. Yaitu angka 0-9 dan huruf A s.d Z.

Dari hasil proses pelatihan ELM diperoleh hasil seperti terlihat pada Tabel 4.5. Dan hasil pengujian pengenalan karakter pelat nomor kendaraan dengan menggunakan ELM dibandingkan dengan *Template Matching*, disajikan pada Tabel 4.6.



Gambar 4.16. Pengenalan karakter pelat nomor kendaraan dengan ELM

Tabel 4.5. Hasil proses pelatihan ELM

Data Latih Karakter	Jumlah Neuron Tersem-bunyi	Hasil Proses Pelatihan	
		RMSE	Waktu Proses (detik)
0-9 dan A-Z	1000	5.6459e-14	31.5781

Tabel 4.6. Hasil pengenalan karakter pelat nomor kendaraan dengan ELM

NO	Gambar JPG	Skala keabuan	Binerisasi	Erosi, Dilasi, Konvolusi	Pengolesan dan penentuan lokasi	Hasil Segmentasi	Template Matching		ELM	
							Hasil	Akurasi (%)	Hasil	Akurasi (%)
1							L 90 CL	71.4286	K503CL	71.4286
2							W 68 XN	66.6667	W545XN	66.6667
3							L 1959 RL	57.1429	LT9690L	71.4286
4							L 1346 EO	71.4286	LT346EL	71.4286
5							L 1793 P	83.3333	LY793P	83.3333
6							W 1233 XW	71.4286	W7233XW	71.4286
7							LT539 QS	71.4286	LT5390L	71.4286
8							L 999 NN	66.6667	L999A4P	66.6667
9							L 1467 ES	71.4286	L1467S	71.4286
10							D1367 NN	71.4286	D4367NO	57.1429
11							L 518 SV	71.4286	L5T8SV	71.4286
12							L 883 LR	85.7143	L8B3LR	71.4286
13							BM95 NV	71.4286	BM95NV	71.4286
14							BK239CB	57.1429	BK239CB	71.4286
15							W 592 PO	66.6667	KU92PO	66.6667
	Hasil Akurasi Rata-rata								70.3175	70.3175

Dari 15 pengujian pengenalan karakter pelat nomor kendaraan, diperoleh hasil sebagai berikut. ELM dibandingkan dengan *Template Matching*, 11 kali pengujian diperoleh hasil yang sama, 2 kali lebih baik, yaitu pada pengujian ke-3 dan ke-14, dan 2 kali lebih buruk, yaitu pada pengujian ke-10 dan ke-12. Hasil akhir rata-rata akurasi kedua metode sama, yaitu 70.3175 %.

4.3.4. Ringkasan Hasil

Pengenalan Karakter Pelat Nomor Kendaraan dengan ELM dibandingkan dengan metode *template matching*, diperoleh hasil akhir rata-rata akurasi kedua metode sama, yaitu 70.3175%.

BAB 5

KESIMPULAN & RENCANA PENELITIAN SELANJUTNYA

5.1. Kesimpulan

Dari keseluruhan pengujian sistem, hasil dari penelitian ini dapat disimpulkan antara lain sebagai berikut:

