taW

;/--n

**TUGAS AKHIR – KI141502**

**IMPLEMENTASI PENGENALAN IRIS MATA MENGGUNAKAN METODE SUPPORT VECTOR MACHINES DAN HAMMING DISTANCE**

**AFDHAL BASITH ANUGRAH**

**NRP 5112100153**

**Dosen Pembimbing I**

**Dr.Eng. Nanik Suciati, S.Kom, M.Kom**

**Dosen Pembimbing II**

**Dr.Eng. Chastine Fatichah, S.Kom, M.Kom**

**JURUSAN TEKNIK INFORMATIKA**

**Fakultas Teknologi Informasi**

**Institut Teknologi Sepuluh Nopember**

**Surabaya 2016**

***[Halaman ini sengaja dikosongkan]***

**TUGAS AKHIR – KI141502**

**IMPLEMENTASI PENGENALAN IRIS MATA MENGGUNAKAN METODE *SUPPORT VECTOR MACHINES* DAN *HAMMING DISTANCE***

**AFDHAL BASITH ANUGRAH**

**NRP 5112100153**

**Dosen Pembimbing I**

**Dr.Eng. Nanik Suciati, S.Kom, M.Kom**

**Dosen Pembimbing II**

**Dr.Eng. Chastine Fatichah, S.Kom, M.Kom**

**JURUSAN TEKNIK INFORMATIKA**

**Fakultas Teknologi Informasi**

**Institut Teknologi Sepuluh Nopember**

**Surabaya 2016**

****

***[Halaman ini sengaja dikosongkan]***

**FINAL PROJECT – KI141502**

**IMPLEMENTATION OF IRIS RECOGNITION USING SUPPORT VECTOR MACHINES AND HAMMING DISTANCE**

**AFDHAL BASITH ANUGRAH**

**NRP 5112100153**

**Supervisor I**

**Dr.Eng. Nanik Suciati, S.Kom, M.Kom**

**Supervisor II**

**Dr.Eng. Chastine Fatichah, S.Kom, M.Kom**

**DEPARTMENT OF INFORMATICS**

**FACULTY OF INFORMATION TECHNOLOGY**

**INSTITUT TEKNOLOGI SEPULUH NOPEMBER**

**SURABAYA 2016**

# LEMBAR PENGESAHAN

**IMPLEMENTASI PENGENALAN IRIS MATA MENGGUNAKAN METODE SUPPORT VECTOR MACHINES DAN HAMMING DISTANCE**

**TUGAS AKHIR**

Diajukan Untuk Memenuhi Salah Satu Syarat

Memperoleh Gelar Sarjana Komputer

pada

Bidang Studi Komputasi Cerdas dan Visualisasi

Program Studi S-1 Jurusan Teknik Informatika

Fakultas Teknologi Informasi

Institut Teknologi Sepuluh Nopember

Oleh

**AFDHAL BASITH ANUGRAH**

**NRP : 5112 100 153**

Disetujui oleh Dosen Pembimbing Tugas Akhir:

1. Dr.Eng. Nanik Suciati, S.Kom, M.Kom .....................

(NIP *197104281994122001*) (Pembimbing 1)

1. Dr.Eng. Chastine Fatichah, S.Kom, M.Kom .....................

(NIP *197512202001122002*) (Pembimbing 2)

**SURABAYA**

**JUNI, 2016**

***[Halaman ini sengaja dikosongkan]***

**IMPLEMENTASI PENGENALAN IRIS MATA MENGGUNAKAN METODE SUPPORT VECTOR MACHINE DAN HAMMING DISTANCE**

|  |  |  |
| --- | --- | --- |
| **Nama Mahasiswa** | **:** | **Afdhal Basith Anugrah** |
| **NRP** | **:** | **5112100153** |
| **Jurusan** | **:** | **Teknik Informatika FTIF-ITS** |
| **Dosen Pembimbing 1** | **:** | **Dr.Eng. Nanik Suciati, S.Kom, M.Kom** |
| **Dosen Pembimbing 2** | **:** | **Dr.Eng. Chastine Fatichah, S.Kom, M.Kom** |

# *Abstrak*

*Pengenalan iris mata adalah teknik pengenalan pola yang digunakan dalam aplikasi biometrik. Teknik biometrik ini menghasilkan pola acak yang unik secara statistik. Dengan kata lain, tekstur iris mata adalah bagian yang unik pada masing-masing orang.*

*Dalam Tugas Akhir ini mengimplementasikan metode ekstraksi fitur Wavelet dan Log-Gabor Filter. Kemudian dilanjutkan dengan metode klasifikasi Support Vector Machines dan Hamming Distance. Sebelum dilakukan ekstraksi fitur, terlebih dahulu dilakukan praproses untuk identifikasi bagian iris mata dan melakukan proses normalisasi.*

*Hasil uji coba pada dataset mata CASIA dengan menggunakan klasifikasi SVM dan Hamming distance terbukti cukup efektif dalam proses pengenalan iris. Pada proses menggunakan metode klasifikasi SVM saja menghasilkan akurasi 94.44%, dengan klasifikasi Hamming distance saja menghasilkan akurasi 94.14%, dan menghasilkan 98.45% dengan menggunakan kombinasi dari SVM dan Hamming distance.*

***Kata Kunci: Pengenalan iris, Log-Gabor Filter, Hamming Distance, Wavelet, Support Vector Machines.***

***[Halaman ini sengaja dikosongkan]***

**IMPLEMENTATION OF IRIS RECOGNITION USING SUPPORT VECTOR MACHINES AND HAMMING DISTANCE**

|  |  |  |
| --- | --- | --- |
| **Student’s Name** | **:** | **Afdhal Basith Anugrah** |
| **Student’s ID** | **:** | **5112100153** |
| **Department** | **:** | **Teknik Informatika FTIF-ITS** |
| **First Advisor** | **:** | **Dr.Eng. Nanik Suciati, S.Kom, M.Kom** |
| **Second Advisor** | **:** | **Dr.Eng. Chastine Fatichah, S.Kom, M.Kom** |

# *Abstract*

*Iris recognition is a pattern recognition technique used in biometric applications. This biometric technique generates a unique random pattern statistically. In other words, the texture of the iris is the part that is unique to each person.*

*In this final project implements iris recognition using Wavelet and Log-Gabor Filter as feature extraction. Then continued with Support Vector Machines and Hamming Distance as the classification method. Before feature extraction, first performed preprocessing for part identification iris of the eye and the process of normalization.*

*The test result with CASIA eye dataset using SVM classification and Hamming distance proved to be quite effective in the process of iris recognition. In the process using SVM classification method only achieved accuracy 94.44%, with a Hamming distance classification accuracy alone achieved 94.14%, and achieved 98.45% by using a combination of SVM and Hamming distance*.

***Keywords: Iris Recognition, Log-Gabor Filter, Hamming Distance, Wavelet, Support Vector Machines.***

***[Halaman ini sengaja dikosongkan]***

# KATA PENGANTAR



Segala puji syukur ditujukan kepada Allah SWT yang telah melimpahkan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul

**IMPLEMENTASI PENGENALAN IRIS MATA MENGGUNAKAN METODE SUPPORT VECTOR MACHINES DAN HAMMING DISTANCE**

Tugas Akhir ini merupakan salah satu syarat untuk memperoleh gelar Sarjana Komputer di Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember Surabaya.

Penulis ingin menyampaikan terima kasih yang sebesar-besarnya atas dukungan dan semangat yang diberikan dan membantu penulis baik secara langsung ataupun tidak dalam menyelesaikan Tugas Akhir ini. Penulis ingin mengucapkan terima kasih kepada

1. Ibu, bapak, dan kakak-kakak tercinta yang telah membantu doa, moral dan material selama penulis belajar di Teknik Informatika ITS.
2. Bapak Dr. Darlis Herumurti, S.Kom., M.Kom., selaku ketua jurusan, Bapak Radityo Anggoro, S.Kom., M.Sc. selaku Koordinator Tugas Akhir, dan segenap dosen Teknik Informatika yang telah memberikan ilmunya selama penulis berkuliah di Teknik Informatika ITS.
3. Ibu Dr.Eng. Nanik Suciati, S.Kom, M.Kom selaku Dosen Pembimbing I Tugas Akhir yang telah memberikan bimbingan dan dukungan selama penulis menyelesaikan Tugas Akhir sampai selesai.
4. Ibu Dr.Eng. Chastine Fatichah, S.Kom, M.Kom selaku pembimbing II Tugas Akhir yang telah memberikan bimbingan dan dukungan selama penulis menyelesaikan Tugas Akhir sampai selesai.
5. Para penulis artikel ilmiah yang digunakan pada Tugas akhir ini, pendiri Google, pendiri Mathworks, serta berbagai referensi lain, yang telah membantu mempermudah penyelesaian Tugas Akhir ini.
6. Bu Eva di Ruang Baca dan seluruh Staf dan karyawan Teknik Informatika yang telah memberikan bantuan selama penulis kuliah di Teknik Informatika.
7. Kawan-kawan angkatan 2012 yang menemani, membantu, dan memotivasi selama penulis menjalankan seluruh tugas perkuliahan hingga penulis sampai pada tahap pengerjaan Tugas Akhir ini.
8. Serta semua pihak yang yang telah turut membantu penulis dalam menyelesaikan Tugas Akhir ini.

Penulis menyadari bahwa Tugas Akhir ini masih memiliki banyak kekurangan. Sehingga dengan kerendahan hati, penulis mengharapkan kritik dan saran dari pembaca untuk perbaikan ke depan. Semoga Tugas Akhir ini dapat memberikan Manfaat yang sebesar besarnya.

Surabaya, Juni 2016

Penulis

# DAFTAR ISI

[LEMBAR PENGESAHAN v](#_Toc441189682)

[*Abstrak* vii](#_Toc441189683)

[*Abstract* ix](#_Toc441189684)

[KATA PENGANTAR xi](#_Toc441189685)

[DAFTAR ISI xiii](#_Toc441189686)

[DAFTAR GAMBAR xvii](#_Toc441189687)

[DAFTAR TABEL xix](#_Toc441189688)

[DAFTAR KODE SUMBER xxi](#_Toc441189689)

[BAB I PENDAHULUAN 1](#_Toc441189690)

[1.1 Latar Belakang 1](#_Toc441189691)

[1.2 Rumusan Masalah 2](#_Toc441189692)

[1.3 Batasan Masalah 2](#_Toc441189693)

[1.4 Tujuan 2](#_Toc441189694)

[1.5 Manfaat 3](#_Toc441189695)

[1.6 Metodologi 3](#_Toc441189696)

[BAB II TINJAUAN PUSTAKA 5](#_Toc441189697)

[2.1 Gaussian Mixture Models 5](#_Toc441189698)

[2.2 Gaussian Pyramid 7](#_Toc441189699)

[2.3 Probabilitas Distribusi Gaussian 8](#_Toc441189700)

[2.4 Region Growing 9](#_Toc441189701)

[2.5 Daubachies 4 Wavelet 10](#_Toc441189702)

[2.6 Normalisasi Min-Max 13](#_Toc441189703)

[2.7 Support Vector Machines 14](#_Toc441189704)

[BAB III DESAIN PERANGKAT LUNAK 19](#_Toc441189705)

[3.1 Data 19](#_Toc441189706)

[3.1.1 Data Masukan 19](#_Toc441189707)

[3.1.2 Data Pembelajaran 19](#_Toc441189708)

[3.1.3 Data Keluaran 20](#_Toc441189709)

[3.2 Desain Sistem Secara Umum 21](#_Toc441189710)

[3.3 Preprocessing 22](#_Toc441189711)

[3.3.1 Reduksi Size Frame 23](#_Toc441189712)

[3.3.2 Deteksi Gerak 25](#_Toc441189713)

[3.3.3 Deteksi Warna Piksel 25](#_Toc441189714)

[3.3.4 Region Growing 27](#_Toc441189715)

[3.3.5 Perhitungan Luasan Region 28](#_Toc441189716)

[3.4 Verifikasi 28](#_Toc441189717)

[3.4.1 Ekstraksi Fitur dengan Wavelet 30](#_Toc441189718)

[3.4.2 Klasifikasi 31](#_Toc441189719)

[3.5 Menandai Region Api 32](#_Toc441189720)

[BAB IV IMPLEMENTASI 35](#_Toc441189721)

[4.1 Lingkungan Implementasi 35](#_Toc441189722)

[4.2 Implementasi 35](#_Toc441189723)

[4.2.1 Implementasi Tahap Reduksi Size Frame 35](#_Toc441189724)

[4.2.2 Implementasi Tahap Deteksi Gerak 36](#_Toc441189725)

[4.2.3 Implementasi Tahap Deteksi Warna Piksel 37](#_Toc441189726)

[4.2.4 Implementasi Tahap Region Growing 41](#_Toc441189727)

[4.2.5 Implementasi Tahap Perhitungan Luasan Region 44](#_Toc441189728)

[4.2.6 Implementasi Tahap Ekstraksi Fitur dengan Wavelet 45](#_Toc441189729)

[4.2.7 Implementasi Tahap Klasifikasi 45](#_Toc441189730)

[4.2.8 Implementasi Tahap Menandai Region Api 47](#_Toc441189731)

[BAB V UJI COBA DAN EVALUASI 49](#_Toc441189732)

[5.1 Lingkungan Uji Coba 49](#_Toc441189733)

[5.2 Data Uji Coba 49](#_Toc441189734)

[5.3 Alur Uji Coba 49](#_Toc441189735)

[5.3.1 Preprocessing 50](#_Toc441189736)

[5.3.2 Verifikasi 52](#_Toc441189737)

[5.4 Skenario Uji Coba 54](#_Toc441189738)

[5.4.1 Skenario Uji Coba 1 55](#_Toc441189739)

[5.4.2 Skenario Uji Coba 2 56](#_Toc441189740)

[5.4.3 Skenario Uji Coba 3 56](#_Toc441189741)

[5.4.4 Skenario Uji Coba 4 57](#_Toc441189742)

[5.4.5 Skenario Uji Coba 5 57](#_Toc441189743)

[5.4.6 Skenario Uji Coba 6 58](#_Toc441189744)

[5.5 Analisis Hasil Uji Coba 58](#_Toc441189745)

[BAB VI KESIMPULAN DAN SARAN 61](#_Toc441189746)

[6.1 Kesimpulan 61](#_Toc441189747)

[6.2 Saran 62](#_Toc441189748)

[DAFTAR PUSTAKA 63](#_Toc441189749)

[LAMPIRAN A 65](#_Toc441189750)

[BIODATA PENULIS 107](#_Toc441189751)

***[Halaman ini sengaja dikosongkan]***

# DAFTAR GAMBAR

[Gambar 2.1 Ilustrasi *Gaussian Pyramid* [3] 7](#_Toc441189752)

[Gambar 2.2 *Seed* Awal *Region Growing* 9](#_Toc441189753)

[Gambar 2.3 Piksel yang diamati dan *Region* 10](#_Toc441189754)

[Gambar 2.4 Ilustrasi *Low Pass Filter* 12](#_Toc441189755)

[Gambar 2.5 Ilustrasi *Filter* yang dilakukan 12](#_Toc441189756)

[Gambar 2.6 Citra Masukan 13](#_Toc441189757)

[Gambar 2.7 Citra Keluaran Hasil *Wavelet* 13](#_Toc441189758)

[Gambar 2.8 Ilustrasi *Support Vector Machines* 15](#_Toc441189759)

[Gambar 3.1 Contoh Data Masukan 20](#_Toc441189760)

[Gambar 3.2 Contoh Data Keluaran 20](#_Toc441189761)

[Gambar 3.3 Diagram Alir Rancangan Perangkat Lunak Secara Umum 21](#_Toc441189762)

[Gambar 3.4 Digram Alir Reduksi *Size Frame* 23](#_Toc441189763)

[Gambar 3.5 Diagram Alir *Preprocessing* 24](#_Toc441189764)

[Gambar 3.6 Contoh Deteksi Gerak 25](#_Toc441189765)

[Gambar 3.7 Contoh *Dataset* Gambar Api 26](#_Toc441189766)

[Gambar 3.8 *Pseudocode* Deteksi Warna Piksel 26](#_Toc441189767)

[Gambar 3.9 *Pseudocode Regon Growing* 27](#_Toc441189768)

[Gambar 3.10 Diagram Alir Perhitungan Luasan *Region* 29](#_Toc441189769)

[Gambar 3.11 Diagram Alir Verifikasi 30](#_Toc441189770)

[Gambar 3.12 Diagram Alir Ekstraksi Fitur 32](#_Toc441189771)

[Gambar 3.13 *Pseudocode* fungsi menandai *region* 33](#_Toc441189772)

[Gambar 3.14 Menandai *region* api 33](#_Toc441189773)

[Gambar 5.1 Contoh Video Kejadian 50](#_Toc441189774)

[Gambar 5.2 Tahap *Preprocessing* 52](#_Toc441189775)

[Gambar 5.3 Tahap Verifikasi 54](#_Toc441189776)

***[Halaman ini sengaja dikosongkan]***

# DAFTAR TABEL

[Tabel 4.1 Lingkungan Perancangan Perangkat Lunak 35](#_Toc441189777)

[Tabel 5.1 Hasil Uji Coba 1 55](#_Toc441189778)

[Tabel 5.2 Hasil Uji Coba 2 56](#_Toc441189779)

[Tabel 5.3 Hasil Uji Coba 3 57](#_Toc441189780)

[Tabel 5.4 Hasil Uji Coba 4 57](#_Toc441189781)

[Tabel 5.5 Hasil Uji Coba 5 58](#_Toc441189782)

[Tabel 5.6 Hasil Uji Coba 6 58](#_Toc441189783)

[Tabel A.1 Hasil Uji Coba Menggunakan Parameter *Threshold*  = 10-7, = 5 dan kernel RBF 65](#_Toc441189784)

[Tabel A.2 Hasil Uji Coba Menggunakan Parameter *Threshold*  = 10-8, = 5 dan kernel RBF 68](#_Toc441189785)

[Tabel A.3 Hasil Uji Coba Menggunakan Parameter *Threshold*  = 5x10-9, = 5 dan kernel RBF 71](#_Toc441189786)

[Tabel A.4 Hasil Uji Coba Menggunakan Parameter *Threshold*  = 10-9, = 5 dan kernel RBF 74](#_Toc441189787)

[Tabel A.5 Hasil Uji Coba Menggunakan Parameter *Threshold*  = 5x10-9, = 1 dan kernel RBF 77](#_Toc441189788)

[Tabel A.6 Hasil Uji Coba Menggunakan Parameter *Threshold*  = 5x10-9, = 3.5 dan kernel RBF 80](#_Toc441189789)

[Tabel A.7 Hasil Uji Coba Menggunakan Parameter *Threshold*  = 5x10-9, = 7 dan kernel RBF 83](#_Toc441189790)

[Tabel A.8 Hasil Uji Coba Menggunakan Parameter *Threshold*  = 5x10-9, = 5 dan kernel *Polynomial 2* 86](#_Toc441189791)

[Tabel A.9 Hasil Uji Coba Menggunakan Parameter *Threshold*  = 5x10-9, = 5 dan kernel *Polynomial 3* 89](#_Toc441189792)

[Tabel A.10 Hasil Uji Coba Menggunakan Parameter konstanta *region* = 5%, *Threshold*  = 5x10-9, = 5 dan kernel RBF 92](#_Toc441189793)

[Tabel A.11 Hasil Uji Coba Menggunakan Parameter konstanta *region* = 10%, *Threshold*  = 5x10-9, = 5 dan kernel RBF 95](#_Toc441189794)

[Tabel A.12 Hasil Uji Coba Menggunakan Parameter *size frame* = 240 x 320, *Threshold*  = 5x10-9, = 5 dan kernel RBF 98](#_Toc441189795)

[Tabel A.13 Hasil Uji Coba Menggunakan Parameter *size frame* = 60 x 80, *Threshold*  = 5x10-9, = 5 dan kernel RBF 101](#_Toc441189796)

[Tabel A.14 Hasil Uji Coba Tanpa Menggunakan *region growing* dan perhitungan luasan *region* 104](#_Toc441189797)

# DAFTAR KODE SUMBER

[Kode Sumber 4.1 Implementasi Tahap Reduksi *Size Frame* 36](#_Toc441189798)

[Kode Sumber 4.2 Penggunaan Fungsi pyrDown() 36](#_Toc441189799)

[Kode Sumber 4.3 Implementasi Tahap Deteksi Gerak 36](#_Toc441189800)

[Kode Sumber 4.4 Implementasi Penyimpanan Piksel 37](#_Toc441189801)

[Kode Sumber 4.5 Implementasi Menghitung Nilai Standar Deviasi dan Rata-Rata Setiap *Channel* 38](#_Toc441189802)

[Kode Sumber 4.6 *Generate list* piksel api 39](#_Toc441189803)

[Kode Sumber 4.7 Fungsi Menghitung Nilai Probabilitas Distribusi Gaussian 40](#_Toc441189804)

[Kode Sumber 4.8 Mendapatkan Threshold 40](#_Toc441189805)

[Kode Sumber 4.9 Membaca *List* Piksel Api 40](#_Toc441189806)

[Kode Sumber 4.10 Implementasi Tahap Deteksi Warna Piksel 41](#_Toc441189807)

[Kode Sumber 4.11 Implementasi Tahap *Region Growing* 42](#_Toc441189808)

[Kode Sumber 4.12 Implementasi *Growing* 43](#_Toc441189809)

[Kode Sumber 4.13 Implementasi Tahap *Clock Wise* 43](#_Toc441189810)

[Kode Sumber 4.14 Implementasi Tahap Variasi Warna *Region* 44](#_Toc441189811)

[Kode Sumber 4.15 Implementasi Memasukan Nilai *Wavelet* Kedalam *List* 45](#_Toc441189812)

[Kode Sumber 4.16 Pemanggilan Fungsi *Wavelet* 45](#_Toc441189813)

[Kode Sumber 4.17 *Training* Klasifikasi 46](#_Toc441189814)

[Kode Sumber 4.18 Implementasi Tahap Klasifikasi 47](#_Toc441189815)

[Kode Sumber 4.19 Implementasi Tahap Menandai Region Api 48](#_Toc441189816)

***[Halaman ini sengaja dikosongkan]***

# PENDAHULUAN

Pada bab ini dibahas mengenai latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat, metedologi, dan sistematika laporan tugas akhir. Diharapkan dari penjelasan dalam bab ini gambaran Tugas Akhir secara umum dapat dipahami.

## Latar Belakang

Teknologi Biometrik adalah metode untuk mengenali seseorang melalui ciri-ciri fisik, karakter, dan kebiasaan. Biasanya ciri-ciri fisik khas yang dijadikan indikator untuk mengenali seseorang adalah wajah, sidik jari, telapak tangan, retina, atau iris mata. Dapat pula mengenali seseorang melalui ciri-ciri lainnya seperti cara berjalan, tanda tangan, atau suara. Dengan menggunakan macam-macam ciri yang unik tersebut, dapat dijadikan solusi untuk mengidentifikasi dan mengenali seseorang.

Iris atau selaput pelangi adalah daerah berbentuk gelang pada mata yang dibatasi oleh pupil dan sklera (bagian putih dari mata). Dalam penerapannya pada teknik biometrik, iris memiliki pola yang sangat unik, berbeda pada tiap individu dan pola itu akan tetap stabil. Sebelum kelahiran, degenerasi terjadi sehingga menghasilkan pembukaan pupil dan acak, serta pola-pola unik dari iris. Walaupun genetik serupa, seseorang yang memiliki struktur iris yang unik dan berbeda, dapat memungkinkan untuk digunakan untuk tujuan pengenalan. Iris mata juga mudah untuk dicitrakan pada jarak yang sesuai dari subjek dengan penggunaan alat yang ada, misalnya kamera. Atas dasar inilah iris mata dapat dijadikan dasar bagi pengenalan biometrik [1].

Metode pengklasifikasian yang dapat digunakan adalah *Support Vector Machines*(SVM). Pengklasifikasian dengan metode ini terdiri dari dua modul utama, yaitu pembuatan model dan klasifikasi data. Model SVM yang dibuat dimaksudkan untuk memisahkan satu kategori dengan satu kategori yang lain dengan menggunakan sebuah bidang *hyperlane*. Selanjutnya klasifikasi akan dilakukan dengan menggunakan model yang dibuat, membuatnya mampu memberikan kinerja yang lebih tinggi dalam hal akurasi dari algoritma klasifikasi lainnya [2].

Salah satu metode lainnya dalam digunakan dalam pengklasifikasian adalah *Hamming distance*. Metode ini merepresentasikan pola iris kedalam bentuk biner, karena lebih mudah untuk menentukan perbedaan antara dua kode biner daripada bilangan biasa. Metode ini pada dasarnya adalah fungsi eksklusif OR (XOR) antara dua pola bit.