1. *Smearing Algorithm* dapat digunakan sebagai salah satu metode ekstraksi pelat nomor kendaraan.
2. Pelacakan pelat nomor kendaraan dengan metode BIM-Mean Shift baik dengan *Switching Kernel* maupun *Adaptive Fuzzy Gaussian Kernel*, masing-masing menghasilkan kinerja yang lebih baik daripada Mean Shift standar. Pelacakan pelat nomor kendaraan dengan BIM-Mean Shift dengan *Switching Kernel* memberikan rata-rata persentase akurasi pelacakan 71.57 % lebih baik daripada Mean Shift standar Gaussian yang hanya 68.51%. Sedangkan BIM-Mean Shift dengan *Adaptive Fuzzy Gaussian Kernel* memberikan rata-rata persentase akurasi pelacakan sebesar 66.0970%, lebih baik daripada Mean Shift standar dengan *Static Gaussian Kernel* yang hanya 54.5228%, namun dengan keseluruhan ujicoba 4 video (100%) secara “*head to head*” lebih unggul daripada Mean Shift standar.
3. Ekstraksi karakter pelat nomor kendaraan, untuk proses segmentasi karakter pelat nomor kendaraan menggunakan BIM-HCNN dengan ANFIS menghasilkan kinerja 80% lebih baik daripada CNN standar, dan nilai MSE dan RMSE 70% lebih baik daripada CNN standar. Sedangkan proses segmentasi karakter pelat nomor kendaraan menggunakan BIM-HCNN dengan *Adaptive Fuzzy* menghasilkan PSNR 90% lebih baik daripada CNN standar, tetapi menghasilkan nilai MSE dan RMSE 40% lebih buruk daripada CNN standar.
4. Pengenalan Karakter Pelat Nomor Kendaraan dengan ELM dibandingkan dengan metode *template matching*, diperoleh hasil akhir rata-rata akurasi kedua metode sama, yaitu 70.3175%.

5.2. Rencana Penelitian Selanjutnya

Sesuai dengan Gambar 1.2 dan Gambar 1.3 pada Bab 1 mengenai gambaran umum dan posisi penelitian pelacakan dan pengenalan pelat nomor kendaraan Indonesia diantara beberapa penelitian yang ada didunia saat ini, maka rencana pengembangan selanjutnya tetap bisa dilakukan pada pengembangan metode/algoritma pelacakan dan pengenalan pelat nomor kendaraan dengan memasukkan ide-ide dan *novelty* baru.

Selain itu, untuk Ekstraksi Pelat Nomor Kendaraan, bisa dikembangkan ke arah otomatis, sehingga menjadi *Automatic Indonesia Vehicle License Plate Extraction* atau *Automatic Indonesia Vehicle License Plate Localization*. Pelacakan Pelat Nomor Kendaraan dengan BIM-Mean Shift, bisa dilanjutkan dengan penambahan algoritma sistem cerdas yang lain sehingga menjadi *Robust BIM-Mean Shift for Indonesia Vehicle License Plate Tracking Video*.

Sedangkan untuk Ekstraksi Karakter Pelat Nomor Kendaraan, baik untuk proses Segmentasi Karakter Pelat Nomor Kendaraan maupun Pengenalan Karakter Pelat Nomor Kendaraan banyak sekali peluang riset pengembangan algoritma sistem cerdas yang dapat diterapkan untuk meningkatkan akurasi pengenalan karakternya, seperti *Modified Extreme Learning Machine*, *Deep Convolutional Extreme Learning Machine*, dan lain-lain.

Pengembangan lebih lanjut untuk pelacakan dan pengenalan pelat nomor kendaraan Indonesia ini dapat dikembangkan ke arah *real time* dan berbasis *Mobile Technology* serta penanganan *big data* dari *intelligent transportastion system* berbasis *Cloud Computing*.

DAFTAR PUSTAKA

- Almohammad, A. and Ghinea, G. (2010) ‘Stego image quality and the reliability of PSNR’, in, pp. 215–220. doi: 10.1109/IPTA.2010.5586786.
- Anakavej, T., Kawewong, A. and Patanukhom, K. (2014) ‘License plate localization using MSERs and vehicle frontal mask localization using visual saliency for vehicle recognition’, in *2014 Fourth International Conference on Digital Information and Communication Technology and its Applications (DICTAP)*, pp. 31–36. doi: 10.1109/DICTAP.2014.6821652.
- Anandhalli, M. and Baligar, V. P. (2017) ‘A novel approach in real-time vehicle detection and tracking using Raspberry Pi’, *Alexandria Engineering Journal*. doi: <http://dx.doi.org/10.1016/j.aej.2017.06.008>.
- Azghani, M., Aghagolzadeh, A., Ghaemi, S. and Kouzehgar, M. (2010) ‘Intelligent modified mean shift tracking using genetic algorithm’, in *2010 5th International Symposium on Telecommunications*, pp. 806–811. doi: 10.1109/ISTEL.2010.5734133.
- Balya, D. and Gal, V. (2006) ‘Analogic Implementation of the Genetic Algorithm’, in *Cellular Neural Networks and Their Applications, 2006. CNNA '06. 10th International Workshop on*, pp. 1–6. doi: 10.1109/CNNA.2006.341617.
- Bang, S. H., Sheu, B. J. and Chou, E. Y. (1996) ‘A hardware annealing method for optimal solutions on cellular neural networks’, *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on*, 43(6), pp. 409–421. doi: 10.1109/82.502314.
- Bang, S. H., Sheu, B. J. and Wu, T. H.-Y. (1996) ‘Optimal solutions for cellular neural networks by paralleled hardware annealing’, *Neural Networks, IEEE Transactions on*, 7(2), pp. 440–454. doi: 10.1109/72.485679.
- Basturk, A. and Gunay, E. (2009) ‘Efficient edge detection in digital images using a cellular neural network optimized by differential evolution algorithm’, *Expert Systems with Applications*, 36(2, Part 2), pp. 2645–2650. doi: <http://dx.doi.org/10.1016/j.eswa.2008.01.082>.
- Bernhardt, S. (2016) ‘Mean-shift video tracking’.
- Chan, S., Zhou, X. and Chen, S. (2015) ‘Online learning for classification and object tracking with superpixel’, in *2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 1758–1763. doi: 10.1109/ROBIO.2015.7419026.
- Chang, Y., Su, Z. and Qian-Yu, L. (2012) ‘A New Traffic Incident Detection Method Under Low-Volume Condition Based On Automatic Vehicle Identification’, in *2012 9th International Conference on Fuzzy Systems and Knowledge Discovery*, pp. 2853–2859. doi: 10.1109/FSKD.2012.6233788.
- Chen, K., Fu, S., Song, K. and Jhun, C. G. (2012) ‘A Meanshift-based imbedded computer vision system design for real-time target tracking’, in *Computer Science Education (ICCSE), 2012 7th International Conference on*, pp. 1298–1303. doi: 10.1109/ICCSE.2012.6295302.
- Chua, L. O. and Roska, T. (2004) ‘Cellular Neural Networks and Visual Computing Foundations and Applications’, pp. 1–396.
- Chua, L. O. and Yang, L. (1988) ‘Cellular neural networks: theory’, *Circuits and Systems, IEEE Transactions on*, 35(10), pp. 1257–1272. doi: 10.1109/31.7600.
- Comaniciu, D., Ramesh, V. and Meer, P. (2003) ‘Kernel-based object tracking’, *IEEE*