Tugas Akhir ini mengimplementasikan pengklasifikasi tekstur iris mata dengan menggunakan masing-masing dua metode klasifiskasi tersebut dan kombinasi dari keduanya. Sebelum dilakukan klasifikasi, terlebih dahulu dilakukan praproses untuk identifikasi bagian iris pada citra mata. Selanjutnya dilakukan normalisasi iris pada *region of interest*, serta ekstraksi fitur untuk menghasilkan vektor fitur yang digunakan untuk proses klasifikasi.

## Rumusan Masalah

Rumusan masalah yang diangkat dalam Tugas Akhir ini adalah sebagai berikut:

1. Bagaimana penggunaan *Support Vector Machines* pada pengenalan tekstur iris mata.
2. Bagaimana penggunaan *Hamming distance* pada pengenalan tekstur iris mata.
3. Menyusun uji coba pengklasifikasi tekstur iris mata menggunakan *kernel* pada SVM.
4. Bagaimana melakukan identifikasi bagian iris pada citra mata.

## Batasan Masalah

Permasalahan yang dibahas dalam tugas akhir memiliki beberapa batasan, yakni sebagai berikut.

1. Database yang digunakan adalah data citra mata dari basis data *Chinese Academy of Sciences Institute of Automation* (CASIA).
2. Implementasi menggunakan perangkat lunak MATLAB 8.3 dan *Image Processing* *Toolbox* didalamnya.
3. Data yang digunakan untuk data latih dan data uji berupa citra mata dari database CASIA.

## Tujuan

Tujuan dari pembuatan tugas akhir ini adalah untuk membuat sebuah implementasi perangkat lunak yang dapat melakukan pengenalan iris mata.

## Manfaat

Pengerjaan tugas akhir ini dilakukan dengan harapan bisa memberikan kontribusi pada teknologi biometrik yang dapat melakukan pengenalan melalui iris mata. Harapannya, dengan pembuatan perangkat lunak ini, akan dapat melakukan pendeteksian identitas seseorang melalui iris mata. Sehingga dapat melakukan proses verifikasi menjadi lebih cepat.

## Metodologi

Metodologi yang dipakai pada pengerjaan Tugas Akhir ini adalah sebagai berikut:

1. Penyusunan Proposal Tugas Akhir

Tahap awal yang dilakukan dalam pengerjaan Tugas Akhir ini adalah penyusunan proposal Tugas Akhir. Di dalam proposal diajukan suatu gagasan untuk melakukan implementasi pengklasifikasian pada iris mata pada metode Support Vector Machines dan Hamming distance.

1. Studi Literatur

Pada tahap ini dilakukan pencarian, pengumpulan, pembelajaran dan pemahaman informasi dan literatur yang diperlukan untuk pembuatan implementasi pada tahap praproses citra mata, sehingga menghasilkan daerah iris, ekstraksi fitur serta metode-metode klasifikasi. Dasar informasi yang diperlukan pada pembuatan implementasi ini diantaranya mengenai citra secara umum dan algoritma tahap praproses. Serta metode pengklasifikasi pada iris mata. Literatur yang digunakan meliputi: buku referensi, jurnal, dan dokumentasi internet.

1. Implementasi perangkat lunak

Pada tahap ini dilakukan implementasi perangkat lunak sesuai dengan rancangan perangkat lunak yang dibuat pada pengerjaan Tugas Akhir.

1. Uji coba dan Evaluasi

Pada tahap ini dilakukan uji coba terhadap perangkat lunak yang telah dibuat untuk mengetahui kemampuan algoritma yang dipakai, mengamati kinerja sistem, serta mengidentifikasi kendala yang mungkin timbul. Parameter yang diujicobakan adalah fungsi kernel pada metode klasifikasi *support vector machines*.

1. Penyusunan Laporan Tugas Akhir

Pada tahap ini dilakukan penyusunan laporan pengerjaan Tugas Akhir yang berisi dasar teori, dokumentasi dari perangkat lunak, dan hasil yang diperoleh selama pengerjaan Tugas Akhir.

## Sistematika Penulisan

Buku Tugas Akhir ini bertujuan untuk mendapatkan gambaran dari pengerjaan Tugas Akhir ini. Selain itu, diharapkan dapat berguna untuk pembaca yang tertarik untuk melakukan pengembangan lebih lanjut.

Secara garis besar, buku Tugas Akhir terdiri atas beberapa bagian seperti berikut ini:

1. **Pendahuluan**

Bab ini berisi latar belakang masalah, tujuan dan manfaat pembuatan Tugas Akhir, permasalahan, batasan masalah, metodologi yang digunakan, dan sistematika penyusunan Tugas Akhir.

1. **Dasar Teori**

Bab ini membahas beberapa teori penunjang yang berhubungan dengan pokok pembahasan dan mendasari pembuatan Tugas Akhir ini.

1. **Perancangan Perangkat Lunak**

Bab ini berisi tentang desain sistem yang disajikan dalam bentuk *flowchart*.

1. **Implementasi**

Bab ini berisi implementasi dari perancangan yang telah dibuat sebelumnya. Penjelasan berupa code yang digunakan untuk proses implementasi.

1. **Uji Coba dan Evaluasi**

Bab ini menjelaskan kemampuan perangkat lunak dengan melakukan pengujian kebenaran dan pengujian kinerja dari sistem yang telah dibuat*.*

1. **Kesimpulan dan Saran**

Bab ini berisi kesimpulan dari hasil pengujian yang dilakukan. Bab ini membahas saran-saran untuk pengembangan sistem lebih lanjut.

***[Halaman ini sengaja dikosongkan]***

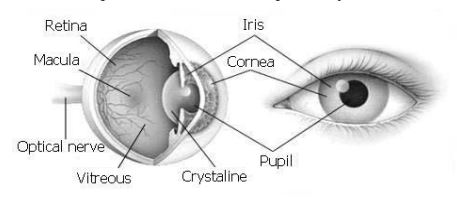
# TINJAUAN PUSTAKA

Pada bab ini dibahas mengenai teori-teori yang menjadi dasar dari pembuatan Tugas Akhir.

## Anatomi Mata

Bagian organ mata yang akan diteliti dalam tugas akhir ini adalah iris atau selaput pelangi. Bagian-bagian mata seperti yang tampak pada gambar 2.1 dijelaskan sebagai berikut [3]:

1. Retina adalah bagian yang berfungsi untuk menangkap bayangan benda
2. Iris adalah bagian yang mempunyai pigmen untuk memberi warna pada mata. Bagian inilah yang akan diteliti pada Tugas Akhir ini
3. Kornea berfungsi untuk meneruskan cahaya yang masuk ke mata
4. Pupil merupakan bagian mata yang berbentuk lingkaran yang mengatur banyaknya cahaya yang masuk ke mata. Bagian ini akan diteliti juga untuk mendapatkan bagian iris
5. *Vitreous* adalah gel yang mengisi ruang antara lensa mata dan retina
6. *Optical Nerve* adalah saraf mata yang meneruskan atau membelokan sinar menuju ke otak.



Gambar 2.1 Anatomi Mata

## Sistem Pengenalan Iris

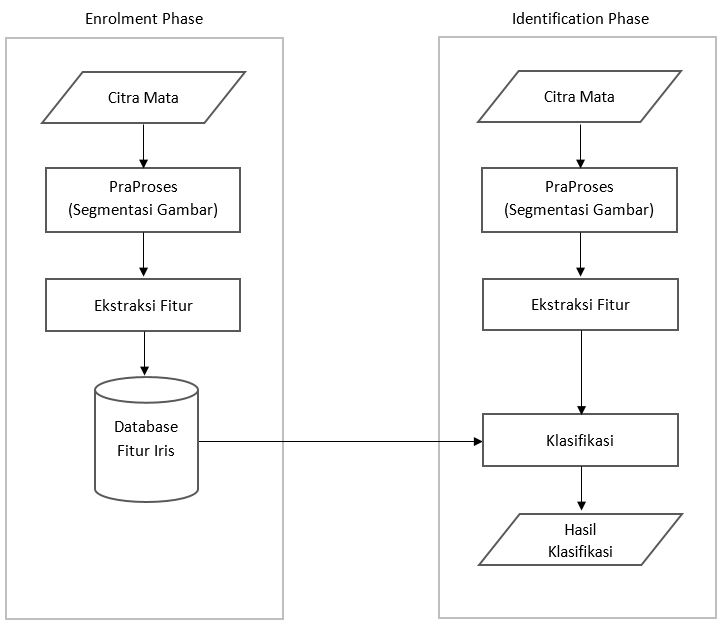
Pengenalan iris adalah suatu proses untuk mengenal seseorang dengan menganalisa pola acak dari iris. Dimana pengenalan iris bertujuan untuk mengenali suatu objek dengan cara mengekstraksi informasi yang terdapat dalam suatu citra tersebut.

Dalam pengaplikasiannya, sudah banyak metode yang digunakan. Pada proses identifikasi iris sampai normalisasi relatif sama. Metode dengan menggunakan *wavelet* [1], *Laplacian of Gaussian Filter* [4], dan berbagai macam metode lainnya. Metode-metode tersebut digumakan pada proses ekstraksi fitur. Sedangkan pada proses pengenalan, ada pula klasifikasi *Support Vector Machines* [5], *Hamming distance* [6], maupun kombinasi antara metode-metode tersebut [6].

Secara garis besar, proses-proses tersebut dikelompokkan pada empat proses utama yaitu:

1. Akuisisi citra
2. Praproses citra (identifikasi objek iris dan normalisasi)
3. Ekstraksi fitur iris
4. Pencocokan iris.

Akuisisi citra adalah proses mendapatkan citra mata yang akan dilakukan pengenalan. Citra mata yang digunakan pada tugas akhir ini yaitu citra mata CASIA [7]. Proses selanjutnya adalah praproses citra atau melakukan pengolahan awal pada citra mata yang bertujuan untuk mengambil karakteristik tekstur iris mata dan melakukan normalisasi iris untuk mengatasi kondisi citra mata yang bervariasi pada setiap orang. Tekstur iris yang sudah terambil akan diekstraksi citi atau fitur iris. Setiap fitur akan disimpan didalam sebuah database yang nantinya akan dibandingkan dengan fitur masukan dalam proses pencocokan iris. Proses tersebut mencocokan fitur masukan dengan seluruh fitur yang terdapat didalam database dan menghasikan bobot atau tingkat kemiripan. Nilai tingkat kemiripan yang paling tinggi adalah citra yang dikenali paling mirip dengan citra masukan. Gambar 2.2 menjelaskan pengenalan iris secara umum.

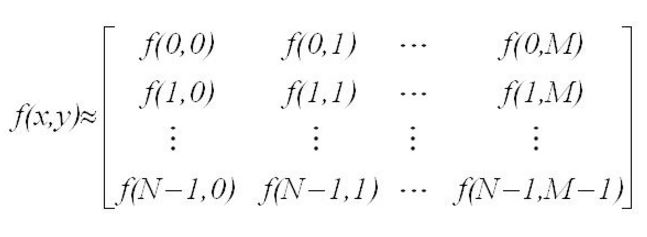


Gambar 2.2 Sistem pengenalan iris secara umum

## Representasi Citra Digital

Citra dapat didefinisikan sebagai f(x,y), yaitu fungsi dua dimensi, dengan x dan y menyatakan koordinat spasial dan nilai f pada sembarang titik (x,y) disebut dengan intensitas atau tingkat keabuan dari citra pada koordinat tersebut. Jika nilai x,y dan nilai f adalah *finite*, bernilai diskrit atau terbatas, maka dapat disebut bahwa citra tersebut merupakan citra digital [8].

Sebuah citra digital f(x,y) dapat direpresentasikan sebagai sebuah matriks yang indeks baris dan kolomnya mengidentifikasikan sebuah titik pada citra dan nilai dari elemen matriks yang bersangkutan merupakan tingkat warna pada titik tersebut, seperti yang ditunjukan pada Gambar 2.3 Elemen tersebut disebut elemen citra, elemen gambar, pixels, atau pels. *Picture elements* atau pixel dapat didefinisikan sebagai elemen terkecil dari sebuah citra digital yang menentukan resolusi citra tersebut.



Gambar 2.3 Representasi citra digital

## Gaussian Mixture Models

*Gaussian mixture models* adalah salah satu metode yang digunakan untuk mendeteksi gerak. Didefinisikan piksel pada *frame* yang sedang diproses sebagai dengan nilai sebelumnya dari *frame* 1 sampai *frame* . Model dapat dilihat pada Persamaan (2.1).

|  |  |
| --- | --- |
|  | (2.1) |
|  |  |

Probabilitas piksel dapat didefinisikan menggunakan Persamaan (2.2).

|  |  |
| --- | --- |
|  | (2.2) |

Dimana adalah bilangan gaussian (antara 3 hingga 5), adalah *covariance* yang didapatkan dari , adalah *weight*, adalah *gaussian probability density function*. *Gaussian probability density function* dapat dilihat pada Persamaan (2.3).

|  |  |
| --- | --- |
|  | (2.3) |

Distribusi diurutkan secara *descending* berdasarkan nilai , nilai distribusi akan digunakan sebagai model *background*. Nilai didapatkan menggunakan Persamaan (2.4).

|  |  |
| --- | --- |
|  | (2.4) |

Dimana nilai adalah nilai minimum dari *background model*. Selanjutnya dilakukan *background subtraction* dengan menghitung nilai piksel dengan distribusi . Jika nilai piksel lebih dari 2.5 dari nilai standar deviasi distribusi tersebut (*match*), maka komponen distribusi tersebut dilakukan *update* dan piksel tersebut dianggap sebagai *foreground*. Untuk *update* nilai komponen gaussian dilakukan dengan Persamaan (2.5)-(2.9).

|  |  |
| --- | --- |
|  | (2.5) |
|  | (2.6) |
|  | (2.7) |
|  | (2.8) |
|  | (2.9) |

Jika dari semua distribusi piksel yang dicek tidak memenuhi syarat (*unmatch*) maka komponen gaussian yang terakhir akan dilakukan *update*. *Update* dilakukan dengan mengubah rata-rata () dengan nilai piksel yang sedang dicek dan mengubah nilai variasi () dengan nilai tinggi dan nilai *weight* () dengan nilai yang rendah.

## Gaussian Pyramid

*Gaussian pyramid* digunakan untuk melakukan reduksi resolusi citra. Citra yang tereduksi resolusinya akan berkurang menjadi seperempat dari resolusi awal. Hal ini dilakukan untuk mempercepat proses perhitungan. Ilustrasi *gaussian pyramid* dapat dilihat pada Gambar 2.1.



Gambar 2.4 Ilustrasi *Gaussian Pyramid*

*Gaussian pyramid* dilakukan dengan dua operasi, yaitu *smoothing* dan *down sampling*. *Smoothing* dilakukan dengan menggunakan filter 5x5. *Smoothing* dilakukan dengan menggunakan Persamaan (2.10).

|  |  |
| --- | --- |
|  | (2.10) |

Dimana adalah filter 5x5. Detail persamaan setiap piksel dilakukan menggunakan Persamaan (2.11).

|  |  |
| --- | --- |
|  | (2.11) |

Setelah mendapatkan citra yang telah di *smoothing*, langkah selanjutnya adalah melakukan *down sampling*. *Down sampling* dilakukan untuk mengubah resolusi citra asli menjadi resolusi yang lebih kecil. *Down sampling* dilakukan dengan Persamaan (2.12).

|  |  |
| --- | --- |
|  | (2.12) |

Untuk melakukan reduksi, dilakukan menggunakan dua proses tersebut, yaitu *smoothing* dan *down sampling*. Perhitungan dapat dilakukan dengan menggabungkan ke dua persamaan tersebut menjadi Persamaan (2.13).

|  |  |
| --- | --- |
|  | (2.13) |

## Probabilitas Distribusi Gaussian

Probabilitas distribusi gaussian adalah sebuah metode untuk menghitung probabilitas dari suatu data. Probabilitas dilakukan dengan menghitung nilai rata-rata dan standar deviasi dari suatu data. Persamaan umum probabilitas distribusi gausian dapat dilihat pada Persamaan (2.14).

|  |  |
| --- | --- |
|  | (2.14) |

Pada kasus tugas akhir ini, probabilitas gaussian digunakan untuk mendapatkan probabilitas warna piksel. Setiap piksel dihitung nilai probabilitas R,G,B. Setelah dilakukan perhitungan probabilitas R,G,B selanjunya probabilitas tersebut dikalikan sehingga mendapatkan nilai probabilitas piksel. Perhitungan probabilitas setiap piksel dapat dilihat pada Persamaan (2.15).

|  |  |
| --- | --- |
|  | (2.15) |

## Region Growing

*Region Growing* adalah metode untuk melakukan segmentasi citra. Pendekatan dasar adalah dengan memulai titik yang sudah diinisialisasi (*seed*) dan dari titik tersebut dilakukan menumbuhkan daerah dengan cara menambahkan tetangga piksel dari *seed* yang mempunyai kesamaan dengan *seed*.

Untuk metode ini, dibutuhkan aturan yang mengatur mekanisme tumbuhnya *seed* dan suatu aturan lain yang menguji kehomogenan dari *region* setelah satu tahap tumbuh selesai. Pertumbuhan *region* dimulai dari *seed* awal dengan menambahkan tetangga piksel (menggunakan 8-tetangga) yang serupa untuk menumbuhkan *region.* Ketika suatu pertumbuhan *region* selesai, langkah selanjutnya adalah memilih *seed* baru dan melakukan *region growing* kembali. Proses tersebut dilakukan hingga semua piksel berhasil dikelompokkan dalam beberapa *region*. Ilustrasi *region growing* dapat dilihat pada Gambar 2.2 dan Gambar 2.3.



Gambar 2.5 *Seed* Awal *Region Growing*



Gambar 2.6 Piksel yang diamati dan *Region*

## Daubachies 4 Wavelet

*Wavelet* adalah fungsi matematika yang membagi data menjadi beberapa komponen frekuensi yang berbeda-beda dan menganalisis setiap komponen tersebut dengan menggunakan resolusi yang sesuai dengan skalanya. Transformasi *wavelet* mendekomposisi signal kedalam *frequency bands* dengan memproyeksikan signal kedalam set fungsi dasar. Pada citra, transformasi *wavelet* adalah transformasi yang melakukan filter terhadap suatu masukan. Filter yang digunakan dalam *wavelet* adalah *high pass filter* dan *low pass filter*. Pada daubachies 4, terdapat dua koefisien yang digunakan untuk melakukan *filter*, yaitu *scaling function coefficient* dan *wavelet function coefficient*. Dimana *scaling function coeficient* adalah koefisien yang digunakan dalam melakukan *low pass filter*, sedangkan *wavelet function coeficient* digunakan dalam melakukan *high pass filter* . Persamaan *scaling function coeficient* dapat dilihat pada Persamaan (2.16)*.*

|  |  |
| --- | --- |
|  | (2.16) |

*Wavelet function coeficient* dapat dilihat pada Persamaan (2.17).

|  |  |
| --- | --- |
|  | (2.17) |

Tahapan dari proses *filter* adalah melakukan *filter* terhadap baris citra terlebih dahulu. Dilakukan *high pass filter* dan *low pass filter*, dimana hasil dari proses ini adalah dua citra, yaitu citra *high pass* dan citra *low pass*. Ilustrasi tahap *filter* dapat dilihat pada Gambar 2.4.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  | **.** |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |

Gambar 2.7 Ilustrasi *Low Pass Filter*

Citra yang dilakukan *filter* akan ter-reduksi menjadi setengah. Dilanjutkan dengan melakukan *filter* terhadap kolom citra, dilakukan *high pass filter* dan *low pass filter*. Hasil dari proses ini adalah empat citra dengan satu citra approximasi, dan tiga citra detail. Citra detail yang didapat adalah detail horizontal, vertikal dan diagonal. Gambar 2.5 Adalah lustrasi dari proses *filter* yang dilakukan.

F:\KULIAH\TA\bukuTA\source\wavelt.png

Gambar 2.8 Ilustrasi *Filter* yang dilakukan

Ilustrasi citra yang masukan dan keluar pada proses ini dapat dilihat pada Gambar 2.6 dan Gambar 2.7.



Gambar 2.9 Citra Masukan



Gambar 2.10 Citra Keluaran Hasil *Wavelet*

## Normalisasi Min-Max

Normalisasi adalah sebuah proses untuk mengubah suatu data ke dalam rentang nilai tertentu. Tujuannya adalah untuk menghindari persebaran data yang terlalu jauh sehingga sebuah variabel tidak mendominasi terhadap variabel lain. Salah satu jenis normalisasi adalah normalisasi skala. Pada normalisasi, skala rentang yang umum digunakan yaitu 0 hingga 1. Rumus umum skala adalah sebagai dapat dilihat pada Persamaan (2.18).

|  |  |
| --- | --- |
|  | (2.18) |

Variabel adalah nilai yang akan dinormalisasi. Variabel dan adalah nilai minimum dan maximum pada nilai-nilai atribut dimana berada. Variabel dan adalah nilai minimum dan maximum yang diinginkan. Apabila rentang yang digunakan adalah 0 hingga 1 maka rumus diatas menjadi Persamaan (2.19).

|  |  |
| --- | --- |
|  | (2.19) |

## Support Vector Machines

*Support vector machines* adalah metode klasifikasi yang mengklasifikasikan dua kelas, yaitu kelas +1 dan -1. Pada metode klasifikasi *support vector machines*, dibentuk suatu *hyperplane*. *Hyperplane* adalah garis pemisah yang memisahkan dua kelas yang berbeda. Dalam metode *support vector machines* dikenal istilah *margin*. *Margin* adalah jarak antara kelas +1 dengan kelas -1 yang paling dekat, pada *support vector machines* dicari *margin* terpanjang antara dua kelas tersebut. Ilustrasi *support vector machines* dapat dilihat pada Gambar 2.8.

Diberikan masukan berupa data belajar dan masing-masing kelas dianotasikan untuk i = 1,2,3,…,l , dimana l adalah banyaknya data. Fungsi *hyperplane* dibuat dengan Persamaan (2.20). Dalam mencari nilai w dan b yang optimal, dilakukan dengan Persamaan (2.21).



Gambar 2.11 Ilustrasi *Support Vector Machines*

|  |  |
| --- | --- |
|  | (2.20) |

|  |  |
| --- | --- |
|  | (2.21) |

Persamaan diatas mempunyai variabel , dimana variabel tersebut adalah konstanta nilai pinalti dari kesalahan klasifikasi. Pencarian nilai dan dilakukan dengan batasan yang ditulis menggnakan Persamaan (2.22).

|  |  |
| --- | --- |
|  | (2.22) |

Persamaan (2.21) dilakukan untuk mencari nilai dan yang optimum. Fungsi tujuan Persamaan (2.21) berbentuk kuadrat. Untuk menyelesaikannya, bentuk tersebut ditransformasi kedalam bentuk *dual space*. Persamaan *dual space* dapat ditulis menggunakan Persamaan (2.23).

|  |  |
| --- | --- |
|  | (2.23) |

Dengan batasan pada Persamaan (2.24).

|  |  |
| --- | --- |
|  | (2.24) |

Untuk mencari nilai , digunakan *quadratic programming*. Setelah mendapatkan nilai , persamaan *hyperplane* dilakukan dengan Persamaan (2.25).

|  |  |
| --- | --- |
|  | (2.25) |

Dimana adalah data masukan. Pada banyak kasus, data yang diklasifikasikan tidak bisa langsung dipisahkan dengan garis yang linear. Oleh karena itu, digunakan metode kernel untuk mengatasi permasalahan tersebut. Dengan metode kernel, suatu data di *input space* dimapping ke fitur *space*  dengan dimensi yang lebih tinggi. Salah satu kernel yang biasa dipakai adalah kernel RBF dan *Polynomial*. Persamaan kernel RBF dan *Polynomial* berurutan dapat dilihat pada Persamaan (2.26) dan Persamaan (2.27).

|  |  |
| --- | --- |
|  | (2.26) |
|  | (2.27) |

Penggunaan fungsi kernel mengubah persamaan *training*. Persamaan dapat dilihat pada Persamaan (2.28).

|  |  |
| --- | --- |
|  | (2.28) |

Persamaan *hyperplane* diubah menjadi Persamaan (2.29).

|  |  |
| --- | --- |
|  | (2.29) |

***[Halaman ini sengaja dikosongkan]***

# DESAIN PERANGKAT LUNAK

Pada bab ini dijelaskan mengenai rancangan sistem perangkat lunak yang akan dibuat. Perancangan yang dijelaskan meliputi data dan proses. Data yang dimaksud adalah data yang akan diolah dalam perangkat lunak baik digunakan sebagai pembelajaran maupun pengujian sehingga tujuan Tugas Akhir ini bisa tercapai. Proses yaitu tahap-tahap yang ada dalam sistem sebagai pengolah data meliputi reduksi *size* *frame*, deteksi gerak, deteksi warna piksel, *region growing*, perhitungan luasan *region*, ekstraksi fitur dan klasifikasi.