- Transactions on Pattern Analysis and Machine Intelligence*, 25(5), pp. 564–577. doi: 10.1109/TPAMI.2003.1195991.
- Crounse, K. R. and Chua, L. O. (1995) ‘Methods for image processing and pattern formation in Cellular Neural Networks: a tutorial’, *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, 42(10), pp. 583–601. doi: 10.1109/81.473566.
- Demertzis, K. and Iliadis, L. (2016) ‘Bio-inspired hybrid intelligent method for detecting android malware’, in *Knowledge, Information and Creativity Support Systems*. Springer, pp. 289–304.
- Du, S., Ibrahim, M., Shehata, M. and Badawy, W. (2013) ‘Automatic License Plate Recognition (ALPR): A State-of-the-Art Review’, *Circuits and Systems for Video Technology, IEEE Transactions on*, 23(2), pp. 311–325. doi: 10.1109/TCSVT.2012.2203741.
- Duraisamy, M. and Jane, F. M. M. (2014) ‘cellular neural network based medical image segmentation using artificial bee colony algorithm’, in *2014 International Conference on Green Computing Communication and Electrical Engineering (ICGCCE)*, pp. 1–6. doi: 10.1109/ICGCCE.2014.6922413.
- Fasih, A., Chamberlain Chedjou, J. and Kyamakya, K. (2009) ‘Cellular Neural Networks-Based Genetic Algorithm for Optimizing the Behavior of an Unstructured Robot’, *International Journal of Computational Intelligence Systems*, 2(2), pp. 124–131. doi: 10.1080/18756891.2009.9727647.
- Fasih, A., Chedjou, J. C. and Kyamakya, K. (2008) ‘Cellular Neural Network Trainer and Template Optimization for Advanced Robot Locomotion, Based on Genetic Algorithm’, in, pp. 317–322. doi: 10.1109/MMVIP.2008.4749553.
- Feiden, D. F. and Tetzlaff, R. (2001) ‘Iterative annealing: a new efficient optimization method for cellular neural networks’, in *Image Processing, 2001. Proceedings. 2001 International Conference on*, pp. 549–552 vol.1. doi: 10.1109/ICIP.2001.959075.
- Fernández-Ares, A. J., Mora, A. M., Odeh, S. M., García-Sánchez, P. and Arenas, M. G. (2017) ‘Wireless monitoring and tracking system for vehicles: A study case in an urban scenario’, *Simulation Modelling Practice and Theory*, 73, pp. 22–42. doi: <http://dx.doi.org/10.1016/j.simpat.2016.11.004>.
- Fukunaga, K. and Hostetler, L. (1975) ‘The estimation of the gradient of a density function, with applications in pattern recognition’, *IEEE Transactions on Information Theory*, 21(1), pp. 32–40. doi: 10.1109/TIT.1975.1055330.
- Ghahnavieh, A. E., Amirkhani-Shahraki, A. and Raie, A. A. (2014) ‘Enhancing the license plates character recognition methods by means of SVM’, in *2014 22nd Iranian Conference on Electrical Engineering (ICEE)*, pp. 220–225. doi: 10.1109/IranianCEE.2014.6999536.
- Giaquinto, A. and Fornarelli, G. (2009) ‘PSO-Based Cloning Template Design for CNN Associative Memories’, *Neural Networks, IEEE Transactions on*, 20(11), pp. 1837–1841. doi: 10.1109/TNN.2009.2031870.
- Gou, C., Wang, K., Yu, Z. and Xie, H. (2014) ‘License plate recognition using MSER and HOG based on ELM’, in *Proceedings of 2014 IEEE International Conference on Service Operations and Logistics, and Informatics*, pp. 217–221. doi: 10.1109/SOLI.2014.6960724.
- Gutierrez-Osuna, R. (2013) ‘Pattern Analysis’, CSCE 666.
- He, R. and Zhu, Y. (2009) ‘A Hybrid Image Segmentation Approach Based on Mean Shift and Fuzzy C-Means’, in *2009 Asia-Pacific Conference on Information Processing*, pp. 105–108. doi: 10.1109/APCIP.2009.35.

- Hong, X. (2012) ‘Pattern Recognition’, *Machine Intelligence (SE2MI11)*.
- Huang, G.-B., Li, M.-B., Chen, L. and Siew, C.-K. (2008) ‘Incremental extreme learning machine with fully complex hidden nodes’, *Neurocomputing*. Elsevier, 71(4), pp. 576–583.
- Huang, G.-B., Zhu, Q.-Y. and Siew, C.-K. (2004) ‘Extreme learning machine: a new learning scheme of feedforward neural networks’, in *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on*, pp. 985–990.
- Huang, G.-B., Zhu, Q.-Y. and Siew, C.-K. (2006) ‘Extreme learning machine: theory and applications’, *Neurocomputing*. Elsevier, 70(1), pp. 489–501.
- Hyun, K. (Kate), Tok, A. and Ritchie, S. G. (2017) ‘Long distance truck tracking from advanced point detectors using a selective weighted Bayesian model’, *Transportation Research Part C: Emerging Technologies*, 82, pp. 24–42. doi: <http://dx.doi.org/10.1016/j.trc.2017.06.004>.
- Jain, A. and Sharma, J. (2014) ‘Classification and interpretation of characters in multi-application OCR system’, in *2014 International Conference on Data Mining and Intelligent Computing (ICDMIC)*, pp. 1–6. doi: [10.1109/ICDMIC.2014.6954231](https://doi.org/10.1109/ICDMIC.2014.6954231).
- Jang, J.-S. R. (1993) ‘ANFIS: adaptive-network-based fuzzy inference system’, *Systems, Man and Cybernetics, IEEE Transactions on*, 23(3), pp. 665–685. doi: [10.1109/21.256541](https://doi.org/10.1109/21.256541).
- Jianlai, W., Chunling, Y. and Chao, S. (2009) ‘A Novel Algorithm for Edge Detection of Remote Sensing Image Based on CNN and PSO’, in *Image and Signal Processing, 2009. CISPA '09. 2nd International Congress on*, pp. 1–5. doi: [10.1109/CISP.2009.5304415](https://doi.org/10.1109/CISP.2009.5304415).
- Khakpour, S., Pazzi, R. W. and El-Khatib, K. (2017) ‘Using clustering for target tracking in vehicular ad hoc networks’, *Vehicular Communications*, 9, pp. 83–96. doi: <http://dx.doi.org/10.1016/j.vehcom.2017.02.002>.
- Khan, J. A. and Shah, M. A. (2016) ‘Car Number Plate Recognition (CNPR) system using multiple template matching’, in *2016 22nd International Conference on Automation and Computing (ICAC)*, pp. 290–295. doi: [10.1109/IConAC.2016.7604934](https://doi.org/10.1109/IConAC.2016.7604934).
- Khan, U. A., Fasih, A., Kyamakya, K. and Chedjou, J. C. (2009) ‘Genetic Algorithm Based Template Optimization for a Vision System: Obstacle Detection’, *Theoretical Engineering (ISTET), 2009 XV International Symposium on*, pp. 1–5.
- Khattak, A. S., Raja, G., Anjum, N. and Qasim, M. (2014) ‘Integration of Mean-Shift and Particle Filter: A Survey’, in *2014 12th International Conference on Frontiers of Information Technology*, pp. 286–291. doi: [10.1109/FIT.2014.60](https://doi.org/10.1109/FIT.2014.60).
- Khoshki, R. M. and Ganesan, S. (2015) ‘Improved Automatic License Plate Recognition (ALPR) system based on single pass Connected Component Labeling (CCL) and reign property function’, in, pp. 426–431. doi: [10.1109/EIT.2015.7293378](https://doi.org/10.1109/EIT.2015.7293378).
- Kozek, T., Roska, T. and Chua, L. O. (1993) ‘Genetic algorithm for CNN template learning’, *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, 40(6), pp. 392–402. doi: [10.1109/81.238343](https://doi.org/10.1109/81.238343).
- Li, N., Zhang, D., Gu, X., Huang, L., Liu, W. and Xu, T. (2015) ‘An improved mean shift algorithm for moving object tracking’, in *Electrical and Computer Engineering (CCECE), 2015 IEEE 28th Canadian Conference on*, pp. 1425–1429. doi: [10.1109/CCECE.2015.7129489](https://doi.org/10.1109/CCECE.2015.7129489).
- Linlin, Z., Baojie, F., Benjin, L. and Yandong, T. (2009) ‘A Hybrid Tracking Method