## Data

Pada sub bab ini akan dijelaskan mengenai data yang digunakan sebagai masukan perangkat lunak untuk selanjutnya diolah dan dilakukan pengujian sehingga menghasilkan data keluaran yang diharapkan.

### Data Masukan

Data masukan adalah data yang digunakan sebagai masukan dari sistem. Data yang digunakan adalah data video yang memiliki frekuensi minimal 20 Hz, kualitasvideo yang digunakan adalah 240 x 320 piksel. Ilustrasi data masukan dapat dilihat pada Gambar 3.1.

### Data Pembelajaran

Data pembelajaran digunakan sebagai data belajar klasifikasi. Data yang digunakan adalah data video yang dibagi dalam dua jenis yaitu video berisi objek api dan video berisi objek bukan api. Data yang digunakan sebagai data pembelajaran adalah empat data video bukan api dan enam data video api.



Gambar 3.12 Contoh Data Masukan

### Data Keluaran

Data masukan akan diputar ulang dan diproses setiap *frame* menggunakan metode reduksi *frame*, deteksi gerak, deteksi warna piksel, *region growing*, perhitungan luasan *region*, ekstraksi fitur, dan klasifikasi. Dari proses tersebut diharapkan keluaran berupa tanda pada tiap frame jika memiliki piksel api. Tanda yang dimaksud adalah tanda berwarna biru yang menghubungkan empat nilai titik koordinat ekstrim piksel api. Ilustrasi data keluaran dapat dilihat pada Gambar 3.2.



Gambar 3.13 Contoh Data Keluaran

## Desain Sistem Secara Umum

Rancangan perangkat lunak deteksi api berbasis sensor visual menggunakan *support vector machines* dimulai dengan membaca masukan berupa file video. Proses deteksi api terdiri dari dua proses besar, yaitu *preprocessing* dan verifikasi. Diagram alir desain umum perangkat lunak ditunjukkan pada Gambar 3.3.

F:\KULIAH\TA\bukuTA\source\D0 - Main Process.png

Gambar 3.14 Diagram Alir Rancangan Perangkat Lunak Secara Umum

Setiap *frame* akan dilakukan *preprocessing* sebelum dilakukan verifikasi menggunakan *support vector machines*. Tahap pertama *preprocessing* adalah mengubah *size frame* yang diproses. Ukuran *frame* diubah menjadi emapt kali lebih kecil dari ukuran *frame* yang diproses. Tahap berikutnya adalah malakukan deteksi gerak *frame*, hasil dari deteksi gerak adalah piksel-piksel bergerak. Setelah mendapatkan piksel-piksel bergerak, tahap selanjutnya adalah deteksi warna setiap piksel menggunakan probabilitas distribusi gaussian. Warna api yang didefinisikan pada sistem ini adalah warna yang memiliki *range* antarawarna kuning hingga merah. Hasil keluaran dari metode deteksi warna piksel adalah piksel-piksel yang masuk kedalam kandidat piksel api.

Setelah mendapatkan kandidat piksel api, dilakukan metode *region growing* untuk mendapatkan *region* kandidat piksel api. Hasil dari *region growing* digunakan pada tahap selanjutnya yaitu perhitungan luasan *region*. Pada metode perhitungan luasan *region*, jika luasan *region* melebihi *threshold,* maka kandidat piksel yang masuk pada *region* tersebut merupakan kandidat piksel api selanjutnya.

Setelah melalui tahap *preprocessing*, piksel-piksel yang termasuk kandidat api akan di verifikasi menggunakan metode *support vector machines*. Fitur didapatkan dari nilai konstanta *wavelet* setiap piksel statis dengan sepuluh *frame* yang berurutan. Hasil akhir yang dikeluarkan adalah adanya penanda pada *frame* yang diproses jika *frame* tersebut mengandung piksel api. Data dibagi menjadi data pembelajaran dan data uji sehingga dapat diperoleh nilai *true positif*, *false positif*, dan *missing rate* dari hasil klasifikasi.

## Preprocessing

Setiap piksel pada *frame* yang diproses tidak langsung dilakukan verifikasi untuk menentukan apakah piksel tersebut piksel api atau bukan. Tahap awal yang dilakukan adalah *preprocessing. Preprocessing* bertujuan untuk menghilangkan piksel-piksel yang tidak memiliki karakteristik piksel api. *Preprocessing* dilakukan melalui lima tahap yaitu reduksi *size frame*, deteksi gerak, deteksi warna piksel, *region growing,* perhitungan luasan *region*. Diagram alir tahap *preprocessing* ditunjukkan pada Gambar 3.5.

### Reduksi Size Frame

Tahap reduksi *size frame* adalah tahap dimana *size* dari *frame* direduksi. Kualitas *frame* masukan direduksi dengan tujuan mempercepat proses pendeteksian api. Diagram alir proses reduksi *size frame* dapat dilihat pada Gambar 3.4.

F:\KULIAH\TA\bukuTA\source\reduksi frame.png

Gambar 3.15 Digram Alir Reduksi *Size Frame*



Gambar 3.16 Diagram Alir *Preprocessing*

Masukan dari tahap ini adalah *frame* yang sedang diproses dalam *channel* R,G,B. Proses reduksi dilakukan menggunakan metode *gaussian pyramid*. Hasil keluaran dari tahap ini adalah *frame* yang telah tereduksi kolom dan barisnya menjadi seperempat dari *frame* awal.

### Deteksi Gerak

Didalam proses ini dilakukan proses deteksi gerak piksel. Masukan dari proses ini adalah *frame* yang sedang di proses. Untuk mendapatkan kandidat piksel yang bergerak, dilakukan dengan metode *gaussian mixture models*. Dari metodetersebut didapatkan piksel-piksel yang bergerak. Ilustrasi deteksi gerak ditunjukkan pada Gambar 3.6.



Gambar 3.17 Contoh Deteksi Gerak

Setelah mendapatkan piksel yang bergerak, kumpulan piksel tersebut disimpan untuk diproses pada tahap berikutnya yakni deteksi warna piksel.

### Deteksi Warna Piksel

Penentuan warna piksel yang termasuk kandidat piksel api menggunakan metode distribusi probabilitas gaussian. Masukan dari tahap ini adalah *frame* dengan *channel* R,G,B yang sedang di proses dan *list* piksel bergerak yang didapatkan pada tahap deteksi gerak. Setiap piksel bergerak dihitung nilai perkalian probabilitas R,G,B dan ditentukan apakah piksel tersebut masuk kedalam kandidat piksel api atau tidak.

Sebelum melakukan perhitungan probabilitas pada setiap piksel yang bergerak, dilakukan proses mendapatkan nilai rata-rata dan standar deviasi *channel* R,G,B. Nilai rata-rata dan standar deviasi *channel* R,G,B didapatkan dari data yang diambil dari memproses *dataset* gambar api sebanyak sebelas gambar. Ilustrasi *dataset* gambar api dapat dilihat pada Gambar 3.7.



Gambar 3.18 Contoh *Dataset* Gambar Api

Perhitungan probabilitas piksel dilakukan dengan mencari probabilitas setiap nilai R,G,B menggunakan probabilitas distribusi gaussian. Selanjutnya nilai probabilitas R,G,B dikalikan untuk mendapatkan nilai probabilitas dari piksel tersebut. Jika nilai probabilitas piksel tersebut melebihi *threshold*, maka piksel tersebut dimasukan kedalam kandidat piksel api. *Pseudocode* fungsi dapat dilihat pada Gambar 3.8.

|  |  |
| --- | --- |
| 1.  2.  3.  4.  5.  6.  7. | for i=1 to length(CandidatePiksel)  R,G,B = CandidatePiksel[i].RGBvalue  If probability(R)\*probability(G)\* probability(B) > threshold  add CandidatePiksel[i] to firePiksel  else  add CandidatePiksel[i] to nonFirePiksel  return firePiksel |

Gambar 3.19 *Pseudocode* Deteksi Warna Piksel

Keluaran dari tahap ini adalah *list* kandidat piksel api yang akan diproses selanjutnya di tahap *region growing* dan perhitungan luasan *region*.

### Region Growing

Kandidat piksel api yang didapatkan pada tahap deteksi warna piksel dilakukan *region growing* untuk mendapatkan *region* dari setiap piksel. Masukan dari tahap ini adalah *list* kandidat piksel api dan *frame* yang diproses. Setiap piksel akan dilakukan *region growing* dengan cara melakukan pengecekan terhadap delapan tetangga piksel tersebut. Untuk menentukan apakah piksel tetangga tersebut termasuk *region* api atau tidak, dilakukan dengan cara menentukan nilai probabilitas warna R,G,B menggunakan distribusi probabilitas gaussian. Jika nilai probabilitas tersebut melebihi *threshold*, maka piksel tersebut dianggap satu *region*. *Pseudocode* fungsi dapat dilihat pada Gambar 3.9.

|  |  |
| --- | --- |
| 1.  2.  3.  4.  5.  6.  7.  8.  9.  10.  11.  12.  13.  14.  15.  16. | regionCounter = 0  region.size = image.size  clockWise = clockWise()  for i=1 to length(CandidatePiksel)  push CandidatePiksel[i] to stack regionStack  if CandidatePiksel[i].is\_visit == False  increment regionCounter  while regionStack is not empty  pop regionStack and assign to temporary  R,G,B = temporary.RGBvalue  if probability(R)\*probability(G)\*probability(B) > threshold and temporary.is\_visit == False  temporary.is\_visit = True  region[temporary] = regionCounter  for j to clockWise  push temporary.index – j to stack region Stack  return region |

Gambar 3.20 *Pseudocode Regon Growing*

Keluaran dari tahap *region growing* adalah *region* dengan nilai yang berbeda tiap *region*. Nilai tersebut membedakan antara satu *region* dengan *region* lainnya.

### Perhitungan Luasan Region

Perhitungan luasan *region* adalah metode untuk melakukan *filter* terhadap *region* api. Masukan dari tahap ini adalah *region* yang telah diperoleh dari metode *region growing*, kandidat piksel api yang didapatkan dari metode deteksi warna piksel. Pada tahap ini dilakukan proses perhitungan luasansetiap *region*. Jika luasan *region* melebihi *threshold*, maka kandidat piksel api yang ada pada *region* tersebut masuk kedalam kandidat piksel api. Gambar 3.10 adalah diagram alir dari proses perhitungan luasan *region*. Keluaran dari proses ini adalah *list* piksel yang termasuk kedalam kandidat piksel api.

## Verifikasi

Pada sub bab ini akan dijelaskan mengenai proses verifikasi piksel sehingga menghasilkan keluaran yang diharapkan seperti yang sudah dijelaskan sebelumnya. Setelah melalui tahap *preprocessing*, piksel-piksel yang masuk kedalam kandidat piksel api dilakukan tahap verifikasi menggunakan metode klasifikasi *support vector machines*. Sebelum dilakukan klasifikasi, terdapat proses untuk mendapatkan fitur sebagai data masukan klasifikasi. Pencarian fitur dilakukan dengan mengubah gambar spasial kedalam domain *wavelet*. Setelah mendapatkan fitur yang dicari, dilakukan klasifikasi untuk menentukan apakah piksel tersebut masuk kedalam piksel api atau bukan. Diagram alir tahap *preprocessing* ditunjukkan pada Gambar 3.11.

F:\KULIAH\TA\bukuTA\source\perhitungan luasan.png

Gambar 3.21 Diagram Alir Perhitungan Luasan *Region*

F:\KULIAH\TA\bukuTA\source\varifikasi.png

Gambar 3.22 Diagram Alir Verifikasi

### Ekstraksi Fitur dengan Wavelet

Data masukan yang digunakan pada ekstraksi fitur adalah data *frame* sebanyak sepuluh *frame* secara berurutan. *Frame* diambil dari *frame* yang sedang diproses berserta sembilan *frame* sebelumnya. Ekstraksi fitur dilakukan dengan mengubah *frame* kedalam domain *wavelet*. *Frame* yang diubah kedalam domain *wavelet* ini berupa sepuluh *frame* yang berurutan. Tiap *frame* akan menghasilkan empat sub *frame* baru, yaitu sub *frame* *low-low* (LL), *low-high* (LH), *high-low* (HL), dan *high-high* (HH). Untuk ekstraksi fitur, sub *frame* yang digunakan adalah sub *frame* LH, HL, HH. *Wavelet* yang digunakan adalah *wavelet* daubechies 4 . Tiap kandidat piksel api dicari nilai piksel pada sub *frame* *wavelet* yang telah dilakukan. Setiap piksel mendapatkan tiga nilai untuk setiap *framenya*, yaitu nilai LH, HL, HH. Dari ketiga nilai tersebut akan dihitung menggunakan Persamaan (3.1).

|  |  |
| --- | --- |
|  | (3.1) |

Total nilai yang didapatkan adalah sepuluh buah nilai fitur. Sebelum masuk tahap klasifikasi, nilai fitur yang didapatkan dinormalisasi menggunakan normalisasi min-max. Nilai min-max didapatkan dengan melakukan perhitungan nilai kuadrat sub *frame* LH, HL, HH. Selanjutnya nilai LH, HL, dan HH dikalkulasi menjadi satu, dicari nilai minimum dan maximum dari hasil perhitungan tersebut. Dari proses ini akan dilakukan terhadap seluruh *frame*, dimana nilai min-max yang didapatkan berjumlah sepuluh buah. Normalisasi dilakukan dengan menyesuaikan nilai min-max terhadap *frame* yang dicari nilai *wavelet* setiap pikselnya. Setelah dilakukan normalisasi, dilakukan pengurutan nilai fitur secara *ascending*. Diagram alir ekstraksi fitur dapat dilihat pada Gambar 3.12.

Fitur akhir yang didapatkan adalah fitur dengan dimensi sebesar sepuluh dimensi yang sudah terurut. Selanjutnya akan masuk kedalam tahap klasifikasi.

### Klasifikasi

Klasifikasi dilakukan dengan mengolah data fitur yang sudah didapatkan dari proses ekstraksi fitur. Metode klasifikasi yang digunakan adalah *support vector machines*. Masukan dari tahap klasifikasi adalah nilai fitur *wavelet* sebanyak sepuluh dimensi yang sudah dijelaskan sebelumnya . Hasil akhir dari proses ini adalah piksel-piksel yang masuk kedalam piksel api.

F:\KULIAH\TA\bukuTA\source\ekstraksi fitur.png

Gambar 3.23 Diagram Alir Ekstraksi Fitur

## Menandai Region Api

Penanda *region* api dilakukan untuk menunjukkan bagian yang terdeteksi api. Masukan dari proses ini adalah piksel-piksel yang lolos tahap verifikasi dan *frame* yang sedang di proses. *Pseudocode* fungsi ini dapat dilihat pada Gambar 3.13

|  |  |
| --- | --- |
| 1.  2.  3.  4.  5.  6. | Min\_x, Max\_x = min(X), max(X)  Min\_y, Max\_y = min(Y), max(Y)  For i = Min\_y to Max\_y:  For j = Min\_x to Max\_x :  Mark(Image[i][j])  Return Image |

Gambar 3.24 *Pseudocode* fungsi menandai *region*

Hasil yang dikeluarkan pada proses ini adalah *frame* yang memiliki tanda jika terdapat objek api. Ilustrasi menandai *region* api dapat dilihat pada Gambar 3.14.



Gambar 3.25 Menandai *region* api

***[Halaman ini sengaja dikosongkan]***

# IMPLEMENTASI

Pada bab ini akan dibahas mengenai implementasi yang dilakukan berdasarkan rancangan yang telah dijabarkan pada bab sebelumnya. Implementasi kode program dilakukan menggunakan bahasa Python.

## Lingkungan Implementasi

Spesifikasi perangkat keras dan perangkat lunak yang digunakan dalam implementasi ini ditampilkan pada Tabel 4.1.

Tabel 4.1 Lingkungan Perancangan Perangkat Lunak

|  |  |
| --- | --- |
| Perangkat | Spesifikasi |
| Perangkat keras | Prosesor: Intel® Core™ i3-2350M CPU @ 2.30GHz 2.30GHz  Memori: 4.00 GB |
| Perangkat lunak | Sistem Operasi: Microsoft Windows 8 64-bit Pro  Perangkat Pengembang: PyCharm  Perangkat Pembantu:  Microsoft Excel 2013 |

## Implementasi

Sub bab implementasi ini menjelaskan tentang implementasi proses yang sudah dijelaskan pada bab desain perangkat lunak.

### Implementasi Tahap Reduksi Size Frame

Sub bab ini membahas implementasi tahap reduksi *size frame*. Pada tahap ini data masukan berupa *frame* dan data keluaran yang dihasilkan pada tahap ini adalah *frame* yang telah tereduksi kualitasnya. Implementasi dilakukan dengan menggunakan fungsi yang sudah disediakan oleh OpenCV yaitu *pyrDown()* dan di tunjukkan oleh Kode Sumber 4.1.

|  |  |
| --- | --- |
| 1.  2. | currentFrame2 = copy.copy(currentFrame)  currentFrame = ImageProcessing.getDownSize(currentFrame) |

Kode Sumber 4.1 Implementasi Tahap Reduksi *Size Frame*

|  |  |
| --- | --- |
| 1.  2. | def getDownSize(self,image):  return cv2.pyrDown(image) |

Kode Sumber 4.2 Penggunaan Fungsi pyrDown()

Kode Sumber 4.2 melakukan reduksi dengan memanggil fungsi *getDownSize()* dam disimpan kedalam variabel *currentFrame*. Variabel *currentFrame2* akan digunakan sebagai masukan dari ekstraksi fitur dan *frame* keluaran yang ditampilkan, sedangkan proses lainnya akan menggunakan variabel *currentFrame*.

### Implementasi Tahap Deteksi Gerak

Sub bab ini membahas implementasi tahap deteksi gerak piksel. Masukan dari tahap ini adalah *frame* yang sedang diproses. Implementasi dilakukan menggunakan fungsi yang sudah disediakan oleh OpenCV yaitu *BackgroundSubtractorMOG()*. Implementasi ditunjukkan oleh Kode Sumber 4.3.

|  |  |
| --- | --- |
| 1.  2.  3. | def getMovingForeGround(self, image):  self.BckgrSbsMOG = cv2.BackgroundSubtractorMOG()  return self.BckgrSbsMOG.apply(image,learningRate = 0.0005) |

Kode Sumber 4.3 Implementasi Tahap Deteksi Gerak

Hasil yang dikembalikan oleh fungsi *getMovingForeGround()* adalah gambardengan nilai tiap pikselnya antara 0 atau 255 dengan. Dimana nilai 255 adalah nilai piksel yang bergerak terhadap piksel pada *frame* sebelumnya. Hasil dari fungsi *getMovingForeGround()* tidak secara langsung diproses kedalam tahap selanjutnya. Dari hasil *frame* ini akan diambil indeks piksel yang bergerak, implementasi ditunjukkan oleh Kode Sumber 4.4.

|  |  |
| --- | --- |
| 1.  2.  3. | def getMovingCandidatePiksel(self, moving\_frame):  listY,listX = np.where( moving\_frame == 255 )  return np.vstack((listY,listX)) |

Kode Sumber 4.4 Implementasi Penyimpanan Piksel

Fungsi *getMovingCandidatePiksel()* melakukan iterasi untuk mendapatkan piksel-piksel yang bernilai 255. Fungsi ini mengembalikan hasil berupa *list* indeks piksel yang bergerak. Hasil keluaran dari keseluruhan proses deteksi gerak adalah *list* yang berisi indeks piksel y dan x.

### Implementasi Tahap Deteksi Warna Piksel

Deteksi warna piksel dilakukan dengan pengecekan warna setiap piksel pada *list* pikselyang dihasilkan melalui proses deteksi gerak. Masukan dari tahap ini adalah *list* piksel yang telah didapatkan pada tahap deteksi gerak dan *frame* dengan *channel* R,G,B. Sebelum menghitung probabilitas warna piksel, terlebih dahulu dilakukan perhitungan mencari nilai rata-rata dan standar deviasi untuk nilai piksel R,G,B. Sebelas gambar api disimpan dan dilakukan proses perhitungan nilai rata-rata dan nilai standar deviasi untuk setiap *channel*. Pada implementasi sistem, proses menghitung nilai rata-rata dan standar deviasi setiap *channel* dapat dilihat pada Kode Sumber 4.5.

|  |  |
| --- | --- |
| 1.  2.  3.  4.  5.  6.  7.  8.  9.  10.  11.  12.  13.  14.  15.  16.  17.  18.  19.  20.  21.  22. | def getStdDevAndMean(self,path):  list\_file = File.readFolder(self,path)  R = []  G = []  B = []  for x in list\_file:  image = ImageProcessing.readImage(self,path+'/'+x)  r = np.array(image[:,:,2]).ravel()  g = np.array(image[:,:,1]).ravel()  b = np.array(image[:,:,0]).ravel()  R += (r.tolist())  G += (g.tolist())  B += (b.tolist())  mean = []  mean.append(np.average(B))  mean.append(np.average(G))  mean.append(np.average(R))  standard\_deviasi = []  standard\_deviasi.append(np.std(B))  standard\_deviasi.append(np.std(G))  standard\_deviasi.append(np.std(R))  return standard\_deviasi, mean |

Kode Sumber 4.5 Implementasi Menghitung Nilai Standar Deviasi dan Rata-Rata Setiap *Channel*

Pada Kode Sumber 4.5 setiap gambar *dataset* dibaca dan dilakukan pemisahan setiap *channel*. Selanjutnya, dilakukan perhitungan rata-rata dan perhitungan standar deviasi setiap *channel*. Setelah didapatkan nilai rata-rata dan standar deviasi setiap *channel*, proses penentuan kandidat api menggunakan metode probabilitas warna piksel dilakukan. Setiap piksel dihitung nilai probabilitas *channel* R,G,B. Selanjutnya, menghitung nilai probabilitas piksel dengan mengalikan nilai probabilitas R,G,B. Proses perhitungan probabilitas piksel memakan waktu yang cukup lama jika menghitung setiap kemungkinan kandidat piksel api yang lolos pada tahap deteksi gerak. Pada implementasi, nilai probabilitas tiap piksel yang masuk kedalam probabilitas warna api dimasukan kedalam file. Nilai yang disimpan adalah nilai R,G,B piksel yang termasuk kedalam warna piksel api. Hal ini dilakukan karena proses perhitungan yang lama jika menghitungan probabilitas setiap piksel. Isi dari file adalah *list* kemungkinan piksel-piksel yang masuk dalam probabilitas warna api. Proses *generate* *list* piksel api dapat dilihat pada Kode Sumber 4.6.

|  |  |
| --- | --- |
| 1.  2.  3.  4.  5.  6.  7.  8.  9.  10. | def getListColorPiksel  (self,list\_standard\_deviasi, list\_mean):  res = []  threshold = self.getThreshold()  for B in range(0,256):  for G in range(B,256):  for R in xrange(G,256):  if Data.getGaussianProbability(self,B, list\_standard\_deviasi[0], list\_mean[0])\* Data.getGaussianProbability(self,G, list\_standard\_deviasi[1], list\_mean[1])\* Data.getGaussianProbability(self,R, list\_standard\_deviasi[2], list\_mean[2]) > threshold:  res.append([B,G,R])  return res |

Kode Sumber 4.6 *Generate list* piksel api

Kode Sumber 4.6 melakukan iterasi pengecekan piksel. Iterasi tidak dilakukan sebanyak kombinasi nilai R,G,B. hal ini dikarenakan nilai *channel* R dari piksel api lebih besar dari nilai *channel* G dan nilai *channel* G lebih besar dari nilai *channel* B. Pada Kode Sumber 4.6, dilakukan pemanggilan fungsi *getGaussianProbability()*. Fungsi *getGaussianProbability()* digunakan untuk menghitung nilai probabilitas tiap *channel* piksel. Fungsi tersebut mengimplementasikan rumus dari probabilitas distribusi gaussian. Implementasi fungsi *getGaussianProbability()* dapat dilihat pada Kode Sumber 4.7.

|  |  |
| --- | --- |
| 1.  2.  3.  4.  5.  6.  7.  8.  9.  10 | def getGaussianProbability(self, data, standard\_deviasi, mean):  data = float(data)  standard\_deviasi = float(standard\_deviasi)  mean = float(mean)  result = pow((data-mean),2)  div = 2\*pow(standard\_deviasi,2)  exp = np.exp(-result/div)  result = standard\_deviasi\*np.sqrt(2\*np.pi)  result = 1/result  return result\* exp |

Kode Sumber 4.7 Fungsi Menghitung Nilai Probabilitas Distribusi Gaussian

|  |  |
| --- | --- |
| 1.  2. | def getThreshold(self):  return 5\*pow(10,-9) |

Kode Sumber 4.8 Mendapatkan Threshold

Kode Sumber 4.7 melakukan perhitungan probabilitas piksel api setiap *channel*. Nilai *threshold* pada Kode Sumber 4.8 didapatkan dari hasil analisa yang telah dilakukan. Ketika program dijalankan, program akan membaca file yang berisi *list* piksel yang sudah di *generate* sebelumnya dan menyimpan nilai piksel tersebut kedalam *array*. Proses pembacaan file bisa dilihat pada Kode Sumber 4.9.

|  |  |
| --- | --- |
| 1.  2.  3.  4.  5.  6.  7. | def getFireArray(self,path):  lists = [[[False for k in xrange(256)] for j in xrange(256)]for i in xrange(256)]  data = open(path,'r')  for x in data:  color = (x.split('\n')[0]).split(' ')  lists[int(color[0])] [int(color[1])][int(color[2])] = True  return lists |

Kode Sumber 4.9 Membaca *List* Piksel Api

Pada Kode Sumber 4.9 dilakukan peroses pembuatan *array* tiga dimensi, dengan *default*  *False* pada nilai indeks. Pada *line* enam, dilakukan perubahan nilai sesuai indeks yang ada pada *list* dengan mengubah nilai *array* tiga dimensi tersebut menjadi *True*. Nilai indeks dimensi *array* tersebut mewakilkan indeks R,G,B. Setelah didapatkan *array* dengan indeks yang mewakili nilai R,G,B dan hasilnya, dilakukan pengecekan terhadap kandidat piksel yang didapatkan pada proses deteksi gerak. Implementasi proses pengecekan kandidat piksel dapat dilihat pada Kode Sumber 4.10.

|  |  |
| --- | --- |
| 1.  2.  3.  4.  5.  6.  7.  8.  9.  10.  11.  12. | def getColorCandidatePiksel(self, list\_standard\_deviasi, list\_mean):  true\_piksel = []  false\_piksel = []  for x in range(0,len(list\_candidate[0])):  data = image[list\_candidate[0][x]] [list\_candidate[1][x]]  B,G,R = data[0],data[1],data[2]  if color\_dataset[B][G][R] == True:  true\_piksel.append([list\_candidate[0][x], list\_candidate[1][x]])  else : false\_piksel.append([list\_candidate[0][x], list\_candidate[1][x]])  return true\_piksel,false\_piksel |

Kode Sumber 4.10 Implementasi Tahap Deteksi Warna Piksel

Hasil yang dikeluarkan pada tahap ini adalah *list* nilai indeks piksel yang masuk kedalam piksel api.