- Based on Active Contour and Mean Shift Algorithm', in *2009 Second International Conference on Intelligent Networks and Intelligent Systems*, pp. 70–73. doi: 10.1109/ICINIS.2009.27.
- Lopez, P., Vilarino, D. L. and Cabello, D. (2000) 'Design of multilayer discrete time cellular neural networks for image processing tasks based on genetic algorithms', in *Circuits and Systems, 2000. Proceedings. ISCAS 2000 Geneva. The 2000 IEEE International Symposium on*, pp. 133–136 vol.4. doi: 10.1109/ISCAS.2000.858706.
- Luvizon, D. C., Nassu, B. T. and Minetto, R. (2017) 'A Video-Based System for Vehicle Speed Measurement in Urban Roadways', *IEEE Transactions on Intelligent Transportation Systems*, 18(6), pp. 1393–1404. doi: 10.1109/TITS.2016.2606369.
- Maggio, E. and Cavallaro, A. (2005) 'Hybrid Particle Filter and Mean Shift tracker with adaptive transition model', in *Proceedings. (ICASSP '05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, pp. 221–224. doi: 10.1109/ICASSP.2005.1415381.
- Maggio, E. and Cavallaro, A. (2011) *Video tracking: theory and practice*. John Wiley & Sons.
- Meng, H. and Zhao, J. (2005) 'CNN Template Design Method Based on GQA', in *Neural Networks and Brain, 2005. ICNNB '05. International Conference on*, pp. 941–944. doi: 10.1109/ICNNB.2005.1614774.
- Mokti, M. N. and Salam, R. A. (2008) 'Hybrid of Mean-shift and median-cut algorithm for fish segmentation', in *2008 International Conference on Electronic Design*, pp. 1–5. doi: 10.1109/ICED.2008.4786645.
- Owczarek, M., Baranski, P. and Strumillo, P. (2015) 'Pedestrian tracking in video sequences: A particle filtering approach', in *Computer Science and Information Systems (FedCSIS), 2015 Federated Conference on*, pp. 875–881. doi: 10.15439/2015F158.
- Parmaksizoglu, S., Gunay, E. and Alci, M. (2011) 'Determining cloning templates of CNN via mog: Edge detection', in, pp. 758–761. doi: 10.1109/SIU.2011.5929761.
- Parzen, E. (1962) 'On Estimation of a Probability Density Function and Mode', *Ann. Math. Statist.* The Institute of Mathematical Statistics, 33(3), pp. 1065–1076. doi: 10.1214/aoms/1177704472.
- q. Li, G. and z. Chen, Z. (2011) 'A hybrid algorithm for detecting contour of moving object based on merging Mean Shift and GVF Snake model', in *2011 4th International Congress on Image and Signal Processing*, pp. 1287–1291. doi: 10.1109/CISP.2011.6100464.
- Rabee, A. and Barhumi, I. (2014) 'License plate detection and recognition in complex scenes using mathematical morphology and support vector machines', in *IWSSIP 2014 Proceedings*, pp. 59–62.
- Rahmat, B., Joelianto, E., Purnama, I. K. E. and Purnomo, M. H. (2013) 'CNN-Fuzzy-Based Indonesia License Plate Image Segmentation System', in, pp. 1707–1711. doi: 978-1-4673-6278-8/13/\$31.00 ©2013 IEEE.
- Rahmat, B., Joelianto, E., Purnama, I. K. E. and Purnomo, M. H. (2016) 'Vehicle License Plate Image Segmentation System Using Cellular Neural Network Optimized by Adaptive Fuzzy and Neuro-Fuzzy Algorithms', *International Journal of Multimedia and Ubiquitous Engineering*. SERSC: Science & Engineering Research Support soCiety, 11(12), pp. 383–400.
- Reddy, K. R., Priya, K. H. and Neelima, N. (2015) 'Object Detection and Tracking --