### Implementasi Tahap Region Growing

Sub bab ini membahas implementasi tahap *region growing* yang menggunakan kandidat piksel api pada tahap deteksi warna api sebagai masukan. *Region growing* dilakukan menggunakan *list* kandidat piksel api sebagai masukan awal dari piksel yang dicari *region*nya. Titik piksel kandidat api dilakukan pengecekan probabilitas warna api terhadap delapan tetangga piksel tersebut. Jika piksel tetangga termasuk piksel api, maka piksel tersebut akan ditandai sebagai region dari piksel awalan tersebut. *Region* akan diberi nomor sesuai dengan titik piksel awal dari *region* tersebut. Implementasi *region growing* dapat dilihat pada Kode Sumber 4.11.

|  |  |
| --- | --- |
| 1.  2.  3.  4.  5.  6.  7.  8.  9.  10.  11.  12.  13.  14.  16. | def getRegionGrowing(self, list\_candidate, images, color\_dataset,counter):  gray\_image = ImageProcessing.getRGBtoGray(self,images)  is\_visit = gray\_image\*0  result\_image = copy.copy(gray\_image)  region\_number = 0  for x in list\_candidate:  coor\_y = x[0]  coor\_x = x[1]  if is\_visit[coor\_y][coor\_x] == 0:  stack = []  region\_number+=1  stack.append([coor\_y,coor\_x])  is\_visit[coor\_y][coor\_x] = region\_number  result\_image, is\_visit = self.doFloodFill( gray\_image, result\_image, is\_visit, stack, region\_number, color\_dataset, images)  return is\_visit |

Kode Sumber 4.11 Implementasi Tahap *Region Growing*

Kode Sumber 4.11 melakukan inisialisasi *region* dengan nilai 0 pada variabel *is\_visit*. Selanjutnya melakukan iterasi sebanyak kandidat piksel api yang masuk kedalam tahap ini. Setiap kandidat piksel api dilakukan pengecekan, apakah piksel tersebut sudah dilakukan *region growing* atau belum. Jika belum (nilai variabel indeks yang sedang dicek bernilai 0) , maka koordinat dari titik tersebut akan dijadikan sebagai *seed* untuk dimasukan kedalam *stack* dan dilakukan *growing*. Implementasi *growing* dapat dilihat pada Kode Sumber 4.12.

Kode Sumber 4.12 melakukan pegecekan terhadap *stack* piksel yang akan dicek. Jika stack masih memiliki nilai, maka nilai tersebut akan digunakan sebagai *seed* untuk melakukan cek terhadap tetangga piksel. Jika tetangga piksel belum mempunyai *region* dan masuk kedalam *region* tersebut, maka nilai dari variabel *is\_visit* diganti sesuai dengan nilai *region* dan koordinat piksel tersebut dimasukan kedalam *stack* untuk melakukan pengecekan tetangga selanjutnya. Pada fungsi *growing()* melakukan pemenggilan fungsi *getClockWise()*. Implementasi *getClockWise()* dapat dilihat pada Kode Sumber 4.12.

|  |  |
| --- | --- |
| 1.  2.  3.  4.  5.  6.  7.  8.  9.  10.  11.  12.  13.  14.  15.  16.  17. | def growing(self, gray\_image ,result\_image ,is\_visit, stack, region\_number, color\_dataset, original\_image):  clocks = Data.getClockwise(self)  while len(stack) != 0:  coory,coorx = stack[0]  stack.pop(0)  result\_image[coory][coorx] = 255  data = original\_image[coory][coorx]  B,G,R = data[0],data[1],data[2]  for x in clocks:  try :  if is\_visit[coory+x[0]][coorx+x[1]] == 0 and color\_dataset[B][G][R] == True :  is\_visit[coory+x[0]] [coorx+x[1]] = region\_number  stack.append([coory+x[0], coorx+x[1]]  except :  pass  return result\_image,is\_visit |

Kode Sumber 4.12 Implementasi *Growing*

|  |  |
| --- | --- |
| 1.  2.  3.  4.  5.  6.  7.  8.  9.  10.  11. | def getClockwise(self):  clocks = []  clocks.append([-1,-1])  clocks.append([-1,0])  clocks.append([-1,1])  clocks.append([0,-1])  clocks.append([0,1])  clocks.append([1,-1])  clocks.append([1,0])  clocks.append([1,1])  return clocks |

Kode Sumber 4.13 Implementasi Tahap *Clock Wise*

Iterasi yang dilakukan pada *getClockWise()* akan melakukan pengecekan delapan tetangga searah jarum jam. Hasil keluaran dair ptoses ini adalah *region* yang mempunyai nomor *region* yang berbeda antar *region*.

### Implementasi Tahap Perhitungan Luasan Region

Sub bab ini membahas implementasi tahap perhitungan luasan *region*. Masukan dari tahap ini adalah kandidat piksel api yang telah didapatkan dari tahap probabilitas warna api, dan *region*. Pada tahap perhitungan luasan *region* dihitung banyaknya piksel yang ada pada *region* tersebut. Jika luasan piksel *region* tersebut melebihi batas, maka kandidat piksel yang ada pada *region* tersebut masuk sebagai kandidat piksel api . Implementasi dapat dilihat pada Kode Sumber 4.14.

|  |  |
| --- | --- |
| 1.  2.  3.  4.  5.  6.  7.  8.  9.  10.  11.  12.  13.  14.  15.  16.  17.  18.  19. | def getFilterSizeRegion (self,list\_candidate,region):  true\_piksel = []  false\_piksel = []  list\_region = np.unique(region)  threshold = dict()  for x in range(1,len(list\_region)):  lists = np.where(region == x)  if len(lists[0]) > 1\*len(region)\* len(region[0])/100:  threshold[x] = True  else :  threshold[x] = False  for x in list\_candidate:  coor\_y = x[0]  coor\_x = x[1]  if threshold[region[coor\_y][coor\_x]] == True:  true\_piksel.append([coor\_y,coor\_x])  else :  false\_piksel.append([coor\_y,coor\_x])  return true\_piksel,false\_piksel |

Kode Sumber 4.14 Implementasi Tahap Variasi Warna *Region*

Setiap *region* akan dilakukan iterasi dan dilakukan pengecekan. Jika luasan dari *region* tersebut memenuhi syarat, maka seluruh piksel kandidat api masuk kedalam proses berikutnya. Hasil keluaran dari fungsi ini adalah *list* piksel yang yang masuk kedalam kandidat piksel api.

### Implementasi Tahap Ekstraksi Fitur dengan Wavelet

Sub bab ini membahas implementasi tahap ekstraksi fitur. Ekstraksi fitur dilakukan dengan mengubah *frame* kedalam domain *wavelet*. Digunakan sepuluh buah *frame* yang berurutan untuk diubah kedalam domain *wavelet*. Pada implementasi setiap *frame* diubah kedalam domain *wavelet* dan disimpan kedalam *list*. Implementasi mengubah dan manyimpan *frame* domain *wavelet* dapat dilihat pada Kode Sumber 4.15.

|  |  |
| --- | --- |
| 1.  2. | LL,(HL,LH,HH) = wv.toWavelet(copy.copy(grayImage))  list\_wavelet.append([HL,LH,HH]) |

Kode Sumber 4.15 Implementasi Memasukan Nilai *Wavelet* Kedalam *List*

Pada Kode Sumber 4.16, fungsi *toWavelet()* digunakan untuk mengubah *frame* kedalam domain *wavelet*. Implementasi mengubah *frame* kedalam domain *wavelet* dapat dilihat pada Kode Sumber 4.16.

|  |  |
| --- | --- |
| 1.  2. | def toWavelet(image):  return pywt.dwt2(image,'db2') |

Kode Sumber 4.16 Pemanggilan Fungsi *Wavelet*

### Implementasi Tahap Klasifikasi

Sub bab ini membahas implementasi tahap klasifikasi. Pada tahap ini sistem diberi masukan kandidat piksel api dan sepuluh buah *frame* domain *wavelet*, dimana masing-masing *frame* berisi tiga buah sub *frame* seperti penjelasan sub bab 3.4.1 . Sebelum melakukan klasifikasi, data training terlebih dahulu diproses sebagai data *training* untuk klasifikasi. Implementasi *training* klasifikasi dapat dilihat pada Kode Sumber 4.17.

|  |  |
| --- | --- |
| 1.  2.  3. | def getClassifier(datatraining):  x,y = readDataSet(datatraining)  clf = svm.SVC(kernel = 'rbf',C = 3.5)  clf.fit(x,y)  return clf |

Kode Sumber 4.17 *Training* Klasifikasi

Setiap piksel pada kandidat piksel api dilakukan perhitungan nilai fitur seperti yang sudah dijelaskan pada sub bab 3.4.1. Implementasi klasifikasi dapat dilihat pada Kode Sumber 4.18.

|  |  |
| --- | --- |
| 1.  2.  3.  4.  5.  6.  7.  8.  9.  10.  11.  12.  13.  14.  15.  16.  17.  18.  19.  20.  21.  22.  23.  24.  25.  26.  27.  28. | def doClassification(classifier, list, wavelet):  truePiksel = []  falsePiksel = []  listMax = []  listMin = []  cpyWavelet = np.int\_(copy.copy(wavelet))  cpyWavelet = np.power(cpyWavelet,2)  for x in cpyWavelet:  lists = np.add(np.add(x[0],x[1]) , x[2])  listMax.append(np.max(lists))  listMin.append(np.min(lists))  for x in list:  data = []  cnt= 0  for y in wavelet:  res = pow(y[0][x[0]][x[1]],2) + pow(y[1][x[0]][x[1]],2) + pow(y[2][x[0]][x[1]],2)  res = (float(res)-float(listMin[cnt]))/(float(listMax[cnt])-float(listMin[cnt]))  res = float('%.2f' % res)  cnt+=1  data.append(res)  data = np.sort(data)  classes = classifier.predict(data)  if classes == 'Api':  truePiksel.append([x[0],x[1]])  else :  falsePiksel.append([x[0],x[1]])  return truePiksel,falsePiksel |

Kode Sumber 4.18 Implementasi Tahap Klasifikasi

Pada Kode Sumber 4.18 dilakukan iterasi sebanyak kandidat piksel api yang lolos ketahap verifikasi. Setiap piksel akan dihitung nilai fitur *wavelet*. Dilakukan normalisasi nilai fitur yang dilakukan pada *line* 18. Jika hasil klasifikasi suatu fitur adalah api, maka nilai indeks dari piksel tersebut akan dimasukan kedalam *list*. Hasil yang dikeluarkan dari proses ini adalah *list* piksel api yang lolos tahap verifikasi.

### Implementasi Tahap Menandai Region Api

Sub bab ini membahas implementasi menandai *region* api. Piksel api yang sudah lolos tahap verifikasi selenjutnya diproses sebagai data keluaran yang ditampilkan. Masukan dari tahap ini adalah kandidat piksel api dan *frame* dari variabel *currentFrame2*. Karena perbedaan ukuran antara indeks piksel api, dilakukan normalisasi indeks. Dilakukan penyesuaian indeks-indeks piksel dengan *frame* keluaran. Implementasi *training* klasifikasi dapat dilihat pada Kode Sumber 4.19.

|  |  |
| --- | --- |
| 1.  2.  3.  4.  5.  6.  7.  8.  9.  10.  11.  12.  13.  14.  15.  16.  17. | def markingFire(self, list\_fire, image, constanta):  if len(list\_fire) == 0:  return image  list = np.array(list\_fire)  min\_y,max\_y = min(list[:,0]),max(list[:,0])  min\_x,max\_x = min(list[:,1]),max(list[:,1])  distance\_y = int((max\_y-min\_y)/2)\*constanta  distance\_x = int((max\_x-min\_x)/2)\*constanta  center\_point = [int((max\_y+min\_y)\*constanta/2) , int((max\_x+min\_x)\*constanta/2)]  min\_y,min\_x,max\_y,max\_x = center\_point[0]-distance\_y, center\_point[1] - distance\_x, center\_point[0] + distance\_y, center\_point[1] +distance\_x  for y in range(min\_y,max\_y+1):  image[y][min\_x] = [255,191,0]  image[y][max\_x] = [255,191,0]  for x in range(min\_x,max\_x+1):  image[min\_y][x] = [255,191,0]  image[max\_y][x] = [255,191,0]  return image |

Kode Sumber 4.19 Implementasi Tahap Menandai Region Api

Hasil keluaran dari Kode Sumber 4.19 adalah *frame* dengan tanda persegi jika terdapat piksel api pada *frame* yang diproses.

# UJI COBA DAN EVALUASI

Pada bab ini akan dijelaskan hasil uji coba dan evaluasi program yang telah selesai diimplementasi.

## Lingkungan Uji Coba

Lingkungan uji coba yang akan digunakan adalah,

* 1. Perangkat Keras

Prosesor Intel® Core™ i3-2350M CPU @ 2.30GHz 2.30GHz RAM 4 GB.

Sistem Operasi 64-bit .

* 1. Perangkat Lunak

Sistem Operasi Microsoft Windows 8 64-bit Pro.

Perangkat Pengembang PyCharm.

## Data Uji Coba

Data yang digunakan untuk uji coba implementasi deteksi api berbasis sensor visual menggunakan metode *support vector machines* adalah potongan video yang didapatkan dari berbagai sumber. Kualitas video yang digunakan adalah video dengan *size* 240x320 piksel dan memiliki *channel* R,G,B. Data video yang digunakan diambil dari beberapa kejadian. Data video yang digunakan meliputi dua buah jenis video. Video dengan objek api dan video dengan objek bukan api. Jumlah video yang diuji berjumlah enam puluh tujuh video dengan jumlah video api sejumlah tiga puluh empat dan video bukan api berjumlah tiga puluh tiga. Contoh video kejadian dapat dilihat pada Gambar 5.1.

## Alur Uji Coba

Pada sub bab ini akan dijelaskan mengenai alur kerja dari sistem deteksi api. Dimulai dari *preprocessing* hingga verifikasi.

|  |  |  |  |
| --- | --- | --- | --- |
| Movie 1 | Movie 2 | Movie 3 | Movie 4 |
| Movie 5 | Movie 6 | Movie 7 | Movie 8 |
| Movie 9 | Movie 10 | Movie 11 | Movie 12 |
| Movie 13 | Movie 14 | Movie 15 | Movie 16 |

Gambar 5.26 Contoh Video Kejadian

### Preprocessing

Tahap *preprocessing* akan dijelaskan bagaimana alur setiap *frame* masuk hingga menghasilkan kandidat api yang selanjutnya akan di proses pada tahap verifikasi. Ilustrasi tahap *proprocessing* dapat dilihat pada Gambar 5.2.

|  |  |
| --- | --- |
| F:\KULIAH\TA\bukuTA\source\original_91.png | *Frame* Masukan |
|  |  |
| F:\KULIAH\TA\bukuTA\source\original_91.png | Reduksi *Size Frame* |
|  |  |
| F:\KULIAH\TA\bukuTA\source\moving_91.png | Deteksi Gerak |
|  |  |
| F:\KULIAH\TA\bukuTA\source\color_91.png | Deteksi Warna Piksel |
|  |  |
| F:\KULIAH\TA\bukuTA\source\region_91.png | *Region Growing* |
|  |  |
| F:\KULIAH\TA\bukuTA\source\size_91.png | Perhitungan Luasan *Region* |

Gambar 5.27 Tahap *Preprocessing*

### Verifikasi

Tahap verifikasi akan dijelaskan bagaimana alur verifikasi dilakukan. Masukan dari tahap ini adalah hasil akhir dari tahap *preprocessing*. Hasil akhir dari proses verifikasi adalah *region* yang masuk kedalam objek api. Ilustrasi proses verifikasi dapat dilihat pada Gambar 5.3.

|  |  |
| --- | --- |
| F:\KULIAH\TA\bukuTA\source\size_91.png | *Frame* Masukan dari proses *preprocessing* |
|  |  |
| |  | | --- | |  | |  | |  | | Ekstraksi Fitur, detail gambar vertikal, horizontal, digonal |
|  |  |
| F:\KULIAH\TA\firedetection\originial\-laporan- gambar\final_91.png | *Support Vector Machines* |
|  |  |
| F:\KULIAH\TA\bukuTA\source\fire_91.png | Menandai *Region* Api |

Gambar 5.28 Tahap Verifikasi

## Skenario Uji Coba

Pada sub bab ini akan dijelaskan mengenai skenario uji coba yang telah dilakukan. Telah dilakukan beberapa skenario uji coba, diantaranya yaitu:

1. Perbandingan hasil *true positif*, *false positif*, dan *missing rate* berdasarkan variasi nilai *threshold* pada deteksi warna api. *Threshold* yang akan diuji yaitu 10-7, 10-8, 5 x 10-9, 10-9.
2. Perbandingan hasil *true positif*, *false positif*, dan *missing rate* berdasarkan variasi nilai (*penalty error term*) pada klasifikasi dengan kernel tetap yaitu RBF. Nilai yang akan diuji yaitu 1, 3.5, 5,dan 7.
3. Perbandingan hasi *true positif*, *false positif*, dan *missing rate* berdasarkan variasi kernel yang digunakan pada klasifikasi. Kernel yang akan diuji yaitu *polynomial 2, polynomial 3*, dan RBF.
4. Perbandingan hasil *true positif*, *false positif*, dan *missing rate* berdasarkan variasi besarnya *region* objek. Variasi yang digunakan adalah 1%, 5%, dan 10%.
5. Perbandingan kecepatan deteksi dengan variasi *size frame* yang telah direduksi. *Size frame* yang diuji adalah 240 x 320, 120 x 160, 60 x 80.
6. Perbandingan hasil *true positif*, *false positif*, dan *missing rate* dengan menghilangkan tahap *region growing* dan perhitungan luasan *region*.

### Skenario Uji Coba 1

Skenario uji coba 1 adalah perhitungan *true positif*, *false positif*, dan *missing rate*. Dimana *true positif* adalah kondisi suatu *frame* mengandung gambar api dan terdeteksi api atau *frame* tidak mengandung api dan tidak terdeteksi api. *False positif* adalah kondisi dimana *frame* tidak mengandung gambar api, namun terdeteksi api dan *missing rate* adalah keadaan dimana suatu *frame* yang mempunyai gambar api namun tidak terdeteksi api. Pada skenario uji coba 1 dilakukan uji coba pada tahap probabilitas warna api dengan mengubah nilai *threshold* probabilitas piksel api. Nilai *threshold* yang diuji yaitu 10-7, 10-8, 5 x 10-9, 10-9. Untuk parameter nilai pada uji coba 1 diberikan nilai 5 menggunakan kernel RBF.

Tabel 5.1 Hasil Uji Coba 1

|  |  |  |  |
| --- | --- | --- | --- |
| *Threshold* | *True Positif (%)* | *False Positif (%)* | *Missing Rate (%)* |
| 10-7 | 77.58 | 0.37 | 22.05 |
| 10-8 | 94.58 | 1.22 | 4.19 |
| 5x10-9 | 96.32 | 1.46 | 2.23 |
| 10-9 | 91.95 | 7.85 | 0.20 |

Dari hasil uji yang dilakukan, semakin kecil nilai *threshold* yang digunakan, hasil dari *true positif* akan semakin besar. Begitu juga untuk *false positif*, dimana makin kecil nilai *threshold* makin besar nilai *false positif*. Hal ini dikarenakan piksel yang dianggap piksel api sudah melewati batas warna kuning hingga merah. Dari uji coba tersebut didapatkan nilai *threshold* 5x10-9 sebagai nilai terbaik, karena *False positif* yang dihasilkan tidak terlalu besar dan *True Positif*  bernilai besar. Hasil uji coba 1 lebih lengkap terdapat pada lampiran.

### Skenario Uji Coba 2

Skenario uji coba 2 dilakukan dengan menghitung nilai *true positif, false positif,* dan *missing rate*. Pada skenario uji 2 dilakukan uji coba variasi nilai pada klasifikasi, dimana nilai adalah nilai *penalty error term*. Nilai yang diuji yaitu 1, 3.5, 5, dan 7. Untuk parameter *threshold* warna piksel diberikan nilai 5 x 10-9 menggunakan kernel RBF sebagai klasifikasi. Hasil uji coba 2 lebih lengkap terdapat pada lampiran.

Tabel 5.2 Hasil Uji Coba 2

|  |  |  |  |
| --- | --- | --- | --- |
|  | *True Positif (%)* | *False Positif (%)* | *Missing Rate (%)* |
| 1 | 96.30 | 1.43 | 2.27 |
| 3.5 | 96.32 | 1.46 | 2.23 |
| 5 | 96.32 | 1.46 | 2.23 |
| 7 | 96.32 | 1.46 | 2.23 |

Hasil uji coba tahap 2 hasil terbaik didapatkan ketika nilai = 7, dimana memiliki nilai *true positif* yang lebih tinggi dari yang lainnya.

### Skenario Uji Coba 3

Pada skenario uji 3 dilakukan uji coba variasi kernel klasifikasi. Variasi kernel yang digunakan yaitu *polynomial 2, polynomial 3*, dan RBF. Untuk parameter *threshold* warna piksel diberikan nilai 5 x 10-9 dan nilai diberikan nilai 5.

Tabel 5.3 Hasil Uji Coba 3

|  |  |  |  |
| --- | --- | --- | --- |
| Kernel | *True Positif (%)* | *False Positif (%)* | *Missing Rate (%)* |
| *Polynomial 2* | 87.93 | 0.95 | 11.12 |
| *Polynomial 3* | 65.78 | 0.05 | 34.17 |
| RBF | 96.32 | 1.46 | 2.23 |

Hasil uji coba tahap 3 didapatkan nilai *true positif*  terbaik didapatkan dengan menggunakan kernel RBF. Hasil uji coba 3 lebih lengkap terdapat pada lampiran.

### Skenario Uji Coba 4

Pada skenario uji 4 dilakukan uji coba variasi besarnya *region*. Variasi *region* yang digunakan yaitu 1%, 5%, 10%. Untuk parameter *threshold* warna piksel diberikan nilai 5 x 10-9 , nilai diberikan nilai 5 dan menggunakan kernel RBF.

Tabel 5.4 Hasil Uji Coba 4

|  |  |  |  |
| --- | --- | --- | --- |
| Konstanta *Region* | *True Positif (%)* | *False Positif (%)* | *Missing Rate (%)* |
| 1% | 96.32 | 1.46 | 2.23 |
| 5% | 62.92 | 0.47 | 36.61 |
| 10% | 53.99 | 0.41 | 45.60 |

### Skenario Uji Coba 5

Pada skenario uji 5 dilakukan uji coba variasi *size frame*. Variasi *size frame* yang digunakan yaitu 240 x 320, 120 x 160, 60 x 80. Untuk parameter *threshold* warna piksel diberikan nilai 5 x 10-9 , nilai diberikan nilai 5 dan menggunakan kernel RBF.

Tabel 5.5 Hasil Uji Coba 5

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Piksel | *True Positif (%)* | *False Positif (%)* | *Missing Rate (%)* | *Execution Time (s)* |
| 240 x 320 | 92.96 | 0.79 | 6.25 | 123.03 |
| 120 x 160 | 96.32 | 1.46 | 2.23 | 35.89 |
| 60 x 80 | 96.40 | 3.22 | 0.37 | 11.75 |

### Skenario Uji Coba 6

Pada skenario uji coba 6, dilakukan Perbandingan hasil *true positif*, *false positif*, dan *missing rate* dengan menghilangkan tahap *region growing* dan perhitungan luasan *region*.

Tabel 5.6 Hasil Uji Coba 6

|  |  |  |  |
| --- | --- | --- | --- |
|  | *True Positif (%)* | *False Positif (%)* | *Missing Rate (%)* |
| Tanpa Menggunakan *region growing* dan perhitungan luasan *region* | 88.62 | 11.38 | 0.00 |
| Menggunakan *region growing* dan perhitungan luasan *region* | 96.32 | 1.46 | 2.23 |

## Analisis Hasil Uji Coba

Dari hasil skenario uji coba yang telah dilakukan, beberapa parameter memberikan pengaruh terhadap hasil deteksi. Parameter yang digunakan antara lain nilai *threshold* pada deteksi warna api dan nilai pada klasifikasi. Uji coba dilakukan dengan membandingkan nilai *true positif*, *false pisitif*, dan *missing rate*.

Dari uji coba 1, parameter yang di uji adalah *threshold* pada deteksi warna piksel. Hasil percobaan menunjukkan semakin tinggi nilai *threshold* yang digunakan, semakin besar *missing rate* yang dihasilkan. Hal ini dikarenakan semakin kecil kombinasi warna piksel yang dianggap sebagai warna api. Sebaliknya jika nilai *threshold* yang digunakan terlalu kecil, maka banyak piksel yang tidak termasuk warna api lolos sebagai kandidat piksel berwarna api. Nilai terbaik yang didapatkan dari hasil percobaan adalah nilai *threshold* 5 x 10-9.

Uji coba 2, dapat diambil kesimpulan bahwa nilai tidak berpengaruh besar pada klasifikasi. Pada uji coba 3, variasi kernel yang digunakan adalah *polynomial 3*, dan RBF. Hasil uji coba menunjukkan kernel terbaik dari variasi kernel yang digunakan adalah kernel RBF. Uji coba 4, dapat diambil kesimpulan jika konstanta yang digunakan terlalu besar banyak *region* yang dianggap *noise*. Pada uji coba 5, semakin kecil *size frame* yang diproses, waktu eksekusi yang diperlukan semakin kecil. Pada uji coba 6, penghilangan proses *region growing* dan perhitungan luasan *region* menurunkan hasil yang dikeluarkan. Hal ini disebabkan banyaknya piksel-piksel *noise* yang masuk kedalam piksel api.

Dari keseluruhan uji coba yang dilakukan, parameter-parameter tersebut menghasilkan presentase terbaik ketika *threshold* yang digunakan 5 x 10-9 dan yang digunakan sebesar 7 dan menggunakan kernel RBF.

***[Halaman ini sengaja dikosongkan]***

# KESIMPULAN DAN SARAN

Pada bab ini akan dijelaskan mengenai kesimpulan dari proses dan uji coba dari program dan saran untuk pengembangan dari program itu sendiri.

## Kesimpulan

Dari hasil uji coba yang telah dilakukan, dapat diambil kesimpulan sebagai berikut :

1. Reduksi *size frame* mempercepat proses deteksi. Dari hasil Skenario Uji Coba 5, reduksi *size frame* mempengaruhi hasil deteksi namun tidak terlalu besar.
2. Metode deteksi gerak menggunakan *gaussian mixture model* menyaring piksel-piksel pada *frame* dengan *thereshold* yang berbeda setiap piksel dan dapat beradaptasi dengan waktu. Sehingga piksel-piksel yang bergerak dapat disaring dengan baik.
3. Deteksi warna menyaring piksel-piksel yang tidak masuk kedalam *range* warna api menggunakan probabilitas distribusi gaussian menyaring warna piksel api dengan baik. *Threshold* terbaik didapatkan sebesar 5 x 10-9, dapat dilihat pada Skenario Uji Coba 1.
4. Metode perhitungan luasan *region* dapat menghilangkan *noise* dengan baik. Pada Skenario Uji Coba 6, dapat disimpulkan bahwa penggunaan perhitungan luasan *region* meningkatkan hasil deteksi.
5. Penggunaan kernel pada klasifikasi mempengaruhi hasil dari verifikasi piksel, dapat dilihat pada Skenario Uji Coba 3. Kernel terbaik pada Skenario Uji Coba 3 adalah RBF.
6. Hasil terbaik pada uji coba adalah menggunakan nilai *threshold =* 5 x 10-9 dan nilai = 7. Menghasilkan nilai *true positif* sebesar 96.32, *false positif* sebesar 1.46 dan *missing rate* sebesar 2.23.

## Saran

Saran yang diberikan untuk pengembangan perangkat lunak ini adalah :

1. Analisa fitur pada pross verifikasi perlu dilakukan analisa lebih lanjut.

# DAFTAR PUSTAKA

|  |  |
| --- | --- |
| [1] | K.-H. C. J.-Y. N. ByoungChul Ko, "Early fire detection algorithm based on irregular patterns of flames and," *Fire Safety Journal,* pp. 262-270, 2007. |
| [2] | R. B. P. KaewTraKulPong, "An Improved Adaptive Background Mixture Model for Real- time Tracking with Shadow Detection," Kluwer Academic Publishers, 2001. |
| [3] | "Images Pyramid Open CV," 2011-2014. [Online]. Available: http://docs.opencv.org/2.4/doc/tutorials/imgproc/pyramids/pyramids.html. [Accessed 6 1 2016]. |
| [4] | E. H. A. PETER J. BURT, "The Laplacian Pyramid as a Compact Image Code," *IEEE ,* Vols. COM-31, pp. 522-540, 1983. |
| [5] | I. S. T. Maria Isabel Ribeiro, "Gaussian Probability Density Functions: Properties and Error Characterization," 2004. |
| [6] | R. E. W. Refael C. Gonzalez, Digital Image Processing third edition, p. 785. |
| [7] | R. S. F. D. R. S. Lee A. Barford, "An Introduction to Wavelets," 1992. |
| [8] | B. K. HYERAN BYUN, "ROBUST FACE DETECTION AND TRACKING FOR REAL-LIFE APPLICATIONS," *International Journal of Pattern Recognition,* vol. 17, pp. 1035-1055, 2003. |
| [9] | C.-J. L. Chih-Chung Chang, "A Library for Support Vector Machines," Taipei, Taiwan, 2001. |
| [10] | S. T. Punam Patel, "Flame Detection using Image Processing Techniques," *International Journal of Computer Applications,* vol. 58, pp. 13-16, 2012. |

***[Halaman ini sengaja dikosongkan]***

# LAMPIRAN A

Tabel A.1 Hasil Uji Coba Menggunakan Parameter *Threshold*  = 10-7, = 5 dan kernel RBF

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Nama File | True Positif | False Positif | Missing Rate |
| 1 | api-bakar\_sampah.avi | 6.25 | 0.00 | 93.75 |
| 2 | api-boneka\_dora.avi | 6.37 | 0.00 | 93.63 |
| 3 | api-kayu.avi | 66.18 | 0.00 | 33.82 |
| 4 | api-kayu2.avi | 77.94 | 0.00 | 22.06 |
| 5 | api-kebakaran\_barang.avi | 97.55 | 0.00 | 2.45 |
| 6 | api-kebakaran\_barang2.avi | 100.00 | 0.00 | 0.00 |
| 7 | api-kebakaran\_hutan.avi | 100.00 | 0.00 | 0.00 |
| 8 | api-kebakaran\_hutan2.avi | 100.00 | 0.00 | 0.00 |
| 9 | api-kebakaran\_hutan3.avi | 100.00 | 0.00 | 0.00 |
| 10 | api-kebakaran\_hutan4.avi | 0.48 | 0.00 | 99.52 |
| 11 | api-kebakaran\_ladang.avi | 98.22 | 0.00 | 1.78 |
| 12 | api-kebakaran\_mobil.avi | 98.04 | 0.00 | 1.96 |
| 13 | api-kebakaran\_mobil2.avi | 100.00 | 0.00 | 0.00 |
| 14 | api-kebakaran\_mobil3.avi | 95.59 | 0.00 | 4.41 |
| 15 | api-kebakaran\_mobil4.avi | 97.06 | 0.00 | 2.94 |
| 16 | api-kebakaran\_mobil5.avi | 47.55 | 0.00 | 52.45 |
| 17 | api-kebakaran\_mobil6.avi | 98.53 | 0.00 | 1.47 |
| 18 | api-kebakaran\_rumah.avi | 56.86 | 0.00 | 43.14 |
| 19 | api-kebakaran\_rumah2.avi | 60.29 | 0.00 | 39.71 |
| 20 | api-kebakaran\_rumah3.avi | 16.18 | 0.00 | 83.82 |
| 21 | api-kebakaran\_rumah4.avi | 1.47 | 0.00 | 98.53 |
| 22 | api-kebakaran\_tol.avi | 60.78 | 0.00 | 39.22 |
| 23 | api-kebakaran-truck.avi | 50.49 | 0.00 | 49.51 |
| 24 | api-kebakaran-truck2.avi | 62.75 | 0.00 | 37.25 |
| 25 | api-kebakaran-truck3.avi | 16.18 | 0.00 | 83.82 |
| 26 | api-kertas.avi | 0.00 | 0.00 | 100.00 |
| 27 | api-kertas2.avi | 4.94 | 0.00 | 95.06 |
| 28 | api-miniatur\_mainan.avi | 10.06 | 0.00 | 89.94 |
| 29 | api-mobil\_mainan.avi | 0.00 | 0.00 | 100.00 |
| 30 | api-orang\_terjun.avi | 46.08 | 0.00 | 53.92 |
| 31 | api-pesawat\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 32 | api-pesawat\_mainan2.avi | 47.93 | 0.00 | 52.07 |
| 33 | api-ruang\_tamu.avi | 0.00 | 0.00 | 100.00 |
| 34 | api-rumah\_mainan.avi | 99.02 | 0.00 | 0.98 |
| 35 | non\_api-anak\_kecil.avi | 100.00 | 0.00 | 0.00 |
| 36 | non\_api-anak\_kecil2.avi | 100.00 | 0.00 | 0.00 |
| 37 | non\_api-anak\_kecil3.avi | 100.00 | 0.00 | 0.00 |
| 38 | non\_api-bertemu.avi | 100.00 | 0.00 | 0.00 |
| 39 | non\_api-jaket\_merah.avi | 100.00 | 0.00 | 0.00 |
| 40 | non\_api-jalan\_malam.avi | 100.00 | 0.00 | 0.00 |
| 41 | non\_api-jalan\_malam2.avi | 100.00 | 0.00 | 0.00 |
| 42 | non\_api-jalan\_raya.avi | 100.00 | 0.00 | 0.00 |
| 43 | non\_api-jalan\_raya2.avi | 100.00 | 0.00 | 0.00 |
| 44 | non\_api-jalan\_raya3.avi | 100.00 | 0.00 | 0.00 |
| 45 | non\_api-kantor.avi | 100.00 | 0.00 | 0.00 |
| 46 | non\_api-kecelakaan.avi | 100.00 | 0.00 | 0.00 |
| 47 | non\_api-kecelakaan2.avi | 100.00 | 0.00 | 0.00 |
| 48 | non\_api-kecelakaan3.avi | 100.00 | 0.00 | 0.00 |
| 49 | non\_api-kerusuhan.avi | 100.00 | 0.00 | 0.00 |
| 50 | non\_api-kerusuhan2.avi | 100.00 | 0.00 | 0.00 |
| 51 | non\_api-kerusuhan3.avi | 100.00 | 0.00 | 0.00 |
| 52 | non\_api-las\_vegas.avi | 96.91 | 3.09 | 0.00 |
| 53 | non\_api-manuju\_mobil.avi | 100.00 | 0.00 | 0.00 |
| 54 | non\_api-manuju\_mobil2.avi | 100.00 | 0.00 | 0.00 |
| 55 | non\_api-manuju\_mobil3.avi | 100.00 | 0.00 | 0.00 |
| 56 | non\_api-parkiran.avi | 100.00 | 0.00 | 0.00 |
| 57 | non\_api-parkiran2.avi | 100.00 | 0.00 | 0.00 |
| 58 | non\_api-parkiran3.avi | 100.00 | 0.00 | 0.00 |
| 59 | non\_api-pencuri.avi | 100.00 | 0.00 | 0.00 |
| 60 | non\_api-pencuri2.avi | 100.00 | 0.00 | 0.00 |
| 61 | non\_api-penembakan.avi | 100.00 | 0.00 | 0.00 |
| 62 | non\_api-serbet.avi | 78.45 | 21.55 | 0.00 |
| 63 | non\_api-tas.avi | 100.00 | 0.00 | 0.00 |
| 64 | non\_api-tas2.avi | 100.00 | 0.00 | 0.00 |
| 65 | non\_api-tas3.avi | 100.00 | 0.00 | 0.00 |
| 66 | non\_api-televisi.avi | 100.00 | 0.00 | 0.00 |
| 67 | non\_api-televisi2.avi | 100.00 | 0.00 | 0.00 |

Tabel A.2 Hasil Uji Coba Menggunakan Parameter *Threshold*  = 10-8, = 5 dan kernel RBF

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Nama File | True Positif | False Positif | Missing Rate |
| 1 | api-bakar\_sampah.avi | 89.06 | 0.00 | 10.94 |
| 2 | api-boneka\_dora.avi | 58.82 | 0.00 | 41.18 |
| 3 | api-kayu.avi | 88.73 | 0.00 | 11.27 |
| 4 | api-kayu2.avi | 100.00 | 0.00 | 0.00 |
| 5 | api-kebakaran\_barang.avi | 100.00 | 0.00 | 0.00 |
| 6 | api-kebakaran\_barang2.avi | 100.00 | 0.00 | 0.00 |
| 7 | api-kebakaran\_hutan.avi | 100.00 | 0.00 | 0.00 |
| 8 | api-kebakaran\_hutan2.avi | 100.00 | 0.00 | 0.00 |
| 9 | api-kebakaran\_hutan3.avi | 100.00 | 0.00 | 0.00 |
| 10 | api-kebakaran\_hutan4.avi | 100.00 | 0.00 | 0.00 |
| 11 | api-kebakaran\_ladang.avi | 100.00 | 0.00 | 0.00 |
| 12 | api-kebakaran\_mobil.avi | 100.00 | 0.00 | 0.00 |
| 13 | api-kebakaran\_mobil2.avi | 100.00 | 0.00 | 0.00 |
| 14 | api-kebakaran\_mobil3.avi | 100.00 | 0.00 | 0.00 |
| 15 | api-kebakaran\_mobil4.avi | 100.00 | 0.00 | 0.00 |
| 16 | api-kebakaran\_mobil5.avi | 100.00 | 0.00 | 0.00 |
| 17 | api-kebakaran\_mobil6.avi | 100.00 | 0.00 | 0.00 |
| 18 | api-kebakaran\_rumah.avi | 97.55 | 0.00 | 2.45 |
| 19 | api-kebakaran\_rumah2.avi | 100.00 | 0.00 | 0.00 |
| 20 | api-kebakaran\_rumah3.avi | 100.00 | 0.00 | 0.00 |
| 21 | api-kebakaran\_rumah4.avi | 71.57 | 0.00 | 28.43 |
| 22 | api-kebakaran\_tol.avi | 90.20 | 0.00 | 9.80 |
| 23 | api-kebakaran-truck.avi | 100.00 | 0.00 | 0.00 |
| 24 | api-kebakaran-truck2.avi | 100.00 | 0.00 | 0.00 |
| 25 | api-kebakaran-truck3.avi | 99.51 | 0.00 | 0.49 |
| 26 | api-kertas.avi | 59.88 | 0.00 | 40.12 |
| 27 | api-kertas2.avi | 72.22 | 0.00 | 27.78 |
| 28 | api-miniatur\_mainan.avi | 76.92 | 0.00 | 23.08 |
| 29 | api-mobil\_mainan.avi | 88.73 | 0.00 | 11.27 |
| 30 | api-orang\_terjun.avi | 74.51 | 0.00 | 25.49 |
| 31 | api-pesawat\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 32 | api-pesawat\_mainan2.avi | 100.00 | 0.00 | 0.00 |
| 33 | api-ruang\_tamu.avi | 51.48 | 0.00 | 48.52 |
| 34 | api-rumah\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 35 | non\_api-anak\_kecil.avi | 100.00 | 0.00 | 0.00 |
| 36 | non\_api-anak\_kecil2.avi | 100.00 | 0.00 | 0.00 |
| 37 | non\_api-anak\_kecil3.avi | 100.00 | 0.00 | 0.00 |
| 38 | non\_api-bertemu.avi | 100.00 | 0.00 | 0.00 |
| 39 | non\_api-jaket\_merah.avi | 100.00 | 0.00 | 0.00 |
| 40 | non\_api-jalan\_malam.avi | 100.00 | 0.00 | 0.00 |
| 41 | non\_api-jalan\_malam2.avi | 100.00 | 0.00 | 0.00 |
| 42 | non\_api-jalan\_raya.avi | 100.00 | 0.00 | 0.00 |
| 43 | non\_api-jalan\_raya2.avi | 100.00 | 0.00 | 0.00 |
| 44 | non\_api-jalan\_raya3.avi | 100.00 | 0.00 | 0.00 |
| 45 | non\_api-kantor.avi | 100.00 | 0.00 | 0.00 |
| 46 | non\_api-kecelakaan.avi | 100.00 | 0.00 | 0.00 |
| 47 | non\_api-kecelakaan2.avi | 100.00 | 0.00 | 0.00 |
| 48 | non\_api-kecelakaan3.avi | 100.00 | 0.00 | 0.00 |
| 49 | non\_api-kerusuhan.avi | 100.00 | 0.00 | 0.00 |
| 50 | non\_api-kerusuhan2.avi | 100.00 | 0.00 | 0.00 |
| 51 | non\_api-kerusuhan3.avi | 100.00 | 0.00 | 0.00 |
| 52 | non\_api-las\_vegas.avi | 51.23 | 48.77 | 0.00 |
| 53 | non\_api-manuju\_mobil.avi | 100.00 | 0.00 | 0.00 |
| 54 | non\_api-manuju\_mobil2.avi | 100.00 | 0.00 | 0.00 |
| 55 | non\_api-manuju\_mobil3.avi | 100.00 | 0.00 | 0.00 |
| 56 | non\_api-parkiran.avi | 100.00 | 0.00 | 0.00 |
| 57 | non\_api-parkiran2.avi | 100.00 | 0.00 | 0.00 |
| 58 | non\_api-parkiran3.avi | 100.00 | 0.00 | 0.00 |
| 59 | non\_api-pencuri.avi | 100.00 | 0.00 | 0.00 |
| 60 | non\_api-pencuri2.avi | 97.63 | 2.37 | 0.00 |
| 61 | non\_api-penembakan.avi | 100.00 | 0.00 | 0.00 |
| 62 | non\_api-serbet.avi | 69.06 | 30.94 | 0.00 |
| 63 | non\_api-tas.avi | 100.00 | 0.00 | 0.00 |
| 64 | non\_api-tas2.avi | 100.00 | 0.00 | 0.00 |
| 65 | non\_api-tas3.avi | 100.00 | 0.00 | 0.00 |
| 66 | non\_api-televisi.avi | 100.00 | 0.00 | 0.00 |
| 67 | non\_api-televisi2.avi | 100.00 | 0.00 | 0.00 |

Tabel A.3 Hasil Uji Coba Menggunakan Parameter *Threshold*  = 5x10-9, = 5 dan kernel RBF

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Nama File | True Positif | False Positif | Missing Rate |
| 1 | api-bakar\_sampah.avi | 93.75 | 0.00 | 6.25 |
| 2 | api-boneka\_dora.avi | 80.39 | 0.00 | 19.61 |
| 3 | api-kayu.avi | 93.63 | 0.00 | 6.37 |
| 4 | api-kayu2.avi | 100.00 | 0.00 | 0.00 |
| 5 | api-kebakaran\_barang.avi | 100.00 | 0.00 | 0.00 |
| 6 | api-kebakaran\_barang2.avi | 100.00 | 0.00 | 0.00 |
| 7 | api-kebakaran\_hutan.avi | 100.00 | 0.00 | 0.00 |
| 8 | api-kebakaran\_hutan2.avi | 100.00 | 0.00 | 0.00 |
| 9 | api-kebakaran\_hutan3.avi | 100.00 | 0.00 | 0.00 |
| 10 | api-kebakaran\_hutan4.avi | 100.00 | 0.00 | 0.00 |
| 11 | api-kebakaran\_ladang.avi | 100.00 | 0.00 | 0.00 |
| 12 | api-kebakaran\_mobil.avi | 100.00 | 0.00 | 0.00 |
| 13 | api-kebakaran\_mobil2.avi | 100.00 | 0.00 | 0.00 |
| 14 | api-kebakaran\_mobil3.avi | 100.00 | 0.00 | 0.00 |
| 15 | api-kebakaran\_mobil4.avi | 100.00 | 0.00 | 0.00 |
| 16 | api-kebakaran\_mobil5.avi | 100.00 | 0.00 | 0.00 |
| 17 | api-kebakaran\_mobil6.avi | 100.00 | 0.00 | 0.00 |
| 18 | api-kebakaran\_rumah.avi | 100.00 | 0.00 | 0.00 |
| 19 | api-kebakaran\_rumah2.avi | 100.00 | 0.00 | 0.00 |
| 20 | api-kebakaran\_rumah3.avi | 100.00 | 0.00 | 0.00 |
| 21 | api-kebakaran\_rumah4.avi | 79.90 | 0.00 | 20.10 |
| 22 | api-kebakaran\_tol.avi | 90.69 | 0.00 | 9.31 |
| 23 | api-kebakaran-truck.avi | 100.00 | 0.00 | 0.00 |
| 24 | api-kebakaran-truck2.avi | 100.00 | 0.00 | 0.00 |
| 25 | api-kebakaran-truck3.avi | 100.00 | 0.00 | 0.00 |
| 26 | api-kertas.avi | 67.28 | 0.00 | 32.72 |
| 27 | api-kertas2.avi | 98.77 | 0.00 | 1.23 |
| 28 | api-miniatur\_mainan.avi | 89.35 | 0.00 | 10.65 |
| 29 | api-mobil\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 30 | api-orang\_terjun.avi | 79.41 | 0.00 | 20.59 |
| 31 | api-pesawat\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 32 | api-pesawat\_mainan2.avi | 100.00 | 0.00 | 0.00 |
| 33 | api-ruang\_tamu.avi | 77.51 | 0.00 | 22.49 |
| 34 | api-rumah\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 35 | non\_api-anak\_kecil.avi | 100.00 | 0.00 | 0.00 |
| 36 | non\_api-anak\_kecil2.avi | 100.00 | 0.00 | 0.00 |
| 37 | non\_api-anak\_kecil3.avi | 100.00 | 0.00 | 0.00 |
| 38 | non\_api-bertemu.avi | 100.00 | 0.00 | 0.00 |
| 39 | non\_api-jaket\_merah.avi | 100.00 | 0.00 | 0.00 |
| 40 | non\_api-jalan\_malam.avi | 100.00 | 0.00 | 0.00 |
| 41 | non\_api-jalan\_malam2.avi | 100.00 | 0.00 | 0.00 |
| 42 | non\_api-jalan\_raya.avi | 100.00 | 0.00 | 0.00 |
| 43 | non\_api-jalan\_raya2.avi | 100.00 | 0.00 | 0.00 |
| 44 | non\_api-jalan\_raya3.avi | 100.00 | 0.00 | 0.00 |
| 45 | non\_api-kantor.avi | 100.00 | 0.00 | 0.00 |
| 46 | non\_api-kecelakaan.avi | 100.00 | 0.00 | 0.00 |
| 47 | non\_api-kecelakaan2.avi | 100.00 | 0.00 | 0.00 |
| 48 | non\_api-kecelakaan3.avi | 100.00 | 0.00 | 0.00 |
| 49 | non\_api-kerusuhan.avi | 100.00 | 0.00 | 0.00 |
| 50 | non\_api-kerusuhan2.avi | 100.00 | 0.00 | 0.00 |
| 51 | non\_api-kerusuhan3.avi | 100.00 | 0.00 | 0.00 |
| 52 | non\_api-las\_vegas.avi | 46.30 | 53.70 | 0.00 |
| 53 | non\_api-manuju\_mobil.avi | 100.00 | 0.00 | 0.00 |
| 54 | non\_api-manuju\_mobil2.avi | 100.00 | 0.00 | 0.00 |
| 55 | non\_api-manuju\_mobil3.avi | 100.00 | 0.00 | 0.00 |
| 56 | non\_api-parkiran.avi | 100.00 | 0.00 | 0.00 |
| 57 | non\_api-parkiran2.avi | 100.00 | 0.00 | 0.00 |
| 58 | non\_api-parkiran3.avi | 100.00 | 0.00 | 0.00 |
| 59 | non\_api-pencuri.avi | 92.31 | 7.69 | 0.00 |
| 60 | non\_api-pencuri2.avi | 97.04 | 2.96 | 0.00 |
| 61 | non\_api-penembakan.avi | 100.00 | 0.00 | 0.00 |
| 62 | non\_api-serbet.avi | 66.85 | 33.15 | 0.00 |
| 63 | non\_api-tas.avi | 100.00 | 0.00 | 0.00 |
| 64 | non\_api-tas2.avi | 100.00 | 0.00 | 0.00 |
| 65 | non\_api-tas3.avi | 100.00 | 0.00 | 0.00 |
| 66 | non\_api-televisi.avi | 100.00 | 0.00 | 0.00 |
| 67 | non\_api-televisi2.avi | 100.00 | 0.00 | 0.00 |