- A Survey', in *2015 International Conference on Computational Intelligence and Communication Networks (CICN)*, pp. 418–421. doi: 10.1109/CICN.2015.317.
- Rosenblatt, M. (1956) 'Remarks on Some Nonparametric Estimates of a Density Function', *Ann. Math. Statist.* The Institute of Mathematical Statistics, 27(3), pp. 832–837. doi: 10.1214/aoms/1177728190.
- Saabni, R. (2015) 'Ada-boosting Extreme learning machines for handwritten digit and digit strings recognition', in *2015 Fifth International Conference on Digital Information Processing and Communications (ICDIPC)*, pp. 231–236. doi: 10.1109/ICDIPC.2015.7323034.
- Sahoo, S. P. and Ari, S. (2015) 'Automated human tracking using advanced mean shift algorithm', in *Communications and Signal Processing (ICCSP), 2015 International Conference on*, pp. 789–793. doi: 10.1109/ICCSP.2015.7322600.
- Saini, M. K. and Saini, S. (2017) 'Multiwavelet Transform Based Number Plate Detection', *Journal of Visual Communication and Image Representation*, p. doi: <http://dx.doi.org/10.1016/j.jvcir.2017.01.003>.
- Selvathi, D., Selvaraj, H. and Selvi, S. T. (2010) 'Hybrid Approach For Brain Tumor Segmentation In Magnetic Resonance Images Using Cellular Neural Networks And Optimization Techniques', *International Journal of Computational Intelligence and Applications*, 9(1), pp. 17–31. doi: 10.1142/S1469026810002781.
- Setyawan, S. B., Purwanto, D. and Mardiyanto, R. (2015) 'Visual object tracking using improved Mean Shift algorithm', in *2015 International Conference on Information Technology Systems and Innovation (ICITSI)*, pp. 1–7. doi: 10.1109/ICITSI.2015.7437677.
- Su, T.-J., Lin, T.-H. and Liu, J.-W. (2008) 'Particle Swarm Optimization for Gray-Scale Image Noise Cancellation', in *Intelligent Information Hiding and Multimedia Signal Processing, 2008. IIHMSP '08 International Conference on*, pp. 1459–1462. doi: 10.1109/IIH-MSP.2008.85.
- Su, T.-J., Wang, H.-C. and Liu, J.-W. (2007) 'Particle Swarm Optimization for Image Noise Cancellation', in *Intelligent Information Hiding and Multimedia Signal Processing, 2007. IIHMSP 2007. Third International Conference on*, pp. 95–98. doi: 10.1109/IIH-MSP.2007.237.
- Taraglio, S. and Zanella, A. (1996) 'Cellular neural networks: a genetic algorithm for parameters optimization in artificial vision applications', in *Cellular Neural Networks and their Applications, 1996. CNNA-96. Proceedings., 1996 Fourth IEEE International Workshop on*, pp. 315–320. doi: 10.1109/CNNA.1996.566592.
- Teo, L. and Wong, F. (2009) 'Smearing algorithm for vehicle parking management system', in *Proceedings of the 2nd Seminar on Engineering and Information Technology*, pp. 331–337.
- Thirumuruganathan, S. (2010) 'Introduction To Mean Shift Algorithm'.
- Tjahjono, A., Priyadi, A., Purnomo, M. H. and Pujiantara, M. (2014) 'Overcurrent relay curve modeling using adaptive neuro fuzzy inference system', in, pp. 103–108. doi: 10.1109/MICEEI.2014.7067320.
- Ukrainitz, Y. and Sarel, B. (2004) 'Mean Shift Theory and Applications', *Weizmann Institute of Science*, http://www.wisdom.weizmann.ac.il/~vision/courses/2004_2/files/mean_shift/mean_shift.ppt.
- Yang, Y., Wang, X., Wu, J., Chen, H. and Han, Z. (2015) 'An improved mean shift

- object tracking algorithm based on ORB feature matching', in *The 27th Chinese Control and Decision Conference (2015 CCDC)*, pp. 4996–4999. doi: 10.1109/CCDC.2015.7162819.
- Yuan, Y., Zou, W., Zhao, Y., Wang, X., Hu, X. and Komodakis, N. (2017) 'A Robust and Efficient Approach to License Plate Detection', *IEEE Transactions on Image Processing*, 26(3), pp. 1102–1114. doi: 10.1109/TIP.2016.2631901.
- Zeng, N., Wang, Z., Zineddin, B., Li, Y., Du, M., Xiao, L., Liu, X. and Young, T. (2014) 'Image-Based Quantitative Analysis of Gold Immunochromatographic Strip via Cellular Neural Network Approach', *IEEE Transactions on Medical Imaging*, 33(5), pp. 1129–1136. doi: 10.1109/TMI.2014.2305394.
- Zhu, M., Liu, C., Wang, J., Wang, X. and Han, Y. (2016) 'A Service-Friendly Approach to Discover Traveling Companions Based on ANPR Data Stream', in *2016 IEEE International Conference on Services Computing (SCC)*, pp. 171–178. doi: 10.1109/SCC.2016.29.