Tabel A.4 Hasil Uji Coba Menggunakan Parameter *Threshold*  = 10-9, = 5 dan kernel RBF

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Nama File | True Positif | False Positif | Missing Rate |
| 1 | api-bakar\_sampah.avi | 100.00 | 0.00 | 0.00 |
| 2 | api-boneka\_dora.avi | 99.02 | 0.00 | 0.98 |
| 3 | api-kayu.avi | 100.00 | 0.00 | 0.00 |
| 4 | api-kayu2.avi | 100.00 | 0.00 | 0.00 |
| 5 | api-kebakaran\_barang.avi | 100.00 | 0.00 | 0.00 |
| 6 | api-kebakaran\_barang2.avi | 100.00 | 0.00 | 0.00 |
| 7 | api-kebakaran\_hutan.avi | 100.00 | 0.00 | 0.00 |
| 8 | api-kebakaran\_hutan2.avi | 100.00 | 0.00 | 0.00 |
| 9 | api-kebakaran\_hutan3.avi | 100.00 | 0.00 | 0.00 |
| 10 | api-kebakaran\_hutan4.avi | 100.00 | 0.00 | 0.00 |
| 11 | api-kebakaran\_ladang.avi | 100.00 | 0.00 | 0.00 |
| 12 | api-kebakaran\_mobil.avi | 100.00 | 0.00 | 0.00 |
| 13 | api-kebakaran\_mobil2.avi | 100.00 | 0.00 | 0.00 |
| 14 | api-kebakaran\_mobil3.avi | 100.00 | 0.00 | 0.00 |
| 15 | api-kebakaran\_mobil4.avi | 100.00 | 0.00 | 0.00 |
| 16 | api-kebakaran\_mobil5.avi | 100.00 | 0.00 | 0.00 |
| 17 | api-kebakaran\_mobil6.avi | 100.00 | 0.00 | 0.00 |
| 18 | api-kebakaran\_rumah.avi | 100.00 | 0.00 | 0.00 |
| 19 | api-kebakaran\_rumah2.avi | 100.00 | 0.00 | 0.00 |
| 20 | api-kebakaran\_rumah3.avi | 100.00 | 0.00 | 0.00 |
| 21 | api-kebakaran\_rumah4.avi | 100.00 | 0.00 | 0.00 |
| 22 | api-kebakaran\_tol.avi | 99.02 | 0.00 | 0.98 |
| 23 | api-kebakaran-truck.avi | 100.00 | 0.00 | 0.00 |
| 24 | api-kebakaran-truck2.avi | 100.00 | 0.00 | 0.00 |
| 25 | api-kebakaran-truck3.avi | 100.00 | 0.00 | 0.00 |
| 26 | api-kertas.avi | 96.30 | 0.00 | 3.70 |
| 27 | api-kertas2.avi | 100.00 | 0.00 | 0.00 |
| 28 | api-miniatur\_mainan.avi | 99.41 | 0.00 | 0.59 |
| 29 | api-mobil\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 30 | api-orang\_terjun.avi | 94.61 | 0.00 | 5.39 |
| 31 | api-pesawat\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 32 | api-pesawat\_mainan2.avi | 100.00 | 0.00 | 0.00 |
| 33 | api-ruang\_tamu.avi | 98.22 | 0.00 | 1.78 |
| 34 | api-rumah\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 35 | non\_api-anak\_kecil.avi | 100.00 | 0.00 | 0.00 |
| 36 | non\_api-anak\_kecil2.avi | 100.00 | 0.00 | 0.00 |
| 37 | non\_api-anak\_kecil3.avi | 100.00 | 0.00 | 0.00 |
| 38 | non\_api-bertemu.avi | 100.00 | 0.00 | 0.00 |
| 39 | non\_api-jaket\_merah.avi | 100.00 | 0.00 | 0.00 |
| 40 | non\_api-jalan\_malam.avi | 100.00 | 0.00 | 0.00 |
| 41 | non\_api-jalan\_malam2.avi | 100.00 | 0.00 | 0.00 |
| 42 | non\_api-jalan\_raya.avi | 5.33 | 94.67 | 0.00 |
| 43 | non\_api-jalan\_raya2.avi | 87.57 | 12.43 | 0.00 |
| 44 | non\_api-jalan\_raya3.avi | 11.24 | 88.76 | 0.00 |
| 45 | non\_api-kantor.avi | 100.00 | 0.00 | 0.00 |
| 46 | non\_api-kecelakaan.avi | 100.00 | 0.00 | 0.00 |
| 47 | non\_api-kecelakaan2.avi | 100.00 | 0.00 | 0.00 |
| 48 | non\_api-kecelakaan3.avi | 100.00 | 0.00 | 0.00 |
| 49 | non\_api-kerusuhan.avi | 25.44 | 74.56 | 0.00 |
| 50 | non\_api-kerusuhan2.avi | 81.07 | 18.93 | 0.00 |
| 51 | non\_api-kerusuhan3.avi | 100.00 | 0.00 | 0.00 |
| 52 | non\_api-las\_vegas.avi | 43.21 | 56.79 | 0.00 |
| 53 | non\_api-manuju\_mobil.avi | 100.00 | 0.00 | 0.00 |
| 54 | non\_api-manuju\_mobil2.avi | 100.00 | 0.00 | 0.00 |
| 55 | non\_api-manuju\_mobil3.avi | 100.00 | 0.00 | 0.00 |
| 56 | non\_api-parkiran.avi | 100.00 | 0.00 | 0.00 |
| 57 | non\_api-parkiran2.avi | 100.00 | 0.00 | 0.00 |
| 58 | non\_api-parkiran3.avi | 100.00 | 0.00 | 0.00 |
| 59 | non\_api-pencuri.avi | 8.28 | 91.72 | 0.00 |
| 60 | non\_api-pencuri2.avi | 93.49 | 6.51 | 0.00 |
| 61 | non\_api-penembakan.avi | 97.06 | 2.94 | 0.00 |
| 62 | non\_api-serbet.avi | 21.55 | 78.45 | 0.00 |
| 63 | non\_api-tas.avi | 100.00 | 0.00 | 0.00 |
| 64 | non\_api-tas2.avi | 100.00 | 0.00 | 0.00 |
| 65 | non\_api-tas3.avi | 100.00 | 0.00 | 0.00 |
| 66 | non\_api-televisi.avi | 100.00 | 0.00 | 0.00 |
| 67 | non\_api-televisi2.avi | 100.00 | 0.00 | 0.00 |

Tabel A.5 Hasil Uji Coba Menggunakan Parameter *Threshold*  = 5x10-9, = 1 dan kernel RBF

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Nama File | True Positif | False Positif | Missing Rate |
| 1 | api-bakar\_sampah.avi | 92.19 | 0.00 | 7.81 |
| 2 | api-boneka\_dora.avi | 80.39 | 0.00 | 19.61 |
| 3 | api-kayu.avi | 93.63 | 0.00 | 6.37 |
| 4 | api-kayu2.avi | 100.00 | 0.00 | 0.00 |
| 5 | api-kebakaran\_barang.avi | 100.00 | 0.00 | 0.00 |
| 6 | api-kebakaran\_barang2.avi | 100.00 | 0.00 | 0.00 |
| 7 | api-kebakaran\_hutan.avi | 100.00 | 0.00 | 0.00 |
| 8 | api-kebakaran\_hutan2.avi | 100.00 | 0.00 | 0.00 |
| 9 | api-kebakaran\_hutan3.avi | 100.00 | 0.00 | 0.00 |
| 10 | api-kebakaran\_hutan4.avi | 100.00 | 0.00 | 0.00 |
| 11 | api-kebakaran\_ladang.avi | 100.00 | 0.00 | 0.00 |
| 12 | api-kebakaran\_mobil.avi | 100.00 | 0.00 | 0.00 |
| 13 | api-kebakaran\_mobil2.avi | 100.00 | 0.00 | 0.00 |
| 14 | api-kebakaran\_mobil3.avi | 100.00 | 0.00 | 0.00 |
| 15 | api-kebakaran\_mobil4.avi | 100.00 | 0.00 | 0.00 |
| 16 | api-kebakaran\_mobil5.avi | 100.00 | 0.00 | 0.00 |
| 17 | api-kebakaran\_mobil6.avi | 100.00 | 0.00 | 0.00 |
| 18 | api-kebakaran\_rumah.avi | 100.00 | 0.00 | 0.00 |
| 19 | api-kebakaran\_rumah2.avi | 100.00 | 0.00 | 0.00 |
| 20 | api-kebakaran\_rumah3.avi | 100.00 | 0.00 | 0.00 |
| 21 | api-kebakaran\_rumah4.avi | 78.92 | 0.00 | 21.08 |
| 22 | api-kebakaran\_tol.avi | 90.69 | 0.00 | 9.31 |
| 23 | api-kebakaran-truck.avi | 100.00 | 0.00 | 0.00 |
| 24 | api-kebakaran-truck2.avi | 100.00 | 0.00 | 0.00 |
| 25 | api-kebakaran-truck3.avi | 100.00 | 0.00 | 0.00 |
| 26 | api-kertas.avi | 67.28 | 0.00 | 32.72 |
| 27 | api-kertas2.avi | 98.77 | 0.00 | 1.23 |
| 28 | api-miniatur\_mainan.avi | 89.35 | 0.00 | 10.65 |
| 29 | api-mobil\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 30 | api-orang\_terjun.avi | 79.41 | 0.00 | 20.59 |
| 31 | api-pesawat\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 32 | api-pesawat\_mainan2.avi | 100.00 | 0.00 | 0.00 |
| 33 | api-ruang\_tamu.avi | 77.51 | 0.00 | 22.49 |
| 34 | api-rumah\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 35 | non\_api-anak\_kecil.avi | 100.00 | 0.00 | 0.00 |
| 36 | non\_api-anak\_kecil2.avi | 100.00 | 0.00 | 0.00 |
| 37 | non\_api-anak\_kecil3.avi | 100.00 | 0.00 | 0.00 |
| 38 | non\_api-bertemu.avi | 100.00 | 0.00 | 0.00 |
| 39 | non\_api-jaket\_merah.avi | 100.00 | 0.00 | 0.00 |
| 40 | non\_api-jalan\_malam.avi | 100.00 | 0.00 | 0.00 |
| 41 | non\_api-jalan\_malam2.avi | 100.00 | 0.00 | 0.00 |
| 42 | non\_api-jalan\_raya.avi | 100.00 | 0.00 | 0.00 |
| 43 | non\_api-jalan\_raya2.avi | 100.00 | 0.00 | 0.00 |
| 44 | non\_api-jalan\_raya3.avi | 100.00 | 0.00 | 0.00 |
| 45 | non\_api-kantor.avi | 100.00 | 0.00 | 0.00 |
| 46 | non\_api-kecelakaan.avi | 100.00 | 0.00 | 0.00 |
| 47 | non\_api-kecelakaan2.avi | 100.00 | 0.00 | 0.00 |
| 48 | non\_api-kecelakaan3.avi | 100.00 | 0.00 | 0.00 |
| 49 | non\_api-kerusuhan.avi | 100.00 | 0.00 | 0.00 |
| 50 | non\_api-kerusuhan2.avi | 100.00 | 0.00 | 0.00 |
| 51 | non\_api-kerusuhan3.avi | 100.00 | 0.00 | 0.00 |
| 52 | non\_api-las\_vegas.avi | 46.30 | 53.70 | 0.00 |
| 53 | non\_api-manuju\_mobil.avi | 100.00 | 0.00 | 0.00 |
| 54 | non\_api-manuju\_mobil2.avi | 100.00 | 0.00 | 0.00 |
| 55 | non\_api-manuju\_mobil3.avi | 100.00 | 0.00 | 0.00 |
| 56 | non\_api-parkiran.avi | 100.00 | 0.00 | 0.00 |
| 57 | non\_api-parkiran2.avi | 100.00 | 0.00 | 0.00 |
| 58 | non\_api-parkiran3.avi | 100.00 | 0.00 | 0.00 |
| 59 | non\_api-pencuri.avi | 92.31 | 7.69 | 0.00 |
| 60 | non\_api-pencuri2.avi | 97.04 | 2.96 | 0.00 |
| 61 | non\_api-penembakan.avi | 100.00 | 0.00 | 0.00 |
| 62 | non\_api-serbet.avi | 68.51 | 31.49 | 0.00 |
| 63 | non\_api-tas.avi | 100.00 | 0.00 | 0.00 |
| 64 | non\_api-tas2.avi | 100.00 | 0.00 | 0.00 |
| 65 | non\_api-tas3.avi | 100.00 | 0.00 | 0.00 |
| 66 | non\_api-televisi.avi | 100.00 | 0.00 | 0.00 |
| 67 | non\_api-televisi2.avi | 100.00 | 0.00 | 0.00 |

Tabel A.6 Hasil Uji Coba Menggunakan Parameter *Threshold*  = 5x10-9, = 3.5 dan kernel RBF

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Nama File | True Positif | False Positif | Missing Rate |
| 1 | api-bakar\_sampah.avi | 93.75 | 0.00 | 6.25 |
| 2 | api-boneka\_dora.avi | 80.39 | 0.00 | 19.61 |
| 3 | api-kayu.avi | 93.63 | 0.00 | 6.37 |
| 4 | api-kayu2.avi | 100.00 | 0.00 | 0.00 |
| 5 | api-kebakaran\_barang.avi | 100.00 | 0.00 | 0.00 |
| 6 | api-kebakaran\_barang2.avi | 100.00 | 0.00 | 0.00 |
| 7 | api-kebakaran\_hutan.avi | 100.00 | 0.00 | 0.00 |
| 8 | api-kebakaran\_hutan2.avi | 100.00 | 0.00 | 0.00 |
| 9 | api-kebakaran\_hutan3.avi | 100.00 | 0.00 | 0.00 |
| 10 | api-kebakaran\_hutan4.avi | 100.00 | 0.00 | 0.00 |
| 11 | api-kebakaran\_ladang.avi | 100.00 | 0.00 | 0.00 |
| 12 | api-kebakaran\_mobil.avi | 100.00 | 0.00 | 0.00 |
| 13 | api-kebakaran\_mobil2.avi | 100.00 | 0.00 | 0.00 |
| 14 | api-kebakaran\_mobil3.avi | 100.00 | 0.00 | 0.00 |
| 15 | api-kebakaran\_mobil4.avi | 100.00 | 0.00 | 0.00 |
| 16 | api-kebakaran\_mobil5.avi | 100.00 | 0.00 | 0.00 |
| 17 | api-kebakaran\_mobil6.avi | 100.00 | 0.00 | 0.00 |
| 18 | api-kebakaran\_rumah.avi | 100.00 | 0.00 | 0.00 |
| 19 | api-kebakaran\_rumah2.avi | 100.00 | 0.00 | 0.00 |
| 20 | api-kebakaran\_rumah3.avi | 100.00 | 0.00 | 0.00 |
| 21 | api-kebakaran\_rumah4.avi | 79.90 | 0.00 | 20.10 |
| 22 | api-kebakaran\_tol.avi | 90.69 | 0.00 | 9.31 |
| 23 | api-kebakaran-truck.avi | 100.00 | 0.00 | 0.00 |
| 24 | api-kebakaran-truck2.avi | 100.00 | 0.00 | 0.00 |
| 25 | api-kebakaran-truck3.avi | 100.00 | 0.00 | 0.00 |
| 26 | api-kertas.avi | 67.28 | 0.00 | 32.72 |
| 27 | api-kertas2.avi | 98.77 | 0.00 | 1.23 |
| 28 | api-miniatur\_mainan.avi | 89.35 | 0.00 | 10.65 |
| 29 | api-mobil\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 30 | api-orang\_terjun.avi | 79.41 | 0.00 | 20.59 |
| 31 | api-pesawat\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 32 | api-pesawat\_mainan2.avi | 100.00 | 0.00 | 0.00 |
| 33 | api-ruang\_tamu.avi | 77.51 | 0.00 | 22.49 |
| 34 | api-rumah\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 35 | non\_api-anak\_kecil.avi | 100.00 | 0.00 | 0.00 |
| 36 | non\_api-anak\_kecil2.avi | 100.00 | 0.00 | 0.00 |
| 37 | non\_api-anak\_kecil3.avi | 100.00 | 0.00 | 0.00 |
| 38 | non\_api-bertemu.avi | 100.00 | 0.00 | 0.00 |
| 39 | non\_api-jaket\_merah.avi | 100.00 | 0.00 | 0.00 |
| 40 | non\_api-jalan\_malam.avi | 100.00 | 0.00 | 0.00 |
| 41 | non\_api-jalan\_malam2.avi | 100.00 | 0.00 | 0.00 |
| 42 | non\_api-jalan\_raya.avi | 100.00 | 0.00 | 0.00 |
| 43 | non\_api-jalan\_raya2.avi | 100.00 | 0.00 | 0.00 |
| 44 | non\_api-jalan\_raya3.avi | 100.00 | 0.00 | 0.00 |
| 45 | non\_api-kantor.avi | 100.00 | 0.00 | 0.00 |
| 46 | non\_api-kecelakaan.avi | 100.00 | 0.00 | 0.00 |
| 47 | non\_api-kecelakaan2.avi | 100.00 | 0.00 | 0.00 |
| 48 | non\_api-kecelakaan3.avi | 100.00 | 0.00 | 0.00 |
| 49 | non\_api-kerusuhan.avi | 100.00 | 0.00 | 0.00 |
| 50 | non\_api-kerusuhan2.avi | 100.00 | 0.00 | 0.00 |
| 51 | non\_api-kerusuhan3.avi | 100.00 | 0.00 | 0.00 |
| 52 | non\_api-las\_vegas.avi | 46.30 | 53.70 | 0.00 |
| 53 | non\_api-manuju\_mobil.avi | 100.00 | 0.00 | 0.00 |
| 54 | non\_api-manuju\_mobil2.avi | 100.00 | 0.00 | 0.00 |
| 55 | non\_api-manuju\_mobil3.avi | 100.00 | 0.00 | 0.00 |
| 56 | non\_api-parkiran.avi | 100.00 | 0.00 | 0.00 |
| 57 | non\_api-parkiran2.avi | 100.00 | 0.00 | 0.00 |
| 58 | non\_api-parkiran3.avi | 100.00 | 0.00 | 0.00 |
| 59 | non\_api-pencuri.avi | 92.31 | 7.69 | 0.00 |
| 60 | non\_api-pencuri2.avi | 97.04 | 2.96 | 0.00 |
| 61 | non\_api-penembakan.avi | 100.00 | 0.00 | 0.00 |
| 62 | non\_api-serbet.avi | 66.85 | 33.15 | 0.00 |
| 63 | non\_api-tas.avi | 100.00 | 0.00 | 0.00 |
| 64 | non\_api-tas2.avi | 100.00 | 0.00 | 0.00 |
| 65 | non\_api-tas3.avi | 100.00 | 0.00 | 0.00 |
| 66 | non\_api-televisi.avi | 100.00 | 0.00 | 0.00 |
| 67 | non\_api-televisi2.avi | 100.00 | 0.00 | 0.00 |

Tabel A.7 Hasil Uji Coba Menggunakan Parameter *Threshold*  = 5x10-9, = 7 dan kernel RBF

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Nama File | True Positif | False Positif | Missing Rate |
| 1 | api-bakar\_sampah.avi | 93.75 | 0.00 | 6.25 |
| 2 | api-boneka\_dora.avi | 79.90 | 0.00 | 20.10 |
| 3 | api-kayu.avi | 93.63 | 0.00 | 6.37 |
| 4 | api-kayu2.avi | 100.00 | 0.00 | 0.00 |
| 5 | api-kebakaran\_barang.avi | 100.00 | 0.00 | 0.00 |
| 6 | api-kebakaran\_barang2.avi | 100.00 | 0.00 | 0.00 |
| 7 | api-kebakaran\_hutan.avi | 100.00 | 0.00 | 0.00 |
| 8 | api-kebakaran\_hutan2.avi | 100.00 | 0.00 | 0.00 |
| 9 | api-kebakaran\_hutan3.avi | 100.00 | 0.00 | 0.00 |
| 10 | api-kebakaran\_hutan4.avi | 100.00 | 0.00 | 0.00 |
| 11 | api-kebakaran\_ladang.avi | 100.00 | 0.00 | 0.00 |
| 12 | api-kebakaran\_mobil.avi | 100.00 | 0.00 | 0.00 |
| 13 | api-kebakaran\_mobil2.avi | 100.00 | 0.00 | 0.00 |
| 14 | api-kebakaran\_mobil3.avi | 100.00 | 0.00 | 0.00 |
| 15 | api-kebakaran\_mobil4.avi | 100.00 | 0.00 | 0.00 |
| 16 | api-kebakaran\_mobil5.avi | 100.00 | 0.00 | 0.00 |
| 17 | api-kebakaran\_mobil6.avi | 100.00 | 0.00 | 0.00 |
| 18 | api-kebakaran\_rumah.avi | 100.00 | 0.00 | 0.00 |
| 19 | api-kebakaran\_rumah2.avi | 100.00 | 0.00 | 0.00 |
| 20 | api-kebakaran\_rumah3.avi | 100.00 | 0.00 | 0.00 |
| 21 | api-kebakaran\_rumah4.avi | 80.39 | 0.00 | 19.61 |
| 22 | api-kebakaran\_tol.avi | 90.69 | 0.00 | 9.31 |
| 23 | api-kebakaran-truck.avi | 100.00 | 0.00 | 0.00 |
| 24 | api-kebakaran-truck2.avi | 100.00 | 0.00 | 0.00 |
| 25 | api-kebakaran-truck3.avi | 100.00 | 0.00 | 0.00 |
| 26 | api-kertas.avi | 67.28 | 0.00 | 32.72 |
| 27 | api-kertas2.avi | 98.77 | 0.00 | 1.23 |
| 28 | api-miniatur\_mainan.avi | 89.35 | 0.00 | 10.65 |
| 29 | api-mobil\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 30 | api-orang\_terjun.avi | 79.41 | 0.00 | 20.59 |
| 31 | api-pesawat\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 32 | api-pesawat\_mainan2.avi | 100.00 | 0.00 | 0.00 |
| 33 | api-ruang\_tamu.avi | 77.51 | 0.00 | 22.49 |
| 34 | api-rumah\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 35 | non\_api-anak\_kecil.avi | 100.00 | 0.00 | 0.00 |
| 36 | non\_api-anak\_kecil2.avi | 100.00 | 0.00 | 0.00 |
| 37 | non\_api-anak\_kecil3.avi | 100.00 | 0.00 | 0.00 |
| 38 | non\_api-bertemu.avi | 100.00 | 0.00 | 0.00 |
| 39 | non\_api-jaket\_merah.avi | 100.00 | 0.00 | 0.00 |
| 40 | non\_api-jalan\_malam.avi | 100.00 | 0.00 | 0.00 |
| 41 | non\_api-jalan\_malam2.avi | 100.00 | 0.00 | 0.00 |
| 42 | non\_api-jalan\_raya.avi | 100.00 | 0.00 | 0.00 |
| 43 | non\_api-jalan\_raya2.avi | 100.00 | 0.00 | 0.00 |
| 44 | non\_api-jalan\_raya3.avi | 100.00 | 0.00 | 0.00 |
| 45 | non\_api-kantor.avi | 100.00 | 0.00 | 0.00 |
| 46 | non\_api-kecelakaan.avi | 100.00 | 0.00 | 0.00 |
| 47 | non\_api-kecelakaan2.avi | 100.00 | 0.00 | 0.00 |
| 48 | non\_api-kecelakaan3.avi | 100.00 | 0.00 | 0.00 |
| 49 | non\_api-kerusuhan.avi | 100.00 | 0.00 | 0.00 |
| 50 | non\_api-kerusuhan2.avi | 100.00 | 0.00 | 0.00 |
| 51 | non\_api-kerusuhan3.avi | 100.00 | 0.00 | 0.00 |
| 52 | non\_api-las\_vegas.avi | 46.30 | 53.70 | 0.00 |
| 53 | non\_api-manuju\_mobil.avi | 100.00 | 0.00 | 0.00 |
| 54 | non\_api-manuju\_mobil2.avi | 100.00 | 0.00 | 0.00 |
| 55 | non\_api-manuju\_mobil3.avi | 100.00 | 0.00 | 0.00 |
| 56 | non\_api-parkiran.avi | 100.00 | 0.00 | 0.00 |
| 57 | non\_api-parkiran2.avi | 100.00 | 0.00 | 0.00 |
| 58 | non\_api-parkiran3.avi | 100.00 | 0.00 | 0.00 |
| 59 | non\_api-pencuri.avi | 92.31 | 7.69 | 0.00 |
| 60 | non\_api-pencuri2.avi | 97.04 | 2.96 | 0.00 |
| 61 | non\_api-penembakan.avi | 100.00 | 0.00 | 0.00 |
| 62 | non\_api-serbet.avi | 66.85 | 33.15 | 0.00 |
| 63 | non\_api-tas.avi | 100.00 | 0.00 | 0.00 |
| 64 | non\_api-tas2.avi | 100.00 | 0.00 | 0.00 |
| 65 | non\_api-tas3.avi | 100.00 | 0.00 | 0.00 |
| 66 | non\_api-televisi.avi | 100.00 | 0.00 | 0.00 |
| 67 | non\_api-televisi2.avi | 100.00 | 0.00 | 0.00 |

Tabel A.8 Hasil Uji Coba Menggunakan Parameter *Threshold*  = 5x10-9, = 5 dan kernel *Polynomial 2*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Nama File | True Positif | False Positif | Missing Rate |
| 1 | api-bakar\_sampah.avi | 12.50 | 0.00 | 87.50 |
| 2 | api-boneka\_dora.avi | 58.33 | 0.00 | 41.67 |
| 3 | api-kayu.avi | 90.69 | 0.00 | 9.31 |
| 4 | api-kayu2.avi | 98.04 | 0.00 | 1.96 |
| 5 | api-kebakaran\_barang.avi | 68.14 | 0.00 | 31.86 |
| 6 | api-kebakaran\_barang2.avi | 88.73 | 0.00 | 11.27 |
| 7 | api-kebakaran\_hutan.avi | 96.97 | 0.00 | 3.03 |
| 8 | api-kebakaran\_hutan2.avi | 100.00 | 0.00 | 0.00 |
| 9 | api-kebakaran\_hutan3.avi | 100.00 | 0.00 | 0.00 |
| 10 | api-kebakaran\_hutan4.avi | 100.00 | 0.00 | 0.00 |
| 11 | api-kebakaran\_ladang.avi | 100.00 | 0.00 | 0.00 |
| 12 | api-kebakaran\_mobil.avi | 100.00 | 0.00 | 0.00 |
| 13 | api-kebakaran\_mobil2.avi | 53.43 | 0.00 | 46.57 |
| 14 | api-kebakaran\_mobil3.avi | 50.00 | 0.00 | 50.00 |
| 15 | api-kebakaran\_mobil4.avi | 94.61 | 0.00 | 5.39 |
| 16 | api-kebakaran\_mobil5.avi | 100.00 | 0.00 | 0.00 |
| 17 | api-kebakaran\_mobil6.avi | 100.00 | 0.00 | 0.00 |
| 18 | api-kebakaran\_rumah.avi | 50.00 | 0.00 | 50.00 |
| 19 | api-kebakaran\_rumah2.avi | 41.67 | 0.00 | 58.33 |
| 20 | api-kebakaran\_rumah3.avi | 6.86 | 0.00 | 93.14 |
| 21 | api-kebakaran\_rumah4.avi | 65.20 | 0.00 | 34.80 |
| 22 | api-kebakaran\_tol.avi | 47.55 | 0.00 | 52.45 |
| 23 | api-kebakaran-truck.avi | 100.00 | 0.00 | 0.00 |
| 24 | api-kebakaran-truck2.avi | 100.00 | 0.00 | 0.00 |
| 25 | api-kebakaran-truck3.avi | 99.51 | 0.00 | 0.49 |
| 26 | api-kertas.avi | 64.20 | 0.00 | 35.80 |
| 27 | api-kertas2.avi | 98.77 | 0.00 | 1.23 |
| 28 | api-miniatur\_mainan.avi | 72.19 | 0.00 | 27.81 |
| 29 | api-mobil\_mainan.avi | 99.02 | 0.00 | 0.98 |
| 30 | api-orang\_terjun.avi | 77.45 | 0.00 | 22.55 |
| 31 | api-pesawat\_mainan.avi | 92.36 | 0.00 | 7.64 |
| 32 | api-pesawat\_mainan2.avi | 98.82 | 0.00 | 1.18 |
| 33 | api-ruang\_tamu.avi | 30.18 | 0.00 | 69.82 |
| 34 | api-rumah\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 35 | non\_api-anak\_kecil.avi | 100.00 | 0.00 | 0.00 |
| 36 | non\_api-anak\_kecil2.avi | 100.00 | 0.00 | 0.00 |
| 37 | non\_api-anak\_kecil3.avi | 100.00 | 0.00 | 0.00 |
| 38 | non\_api-bertemu.avi | 100.00 | 0.00 | 0.00 |
| 39 | non\_api-jaket\_merah.avi | 100.00 | 0.00 | 0.00 |
| 40 | non\_api-jalan\_malam.avi | 100.00 | 0.00 | 0.00 |
| 41 | non\_api-jalan\_malam2.avi | 100.00 | 0.00 | 0.00 |
| 42 | non\_api-jalan\_raya.avi | 100.00 | 0.00 | 0.00 |
| 43 | non\_api-jalan\_raya2.avi | 100.00 | 0.00 | 0.00 |
| 44 | non\_api-jalan\_raya3.avi | 100.00 | 0.00 | 0.00 |
| 45 | non\_api-kantor.avi | 100.00 | 0.00 | 0.00 |
| 46 | non\_api-kecelakaan.avi | 100.00 | 0.00 | 0.00 |
| 47 | non\_api-kecelakaan2.avi | 100.00 | 0.00 | 0.00 |
| 48 | non\_api-kecelakaan3.avi | 100.00 | 0.00 | 0.00 |
| 49 | non\_api-kerusuhan.avi | 100.00 | 0.00 | 0.00 |
| 50 | non\_api-kerusuhan2.avi | 100.00 | 0.00 | 0.00 |
| 51 | non\_api-kerusuhan3.avi | 100.00 | 0.00 | 0.00 |
| 52 | non\_api-las\_vegas.avi | 48.15 | 51.85 | 0.00 |
| 53 | non\_api-manuju\_mobil.avi | 100.00 | 0.00 | 0.00 |
| 54 | non\_api-manuju\_mobil2.avi | 100.00 | 0.00 | 0.00 |
| 55 | non\_api-manuju\_mobil3.avi | 100.00 | 0.00 | 0.00 |
| 56 | non\_api-parkiran.avi | 100.00 | 0.00 | 0.00 |
| 57 | non\_api-parkiran2.avi | 100.00 | 0.00 | 0.00 |
| 58 | non\_api-parkiran3.avi | 100.00 | 0.00 | 0.00 |
| 59 | non\_api-pencuri.avi | 95.86 | 4.14 | 0.00 |
| 60 | non\_api-pencuri2.avi | 100.00 | 0.00 | 0.00 |
| 61 | non\_api-penembakan.avi | 100.00 | 0.00 | 0.00 |
| 62 | non\_api-serbet.avi | 92.27 | 7.73 | 0.00 |
| 63 | non\_api-tas.avi | 100.00 | 0.00 | 0.00 |
| 64 | non\_api-tas2.avi | 100.00 | 0.00 | 0.00 |
| 65 | non\_api-tas3.avi | 100.00 | 0.00 | 0.00 |
| 66 | non\_api-televisi.avi | 100.00 | 0.00 | 0.00 |
| 67 | non\_api-televisi2.avi | 100.00 | 0.00 | 0.00 |

Tabel A.9 Hasil Uji Coba Menggunakan Parameter *Threshold*  = 5x10-9, = 5 dan kernel *Polynomial 3*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Nama File | True Positif | False Positif | Missing Rate |
| 1 | api-bakar\_sampah.avi | 0.00 | 0.00 | 100.00 |
| 2 | api-boneka\_dora.avi | 1.47 | 0.00 | 98.53 |
| 3 | api-kayu.avi | 8.33 | 0.00 | 91.67 |
| 4 | api-kayu2.avi | 10.78 | 0.00 | 89.22 |
| 5 | api-kebakaran\_barang.avi | 0.00 | 0.00 | 100.00 |
| 6 | api-kebakaran\_barang2.avi | 0.00 | 0.00 | 100.00 |
| 7 | api-kebakaran\_hutan.avi | 20.20 | 0.00 | 79.80 |
| 8 | api-kebakaran\_hutan2.avi | 86.87 | 0.00 | 13.13 |
| 9 | api-kebakaran\_hutan3.avi | 32.32 | 0.00 | 67.68 |
| 10 | api-kebakaran\_hutan4.avi | 25.00 | 0.00 | 75.00 |
| 11 | api-kebakaran\_ladang.avi | 100.00 | 0.00 | 0.00 |
| 12 | api-kebakaran\_mobil.avi | 24.02 | 0.00 | 75.98 |
| 13 | api-kebakaran\_mobil2.avi | 2.45 | 0.00 | 97.55 |
| 14 | api-kebakaran\_mobil3.avi | 4.90 | 0.00 | 95.10 |
| 15 | api-kebakaran\_mobil4.avi | 5.88 | 0.00 | 94.12 |
| 16 | api-kebakaran\_mobil5.avi | 91.18 | 0.00 | 8.82 |
| 17 | api-kebakaran\_mobil6.avi | 97.55 | 0.00 | 2.45 |
| 18 | api-kebakaran\_rumah.avi | 2.45 | 0.00 | 97.55 |
| 19 | api-kebakaran\_rumah2.avi | 0.00 | 0.00 | 100.00 |
| 20 | api-kebakaran\_rumah3.avi | 0.00 | 0.00 | 100.00 |
| 21 | api-kebakaran\_rumah4.avi | 54.41 | 0.00 | 45.59 |
| 22 | api-kebakaran\_tol.avi | 0.00 | 0.00 | 100.00 |
| 23 | api-kebakaran-truck.avi | 79.90 | 0.00 | 20.10 |
| 24 | api-kebakaran-truck2.avi | 90.20 | 0.00 | 9.80 |
| 25 | api-kebakaran-truck3.avi | 81.37 | 0.00 | 18.63 |
| 26 | api-kertas.avi | 58.64 | 0.00 | 41.36 |
| 27 | api-kertas2.avi | 55.56 | 0.00 | 44.44 |
| 28 | api-miniatur\_mainan.avi | 2.96 | 0.00 | 97.04 |
| 29 | api-mobil\_mainan.avi | 38.24 | 0.00 | 61.76 |
| 30 | api-orang\_terjun.avi | 2.45 | 0.00 | 97.55 |
| 31 | api-pesawat\_mainan.avi | 21.53 | 0.00 | 78.47 |
| 32 | api-pesawat\_mainan2.avi | 40.24 | 0.00 | 59.76 |
| 33 | api-ruang\_tamu.avi | 2.37 | 0.00 | 97.63 |
| 34 | api-rumah\_mainan.avi | 69.61 | 0.00 | 30.39 |
| 35 | non\_api-anak\_kecil.avi | 100.00 | 0.00 | 0.00 |
| 36 | non\_api-anak\_kecil2.avi | 100.00 | 0.00 | 0.00 |
| 37 | non\_api-anak\_kecil3.avi | 100.00 | 0.00 | 0.00 |
| 38 | non\_api-bertemu.avi | 100.00 | 0.00 | 0.00 |
| 39 | non\_api-jaket\_merah.avi | 100.00 | 0.00 | 0.00 |
| 40 | non\_api-jalan\_malam.avi | 100.00 | 0.00 | 0.00 |
| 41 | non\_api-jalan\_malam2.avi | 100.00 | 0.00 | 0.00 |
| 42 | non\_api-jalan\_raya.avi | 100.00 | 0.00 | 0.00 |
| 43 | non\_api-jalan\_raya2.avi | 100.00 | 0.00 | 0.00 |
| 44 | non\_api-jalan\_raya3.avi | 100.00 | 0.00 | 0.00 |
| 45 | non\_api-kantor.avi | 100.00 | 0.00 | 0.00 |
| 46 | non\_api-kecelakaan.avi | 100.00 | 0.00 | 0.00 |
| 47 | non\_api-kecelakaan2.avi | 100.00 | 0.00 | 0.00 |
| 48 | non\_api-kecelakaan3.avi | 100.00 | 0.00 | 0.00 |
| 49 | non\_api-kerusuhan.avi | 100.00 | 0.00 | 0.00 |
| 50 | non\_api-kerusuhan2.avi | 100.00 | 0.00 | 0.00 |
| 51 | non\_api-kerusuhan3.avi | 100.00 | 0.00 | 0.00 |
| 52 | non\_api-las\_vegas.avi | 98.15 | 1.85 | 0.00 |
| 53 | non\_api-manuju\_mobil.avi | 100.00 | 0.00 | 0.00 |
| 54 | non\_api-manuju\_mobil2.avi | 100.00 | 0.00 | 0.00 |
| 55 | non\_api-manuju\_mobil3.avi | 100.00 | 0.00 | 0.00 |
| 56 | non\_api-parkiran.avi | 100.00 | 0.00 | 0.00 |
| 57 | non\_api-parkiran2.avi | 100.00 | 0.00 | 0.00 |
| 58 | non\_api-parkiran3.avi | 100.00 | 0.00 | 0.00 |
| 59 | non\_api-pencuri.avi | 100.00 | 0.00 | 0.00 |
| 60 | non\_api-pencuri2.avi | 100.00 | 0.00 | 0.00 |
| 61 | non\_api-penembakan.avi | 100.00 | 0.00 | 0.00 |
| 62 | non\_api-serbet.avi | 98.34 | 1.66 | 0.00 |
| 63 | non\_api-tas.avi | 100.00 | 0.00 | 0.00 |
| 64 | non\_api-tas2.avi | 100.00 | 0.00 | 0.00 |
| 65 | non\_api-tas3.avi | 100.00 | 0.00 | 0.00 |
| 66 | non\_api-televisi.avi | 100.00 | 0.00 | 0.00 |
| 67 | non\_api-televisi2.avi | 100.00 | 0.00 | 0.00 |

Tabel A.10 Hasil Uji Coba Menggunakan Parameter konstanta *region* = 5%, *Threshold*  = 5x10-9, = 5 dan kernel RBF

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Nama File | True Positif | False Positif | Missing Rate |
| 1 | api-bakar\_sampah.avi | 0.00 | 0.00 | 100.00 |
| 2 | api-boneka\_dora.avi | 0.00 | 0.00 | 100.00 |
| 3 | api-kayu.avi | 0.00 | 0.00 | 100.00 |
| 4 | api-kayu2.avi | 0.00 | 0.00 | 100.00 |
| 5 | api-kebakaran\_barang.avi | 0.00 | 0.00 | 100.00 |
| 6 | api-kebakaran\_barang2.avi | 0.00 | 0.00 | 100.00 |
| 7 | api-kebakaran\_hutan.avi | 100.00 | 0.00 | 0.00 |
| 8 | api-kebakaran\_hutan2.avi | 100.00 | 0.00 | 0.00 |
| 9 | api-kebakaran\_hutan3.avi | 100.00 | 0.00 | 0.00 |
| 10 | api-kebakaran\_hutan4.avi | 0.00 | 0.00 | 100.00 |
| 11 | api-kebakaran\_ladang.avi | 100.00 | 0.00 | 0.00 |
| 12 | api-kebakaran\_mobil.avi | 91.67 | 0.00 | 8.33 |
| 13 | api-kebakaran\_mobil2.avi | 32.35 | 0.00 | 67.65 |
| 14 | api-kebakaran\_mobil3.avi | 60.29 | 0.00 | 39.71 |
| 15 | api-kebakaran\_mobil4.avi | 0.00 | 0.00 | 100.00 |
| 16 | api-kebakaran\_mobil5.avi | 14.71 | 0.00 | 85.29 |
| 17 | api-kebakaran\_mobil6.avi | 96.57 | 0.00 | 3.43 |
| 18 | api-kebakaran\_rumah.avi | 0.00 | 0.00 | 100.00 |
| 19 | api-kebakaran\_rumah2.avi | 0.00 | 0.00 | 100.00 |
| 20 | api-kebakaran\_rumah3.avi | 15.20 | 0.00 | 84.80 |
| 21 | api-kebakaran\_rumah4.avi | 2.45 | 0.00 | 97.55 |
| 22 | api-kebakaran\_tol.avi | 21.08 | 0.00 | 78.92 |
| 23 | api-kebakaran-truck.avi | 0.00 | 0.00 | 100.00 |
| 24 | api-kebakaran-truck2.avi | 0.49 | 0.00 | 99.51 |
| 25 | api-kebakaran-truck3.avi | 0.00 | 0.00 | 100.00 |
| 26 | api-kertas.avi | 0.00 | 0.00 | 100.00 |
| 27 | api-kertas2.avi | 0.00 | 0.00 | 100.00 |
| 28 | api-miniatur\_mainan.avi | 0.59 | 0.00 | 99.41 |
| 29 | api-mobil\_mainan.avi | 0.00 | 0.00 | 100.00 |
| 30 | api-orang\_terjun.avi | 0.00 | 0.00 | 100.00 |
| 31 | api-pesawat\_mainan.avi | 95.83 | 0.00 | 4.17 |
| 32 | api-pesawat\_mainan2.avi | 15.98 | 0.00 | 84.02 |
| 33 | api-ruang\_tamu.avi | 0.00 | 0.00 | 100.00 |
| 34 | api-rumah\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 35 | non\_api-anak\_kecil.avi | 100.00 | 0.00 | 0.00 |
| 36 | non\_api-anak\_kecil2.avi | 100.00 | 0.00 | 0.00 |
| 37 | non\_api-anak\_kecil3.avi | 100.00 | 0.00 | 0.00 |
| 38 | non\_api-bertemu.avi | 100.00 | 0.00 | 0.00 |
| 39 | non\_api-jaket\_merah.avi | 100.00 | 0.00 | 0.00 |
| 40 | non\_api-jalan\_malam.avi | 100.00 | 0.00 | 0.00 |
| 41 | non\_api-jalan\_malam2.avi | 100.00 | 0.00 | 0.00 |
| 42 | non\_api-jalan\_raya.avi | 100.00 | 0.00 | 0.00 |
| 43 | non\_api-jalan\_raya2.avi | 100.00 | 0.00 | 0.00 |
| 44 | non\_api-jalan\_raya3.avi | 100.00 | 0.00 | 0.00 |
| 45 | non\_api-kantor.avi | 100.00 | 0.00 | 0.00 |
| 46 | non\_api-kecelakaan.avi | 100.00 | 0.00 | 0.00 |
| 47 | non\_api-kecelakaan2.avi | 100.00 | 0.00 | 0.00 |
| 48 | non\_api-kecelakaan3.avi | 100.00 | 0.00 | 0.00 |
| 49 | non\_api-kerusuhan.avi | 100.00 | 0.00 | 0.00 |
| 50 | non\_api-kerusuhan2.avi | 100.00 | 0.00 | 0.00 |
| 51 | non\_api-kerusuhan3.avi | 100.00 | 0.00 | 0.00 |
| 52 | non\_api-las\_vegas.avi | 100.00 | 0.00 | 0.00 |
| 53 | non\_api-manuju\_mobil.avi | 100.00 | 0.00 | 0.00 |
| 54 | non\_api-manuju\_mobil2.avi | 100.00 | 0.00 | 0.00 |
| 55 | non\_api-manuju\_mobil3.avi | 100.00 | 0.00 | 0.00 |
| 56 | non\_api-parkiran.avi | 100.00 | 0.00 | 0.00 |
| 57 | non\_api-parkiran2.avi | 100.00 | 0.00 | 0.00 |
| 58 | non\_api-parkiran3.avi | 100.00 | 0.00 | 0.00 |
| 59 | non\_api-pencuri.avi | 100.00 | 0.00 | 0.00 |
| 60 | non\_api-pencuri2.avi | 100.00 | 0.00 | 0.00 |
| 61 | non\_api-penembakan.avi | 100.00 | 0.00 | 0.00 |
| 62 | non\_api-serbet.avi | 68.51 | 31.49 | 0.00 |
| 63 | non\_api-tas.avi | 100.00 | 0.00 | 0.00 |
| 64 | non\_api-tas2.avi | 100.00 | 0.00 | 0.00 |
| 65 | non\_api-tas3.avi | 100.00 | 0.00 | 0.00 |
| 66 | non\_api-televisi.avi | 100.00 | 0.00 | 0.00 |
| 67 | non\_api-televisi2.avi | 100.00 | 0.00 | 0.00 |

Tabel A.11 Hasil Uji Coba Menggunakan Parameter konstanta *region* = 10%, *Threshold*  = 5x10-9, = 5 dan kernel RBF

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Nama File | True Positif | False Positif | Missing Rate |
| 1 | api-bakar\_sampah.avi | 0.00 | 0.00 | 100.00 |
| 2 | api-boneka\_dora.avi | 0.00 | 0.00 | 100.00 |
| 3 | api-kayu.avi | 0.00 | 0.00 | 100.00 |
| 4 | api-kayu2.avi | 0.00 | 0.00 | 100.00 |
| 5 | api-kebakaran\_barang.avi | 0.00 | 0.00 | 100.00 |
| 6 | api-kebakaran\_barang2.avi | 0.00 | 0.00 | 100.00 |
| 7 | api-kebakaran\_hutan.avi | 27.27 | 0.00 | 72.73 |
| 8 | api-kebakaran\_hutan2.avi | 81.82 | 0.00 | 18.18 |
| 9 | api-kebakaran\_hutan3.avi | 10.10 | 0.00 | 89.90 |
| 10 | api-kebakaran\_hutan4.avi | 0.00 | 0.00 | 100.00 |
| 11 | api-kebakaran\_ladang.avi | 98.22 | 0.00 | 1.78 |
| 12 | api-kebakaran\_mobil.avi | 0.00 | 0.00 | 100.00 |
| 13 | api-kebakaran\_mobil2.avi | 0.00 | 0.00 | 100.00 |
| 14 | api-kebakaran\_mobil3.avi | 0.49 | 0.00 | 99.51 |
| 15 | api-kebakaran\_mobil4.avi | 0.00 | 0.00 | 100.00 |
| 16 | api-kebakaran\_mobil5.avi | 0.00 | 0.00 | 100.00 |
| 17 | api-kebakaran\_mobil6.avi | 0.00 | 0.00 | 100.00 |
| 18 | api-kebakaran\_rumah.avi | 0.00 | 0.00 | 100.00 |
| 19 | api-kebakaran\_rumah2.avi | 0.00 | 0.00 | 100.00 |
| 20 | api-kebakaran\_rumah3.avi | 0.00 | 0.00 | 100.00 |
| 21 | api-kebakaran\_rumah4.avi | 0.00 | 0.00 | 100.00 |
| 22 | api-kebakaran\_tol.avi | 9.80 | 0.00 | 90.20 |
| 23 | api-kebakaran-truck.avi | 0.00 | 0.00 | 100.00 |
| 24 | api-kebakaran-truck2.avi | 0.00 | 0.00 | 100.00 |
| 25 | api-kebakaran-truck3.avi | 0.00 | 0.00 | 100.00 |
| 26 | api-kertas.avi | 0.00 | 0.00 | 100.00 |
| 27 | api-kertas2.avi | 0.00 | 0.00 | 100.00 |
| 28 | api-miniatur\_mainan.avi | 0.00 | 0.00 | 100.00 |
| 29 | api-mobil\_mainan.avi | 0.00 | 0.00 | 100.00 |
| 30 | api-orang\_terjun.avi | 0.00 | 0.00 | 100.00 |
| 31 | api-pesawat\_mainan.avi | 21.53 | 0.00 | 78.47 |
| 32 | api-pesawat\_mainan2.avi | 0.00 | 0.00 | 100.00 |
| 33 | api-ruang\_tamu.avi | 0.00 | 0.00 | 100.00 |
| 34 | api-rumah\_mainan.avi | 95.59 | 0.00 | 4.41 |
| 35 | non\_api-anak\_kecil.avi | 100.00 | 0.00 | 0.00 |
| 36 | non\_api-anak\_kecil2.avi | 100.00 | 0.00 | 0.00 |
| 37 | non\_api-anak\_kecil3.avi | 100.00 | 0.00 | 0.00 |
| 38 | non\_api-bertemu.avi | 100.00 | 0.00 | 0.00 |
| 39 | non\_api-jaket\_merah.avi | 100.00 | 0.00 | 0.00 |
| 40 | non\_api-jalan\_malam.avi | 100.00 | 0.00 | 0.00 |
| 41 | non\_api-jalan\_malam2.avi | 100.00 | 0.00 | 0.00 |
| 42 | non\_api-jalan\_raya.avi | 100.00 | 0.00 | 0.00 |
| 43 | non\_api-jalan\_raya2.avi | 100.00 | 0.00 | 0.00 |
| 44 | non\_api-jalan\_raya3.avi | 100.00 | 0.00 | 0.00 |
| 45 | non\_api-kantor.avi | 100.00 | 0.00 | 0.00 |
| 46 | non\_api-kecelakaan.avi | 100.00 | 0.00 | 0.00 |
| 47 | non\_api-kecelakaan2.avi | 100.00 | 0.00 | 0.00 |
| 48 | non\_api-kecelakaan3.avi | 100.00 | 0.00 | 0.00 |
| 49 | non\_api-kerusuhan.avi | 100.00 | 0.00 | 0.00 |
| 50 | non\_api-kerusuhan2.avi | 100.00 | 0.00 | 0.00 |
| 51 | non\_api-kerusuhan3.avi | 100.00 | 0.00 | 0.00 |
| 52 | non\_api-las\_vegas.avi | 100.00 | 0.00 | 0.00 |
| 53 | non\_api-manuju\_mobil.avi | 100.00 | 0.00 | 0.00 |
| 54 | non\_api-manuju\_mobil2.avi | 100.00 | 0.00 | 0.00 |
| 55 | non\_api-manuju\_mobil3.avi | 100.00 | 0.00 | 0.00 |
| 56 | non\_api-parkiran.avi | 100.00 | 0.00 | 0.00 |
| 57 | non\_api-parkiran2.avi | 100.00 | 0.00 | 0.00 |
| 58 | non\_api-parkiran3.avi | 100.00 | 0.00 | 0.00 |
| 59 | non\_api-pencuri.avi | 100.00 | 0.00 | 0.00 |
| 60 | non\_api-pencuri2.avi | 100.00 | 0.00 | 0.00 |
| 61 | non\_api-penembakan.avi | 100.00 | 0.00 | 0.00 |
| 62 | non\_api-serbet.avi | 72.38 | 27.62 | 0.00 |
| 63 | non\_api-tas.avi | 100.00 | 0.00 | 0.00 |
| 64 | non\_api-tas2.avi | 100.00 | 0.00 | 0.00 |
| 65 | non\_api-tas3.avi | 100.00 | 0.00 | 0.00 |
| 66 | non\_api-televisi.avi | 100.00 | 0.00 | 0.00 |
| 67 | non\_api-televisi2.avi | 100.00 | 0.00 | 0.00 |

Tabel A.12 Hasil Uji Coba Menggunakan Parameter *size frame* = 240 x 320, *Threshold*  = 5x10-9, = 5 dan kernel RBF

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Nama File | True Positif | False Positif | Missing Rate |
| 1 | api-bakar\_sampah.avi | 71.88 | 0.00 | 28.13 |
| 2 | api-boneka\_dora.avi | 48.04 | 0.00 | 51.96 |
| 3 | api-kayu.avi | 100.00 | 0.00 | 0.00 |
| 4 | api-kayu2.avi | 100.00 | 0.00 | 0.00 |
| 5 | api-kebakaran\_barang.avi | 100.00 | 0.00 | 0.00 |
| 6 | api-kebakaran\_barang2.avi | 100.00 | 0.00 | 0.00 |
| 7 | api-kebakaran\_hutan.avi | 100.00 | 0.00 | 0.00 |
| 8 | api-kebakaran\_hutan2.avi | 100.00 | 0.00 | 0.00 |
| 9 | api-kebakaran\_hutan3.avi | 100.00 | 0.00 | 0.00 |
| 10 | api-kebakaran\_hutan4.avi | 100.00 | 0.00 | 0.00 |
| 11 | api-kebakaran\_ladang.avi | 100.00 | 0.00 | 0.00 |
| 12 | api-kebakaran\_mobil.avi | 100.00 | 0.00 | 0.00 |
| 13 | api-kebakaran\_mobil2.avi | 100.00 | 0.00 | 0.00 |
| 14 | api-kebakaran\_mobil3.avi | 100.00 | 0.00 | 0.00 |
| 15 | api-kebakaran\_mobil4.avi | 100.00 | 0.00 | 0.00 |
| 16 | api-kebakaran\_mobil5.avi | 100.00 | 0.00 | 0.00 |
| 17 | api-kebakaran\_mobil6.avi | 100.00 | 0.00 | 0.00 |
| 18 | api-kebakaran\_rumah.avi | 95.10 | 0.00 | 4.90 |
| 19 | api-kebakaran\_rumah2.avi | 100.00 | 0.00 | 0.00 |
| 20 | api-kebakaran\_rumah3.avi | 100.00 | 0.00 | 0.00 |
| 21 | api-kebakaran\_rumah4.avi | 76.47 | 0.00 | 23.53 |
| 22 | api-kebakaran\_tol.avi | 78.92 | 0.00 | 21.08 |
| 23 | api-kebakaran-truck.avi | 97.55 | 0.00 | 2.45 |
| 24 | api-kebakaran-truck2.avi | 100.00 | 0.00 | 0.00 |
| 25 | api-kebakaran-truck3.avi | 74.51 | 0.00 | 25.49 |
| 26 | api-kertas.avi | 24.69 | 0.00 | 75.31 |
| 27 | api-kertas2.avi | 48.15 | 0.00 | 51.85 |
| 28 | api-miniatur\_mainan.avi | 61.54 | 0.00 | 38.46 |
| 29 | api-mobil\_mainan.avi | 96.08 | 0.00 | 3.92 |
| 30 | api-orang\_terjun.avi | 61.27 | 0.00 | 38.73 |
| 31 | api-pesawat\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 32 | api-pesawat\_mainan2.avi | 100.00 | 0.00 | 0.00 |
| 33 | api-ruang\_tamu.avi | 46.75 | 0.00 | 53.25 |
| 34 | api-rumah\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 35 | non\_api-anak\_kecil.avi | 100.00 | 0.00 | 0.00 |
| 36 | non\_api-anak\_kecil2.avi | 100.00 | 0.00 | 0.00 |
| 37 | non\_api-anak\_kecil3.avi | 100.00 | 0.00 | 0.00 |
| 38 | non\_api-bertemu.avi | 100.00 | 0.00 | 0.00 |
| 39 | non\_api-jaket\_merah.avi | 100.00 | 0.00 | 0.00 |
| 40 | non\_api-jalan\_malam.avi | 100.00 | 0.00 | 0.00 |
| 41 | non\_api-jalan\_malam2.avi | 100.00 | 0.00 | 0.00 |
| 42 | non\_api-jalan\_raya.avi | 100.00 | 0.00 | 0.00 |
| 43 | non\_api-jalan\_raya2.avi | 100.00 | 0.00 | 0.00 |
| 44 | non\_api-jalan\_raya3.avi | 100.00 | 0.00 | 0.00 |
| 45 | non\_api-kantor.avi | 100.00 | 0.00 | 0.00 |
| 46 | non\_api-kecelakaan.avi | 100.00 | 0.00 | 0.00 |
| 47 | non\_api-kecelakaan2.avi | 100.00 | 0.00 | 0.00 |
| 48 | non\_api-kecelakaan3.avi | 100.00 | 0.00 | 0.00 |
| 49 | non\_api-kerusuhan.avi | 100.00 | 0.00 | 0.00 |
| 50 | non\_api-kerusuhan2.avi | 100.00 | 0.00 | 0.00 |
| 51 | non\_api-kerusuhan3.avi | 100.00 | 0.00 | 0.00 |
| 52 | non\_api-las\_vegas.avi | 81.48 | 18.52 | 0.00 |
| 53 | non\_api-manuju\_mobil.avi | 100.00 | 0.00 | 0.00 |
| 54 | non\_api-manuju\_mobil2.avi | 100.00 | 0.00 | 0.00 |
| 55 | non\_api-manuju\_mobil3.avi | 100.00 | 0.00 | 0.00 |
| 56 | non\_api-parkiran.avi | 100.00 | 0.00 | 0.00 |
| 57 | non\_api-parkiran2.avi | 100.00 | 0.00 | 0.00 |
| 58 | non\_api-parkiran3.avi | 100.00 | 0.00 | 0.00 |
| 59 | non\_api-pencuri.avi | 100.00 | 0.00 | 0.00 |
| 60 | non\_api-pencuri2.avi | 98.82 | 1.18 | 0.00 |
| 61 | non\_api-penembakan.avi | 100.00 | 0.00 | 0.00 |
| 62 | non\_api-serbet.avi | 66.85 | 33.15 | 0.00 |
| 63 | non\_api-tas.avi | 100.00 | 0.00 | 0.00 |
| 64 | non\_api-tas2.avi | 100.00 | 0.00 | 0.00 |
| 65 | non\_api-tas3.avi | 100.00 | 0.00 | 0.00 |
| 66 | non\_api-televisi.avi | 100.00 | 0.00 | 0.00 |
| 67 | non\_api-televisi2.avi | 100.00 | 0.00 | 0.00 |

Tabel A.13 Hasil Uji Coba Menggunakan Parameter *size frame* = 60 x 80, *Threshold*  = 5x10-9, = 5 dan kernel RBF

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Nama File | True Positif | False Positif | Missing Rate |
| 1 | api-bakar\_sampah.avi | 100.00 | 0.00 | 0.00 |
| 2 | api-boneka\_dora.avi | 91.67 | 0.00 | 8.33 |
| 3 | api-kayu.avi | 100.00 | 0.00 | 0.00 |
| 4 | api-kayu2.avi | 100.00 | 0.00 | 0.00 |
| 5 | api-kebakaran\_barang.avi | 100.00 | 0.00 | 0.00 |
| 6 | api-kebakaran\_barang2.avi | 100.00 | 0.00 | 0.00 |
| 7 | api-kebakaran\_hutan.avi | 100.00 | 0.00 | 0.00 |
| 8 | api-kebakaran\_hutan2.avi | 100.00 | 0.00 | 0.00 |
| 9 | api-kebakaran\_hutan3.avi | 100.00 | 0.00 | 0.00 |
| 10 | api-kebakaran\_hutan4.avi | 100.00 | 0.00 | 0.00 |
| 11 | api-kebakaran\_ladang.avi | 100.00 | 0.00 | 0.00 |
| 12 | api-kebakaran\_mobil.avi | 100.00 | 0.00 | 0.00 |
| 13 | api-kebakaran\_mobil2.avi | 100.00 | 0.00 | 0.00 |
| 14 | api-kebakaran\_mobil3.avi | 98.53 | 0.00 | 1.47 |
| 15 | api-kebakaran\_mobil4.avi | 100.00 | 0.00 | 0.00 |
| 16 | api-kebakaran\_mobil5.avi | 100.00 | 0.00 | 0.00 |
| 17 | api-kebakaran\_mobil6.avi | 100.00 | 0.00 | 0.00 |
| 18 | api-kebakaran\_rumah.avi | 99.51 | 0.00 | 0.49 |
| 19 | api-kebakaran\_rumah2.avi | 100.00 | 0.00 | 0.00 |
| 20 | api-kebakaran\_rumah3.avi | 100.00 | 0.00 | 0.00 |
| 21 | api-kebakaran\_rumah4.avi | 99.51 | 0.00 | 0.49 |
| 22 | api-kebakaran\_tol.avi | 100.00 | 0.00 | 0.00 |
| 23 | api-kebakaran-truck.avi | 100.00 | 0.00 | 0.00 |
| 24 | api-kebakaran-truck2.avi | 100.00 | 0.00 | 0.00 |
| 25 | api-kebakaran-truck3.avi | 100.00 | 0.00 | 0.00 |
| 26 | api-kertas.avi | 98.77 | 0.00 | 1.23 |
| 27 | api-kertas2.avi | 100.00 | 0.00 | 0.00 |
| 28 | api-miniatur\_mainan.avi | 98.82 | 0.00 | 1.18 |
| 29 | api-mobil\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 30 | api-orang\_terjun.avi | 91.18 | 0.00 | 8.82 |
| 31 | api-pesawat\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 32 | api-pesawat\_mainan2.avi | 100.00 | 0.00 | 0.00 |
| 33 | api-ruang\_tamu.avi | 97.04 | 0.00 | 2.96 |
| 34 | api-rumah\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 35 | non\_api-anak\_kecil.avi | 100.00 | 0.00 | 0.00 |
| 36 | non\_api-anak\_kecil2.avi | 100.00 | 0.00 | 0.00 |
| 37 | non\_api-anak\_kecil3.avi | 100.00 | 0.00 | 0.00 |
| 38 | non\_api-bertemu.avi | 100.00 | 0.00 | 0.00 |
| 39 | non\_api-jaket\_merah.avi | 100.00 | 0.00 | 0.00 |
| 40 | non\_api-jalan\_malam.avi | 100.00 | 0.00 | 0.00 |
| 41 | non\_api-jalan\_malam2.avi | 100.00 | 0.00 | 0.00 |
| 42 | non\_api-jalan\_raya.avi | 100.00 | 0.00 | 0.00 |
| 43 | non\_api-jalan\_raya2.avi | 100.00 | 0.00 | 0.00 |
| 44 | non\_api-jalan\_raya3.avi | 100.00 | 0.00 | 0.00 |
| 45 | non\_api-kantor.avi | 100.00 | 0.00 | 0.00 |
| 46 | non\_api-kecelakaan.avi | 100.00 | 0.00 | 0.00 |
| 47 | non\_api-kecelakaan2.avi | 100.00 | 0.00 | 0.00 |
| 48 | non\_api-kecelakaan3.avi | 100.00 | 0.00 | 0.00 |
| 49 | non\_api-kerusuhan.avi | 100.00 | 0.00 | 0.00 |
| 50 | non\_api-kerusuhan2.avi | 100.00 | 0.00 | 0.00 |
| 51 | non\_api-kerusuhan3.avi | 100.00 | 0.00 | 0.00 |
| 52 | non\_api-las\_vegas.avi | 0.00 | 100.00 | 0.00 |
| 53 | non\_api-manuju\_mobil.avi | 100.00 | 0.00 | 0.00 |
| 54 | non\_api-manuju\_mobil2.avi | 100.00 | 0.00 | 0.00 |
| 55 | non\_api-manuju\_mobil3.avi | 100.00 | 0.00 | 0.00 |
| 56 | non\_api-parkiran.avi | 100.00 | 0.00 | 0.00 |
| 57 | non\_api-parkiran2.avi | 100.00 | 0.00 | 0.00 |
| 58 | non\_api-parkiran3.avi | 100.00 | 0.00 | 0.00 |
| 59 | non\_api-pencuri.avi | 31.36 | 68.64 | 0.00 |
| 60 | non\_api-pencuri2.avi | 98.22 | 1.78 | 0.00 |
| 61 | non\_api-penembakan.avi | 89.22 | 10.78 | 0.00 |
| 62 | non\_api-serbet.avi | 65.19 | 34.81 | 0.00 |
| 63 | non\_api-tas.avi | 100.00 | 0.00 | 0.00 |
| 64 | non\_api-tas2.avi | 100.00 | 0.00 | 0.00 |
| 65 | non\_api-tas3.avi | 100.00 | 0.00 | 0.00 |
| 66 | non\_api-televisi.avi | 100.00 | 0.00 | 0.00 |
| 67 | non\_api-televisi2.avi | 100.00 | 0.00 | 0.00 |

Tabel A.14 Hasil Uji Coba Tanpa Menggunakan *region growing* dan perhitungan luasan *region*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Nama File | True Positif | False Positif | Missing Rate |
| 1 | api-bakar\_sampah.avi | 100.00 | 0.00 | 0.00 |
| 2 | api-boneka\_dora.avi | 100.00 | 0.00 | 0.00 |
| 3 | api-kayu.avi | 100.00 | 0.00 | 0.00 |
| 4 | api-kayu2.avi | 100.00 | 0.00 | 0.00 |
| 5 | api-kebakaran\_barang.avi | 100.00 | 0.00 | 0.00 |
| 6 | api-kebakaran\_barang2.avi | 100.00 | 0.00 | 0.00 |
| 7 | api-kebakaran\_hutan.avi | 100.00 | 0.00 | 0.00 |
| 8 | api-kebakaran\_hutan2.avi | 100.00 | 0.00 | 0.00 |
| 9 | api-kebakaran\_hutan3.avi | 100.00 | 0.00 | 0.00 |
| 10 | api-kebakaran\_hutan4.avi | 100.00 | 0.00 | 0.00 |
| 11 | api-kebakaran\_ladang.avi | 100.00 | 0.00 | 0.00 |
| 12 | api-kebakaran\_mobil.avi | 100.00 | 0.00 | 0.00 |
| 13 | api-kebakaran\_mobil2.avi | 100.00 | 0.00 | 0.00 |
| 14 | api-kebakaran\_mobil3.avi | 100.00 | 0.00 | 0.00 |
| 15 | api-kebakaran\_mobil4.avi | 100.00 | 0.00 | 0.00 |
| 16 | api-kebakaran\_mobil5.avi | 100.00 | 0.00 | 0.00 |
| 17 | api-kebakaran\_mobil6.avi | 100.00 | 0.00 | 0.00 |
| 18 | api-kebakaran\_rumah.avi | 100.00 | 0.00 | 0.00 |
| 19 | api-kebakaran\_rumah2.avi | 100.00 | 0.00 | 0.00 |
| 20 | api-kebakaran\_rumah3.avi | 100.00 | 0.00 | 0.00 |
| 21 | api-kebakaran\_rumah4.avi | 100.00 | 0.00 | 0.00 |
| 22 | api-kebakaran\_tol.avi | 100.00 | 0.00 | 0.00 |
| 23 | api-kebakaran-truck.avi | 100.00 | 0.00 | 0.00 |
| 24 | api-kebakaran-truck2.avi | 100.00 | 0.00 | 0.00 |
| 25 | api-kebakaran-truck3.avi | 100.00 | 0.00 | 0.00 |
| 26 | api-kertas.avi | 100.00 | 0.00 | 0.00 |
| 27 | api-kertas2.avi | 100.00 | 0.00 | 0.00 |
| 28 | api-miniatur\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 29 | api-mobil\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 30 | api-orang\_terjun.avi | 100.00 | 0.00 | 0.00 |
| 31 | api-pesawat\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 32 | api-pesawat\_mainan2.avi | 100.00 | 0.00 | 0.00 |
| 33 | api-ruang\_tamu.avi | 100.00 | 0.00 | 0.00 |
| 34 | api-rumah\_mainan.avi | 100.00 | 0.00 | 0.00 |
| 35 | non\_api-anak\_kecil.avi | 100.00 | 0.00 | 0.00 |
| 36 | non\_api-anak\_kecil2.avi | 100.00 | 0.00 | 0.00 |
| 37 | non\_api-anak\_kecil3.avi | 100.00 | 0.00 | 0.00 |
| 38 | non\_api-bertemu.avi | 100.00 | 0.00 | 0.00 |
| 39 | non\_api-jaket\_merah.avi | 81.99 | 18.01 | 0.00 |
| 40 | non\_api-jalan\_malam.avi | 84.67 | 15.33 | 0.00 |
| 41 | non\_api-jalan\_malam2.avi | 95.86 | 4.14 | 0.00 |
| 42 | non\_api-jalan\_raya.avi | 47.93 | 52.07 | 0.00 |
| 43 | non\_api-jalan\_raya2.avi | 31.36 | 68.64 | 0.00 |
| 44 | non\_api-jalan\_raya3.avi | 24.26 | 75.74 | 0.00 |
| 45 | non\_api-kantor.avi | 100.00 | 0.00 | 0.00 |
| 46 | non\_api-kecelakaan.avi | 100.00 | 0.00 | 0.00 |
| 47 | non\_api-kecelakaan2.avi | 100.00 | 0.00 | 0.00 |
| 48 | non\_api-kecelakaan3.avi | 100.00 | 0.00 | 0.00 |
| 49 | non\_api-kerusuhan.avi | 28.40 | 71.60 | 0.00 |
| 50 | non\_api-kerusuhan2.avi | 58.58 | 41.42 | 0.00 |
| 51 | non\_api-kerusuhan3.avi | 55.03 | 44.97 | 0.00 |
| 52 | non\_api-las\_vegas.avi | 0.00 | 100.00 | 0.00 |
| 53 | non\_api-manuju\_mobil.avi | 100.00 | 0.00 | 0.00 |
| 54 | non\_api-manuju\_mobil2.avi | 100.00 | 0.00 | 0.00 |
| 55 | non\_api-manuju\_mobil3.avi | 100.00 | 0.00 | 0.00 |
| 56 | non\_api-parkiran.avi | 100.00 | 0.00 | 0.00 |
| 57 | non\_api-parkiran2.avi | 100.00 | 0.00 | 0.00 |
| 58 | non\_api-parkiran3.avi | 100.00 | 0.00 | 0.00 |
| 59 | non\_api-pencuri.avi | 5.92 | 94.08 | 0.00 |
| 60 | non\_api-pencuri2.avi | 74.56 | 25.44 | 0.00 |
| 61 | non\_api-penembakan.avi | 0.00 | 100.00 | 0.00 |
| 62 | non\_api-serbet.avi | 51.38 | 48.62 | 0.00 |
| 63 | non\_api-tas.avi | 97.89 | 2.11 | 0.00 |
| 64 | non\_api-tas2.avi | 100.00 | 0.00 | 0.00 |
| 65 | non\_api-tas3.avi | 100.00 | 0.00 | 0.00 |
| 66 | non\_api-televisi.avi | 100.00 | 0.00 | 0.00 |
| 67 | non\_api-televisi2.avi | 100.00 | 0.00 | 0.00 |

# BIODATA PENULIS



Hamdi Ahmadi Muzakkiy atau biasa dipanggil Hamdi dilahirkan di Jakarta pada tanggal 15 April 1994 dan dibesarkan di Jakarta. Penulis adalah anak pertama dari tiga bersaudara.

Penulis menempuh pendidikan di SD kasih Ananda (1999-2006), SMP N 84 Jakarta (2006-2009), dan SMA N 75 Jakarta (2009-2012). Setelah lulus SMA penulis melanjutkan ke jenjang perkuliahan di Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember Surabaya. Bidang Studi yang diambil oleh penulis pada saat kuliah di Teknik Informatika ITS adalah Komputasi Cerdas dan Visualisasi.

Selama menempuh kuliah penulis aktif sebagai anggota Himpunan Mahasiswa Teknik Computer (HMTC) ITS. Penulis juga aktif dalam kegiatan kepanitiaan Schematics sebagai staff hubungan masyarakat (Humas) Schematics 2013 dan Wakil Ketua National Programming Contest (NPC) 2014. Selain itu penulis juga aktif menjadi administrator Lab pemrograman(LP) Teknik Informatika ITS. Penulis pernah menjadi asisten dosen dan praktikum untuk mata kuliah Pemrograman Terstruktur (2013), Algoritma dan Struktur Data (2013) , Basis Data (2014) dan Dasar Pemrograman (2015)

Penulis dapat dihubungi melalui alamat *email* hamdiahmadi1504@gmail.com.