

PENDETEKSIAN DAN PENGENALAN WAJAH MANUSIA UNTUK
PENINGKATAN KINERJA KAMERA PENGAMAN

SKRIPSI

TAMRIN IMANUEL PANGGABEAN

121402033



PROGRAM STUDI S1 TEKNOLOGI INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA

MEDAN

2018

PENDETEKSIAN DAN PENGENALAN WAJAH MANUSIA UNTUK
PENINGKATAN KINERJA KAMERA PENGAMAN

SKRIPSI

Diajukan untuk melengkapi tugas dan memenuhi syarat memperoleh
ijazah Sarjana Teknologi Informasi

TAMRIN IMANUEL PANGGABEAN

121402033



PROGRAM STUDI S1 TEKNOLOGI INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA

MEDAN

2018

PERSETUJUAN

Judul : PENDETEKSIAN DAN PENGENALAN WAJAH
MANUSIA UNTUK PENINGKATAN KINERJA
KAMERA PENGAMAN

Kategori : SKRIPSI

Nama : TAMRIN IMANUEL PANGGABEAN

Nomor Induk Mahasiswa : 121402033

Program Studi : SARJANA (S1) TEKNOLOGI INFORMASI

Departemen : TEKNOLOGI INFORMASI

Fakultas : ILMU KOMPUTER DAN TEKNOLOGI
INFORMASI

Komisi Pembimbing :

Pembimbing 2



Seniman S.Kom., M.Kom
NIP. 198705252014041001

Pembimbing 1



Dani Gunawan S.T., M.T.
NIP. 198209152012121002

Diketahui/Disetujui oleh

Program Studi S1 Teknologi Informasi

Ketua,



Romi Fadillah Rahmat B.Comp.Sc., M.Sc
NIP. 198603032010121004

PERNYATAAN

**PENDETEKSIAN DAN PENGENALAN WAJAH MANUSIA UNTUK
PENINGKATAN KINERJA KAMERA PENGAMAN**

SKRIPSI

Saya mengakui bahwa skripsi ini adalah hasil karya saya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing telah disebutkan sumbernya.

Medan, 22 Februari 2018

Tamrin Imanuel Panggabean

NIM. 121402033

UCAPAN TERIMA KASIH

Puji dan syukur penulis panjatkan kepada Tuhan Yesus, karena atas berkat dan izinNya penulis dapat menyelesaikan penyusunan skripsi ini sebagai syarat untuk memperoleh gelar Sarjana Komputer pada Program Studi S1 Teknologi Informasi Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara. Selama proses penyelesaian skripsi ini, banyak bantuan dan kerja sama serta doa dan dukungan dari berbagai pihak, oleh karena itu penulis ucapkan terima kasih kepada :

1. Kedua orangtua penulis, Bapak Maraden Panggabean dan Ibu Rosmeriani Lubis yang telah membesarkan penulis dan selalu memberikan dukungan moril dan materil, juga kepada kedua saudara kandung penulis Gilbert Panggabean dan Jesiska Panggabean yang memberikan dukungan doa.
2. Mafera Gustina Siagian A.Md selaku orang yang penulis kasihi yang selalu ada memberikan semangat dan dukungan moril kepada penulis untuk bisa menyelesaikan skripsi ini serta tak lupa mendoakan penulis.
3. Bapak Dani Gunawan S.T., M.T. selaku pembimbing 1 dan Bapak Seniman S.Kom., M.Kom. selaku pembimbing 2 yang telah meluangkan waktunya untuk memberikan kritik dan saran terhadap penulis untuk penyelesaian skripsi ini.
4. Ibu DR. Erna Budhiarti Nababan M.IT dan juga bapak Ainul Hizriadi S.Kom.,M.Sc. yang memberikan saran kepada penulis untuk menyelesaikan skripsi ini.
5. Seluruh dosen, staff dan pegawai Teknologi Informasi yang tidak dapat saya sebutkan satu persatu yang telah membantu lancarnya penyelesaian skripsi ini.
6. Seluruh rekan-rekan seperjuangan angkatan 2012 Program Studi Teknologi Informasi yang telah memberikan bantuan khususnya Syafrizal Lubis, Renato Rashidi Siahaan, Efraim Emdastra Sinulingga, Michael Putra Loi, Tito Pandiangan dan kawan-kawan lainnya yang tidak dapat saya sebutkan satu-persatu.
7. Saudara-saudara terdekat.

Penulis menyadari bahwa hasil penulisan skripsi ini masih banyak kekurangan dan masih jauh dari sempurna. Oleh karena itu kritik dan saran pembaca sangatlah penulis

harapkan. Akhir kata penulis ucapkan terima kasih dan juga mohon maaf apabila ada kesalahan dalam skripsi.

Penulis

Tamrin Imanuel Panggabean

ABSTRAK

Kamera pengaman adalah kamera yang digunakan untuk membantu meningkatkan keamanan yang menghasilkan citra berupa video dan disimpan kedalam media penyimpanan. Citra yang disimpan adalah hasil rekaman yang diambil pada saat ada atau tidaknya kejadian dalam rekaman dan hanya bisa menjadi bukti untuk tindak kejahatan. Untuk itu dibutuhkan sebuah pendekatan yang membantu meningkatkan kinerja dari pada kamera pengaman. Pendekatan yang dapat dilakukan adalah melakukan deteksi terhadap objek manusia yang ada didepan kamera dan memberikan informasi tentang objek manusia tersebut tersebut. Metode yang dipakai untuk pendekatan tersebut adalah *Support Vector Machine* dan *Deep Convolutional Neural Network*. Citra yang diinput dari kamera akan mengalami proses *grayscale*, kemudian akan dilakukan *threshold* untuk memisahkan background dan objek yang selanjutnya akan diambil nilai ekstraksi dan diklasifikasikan dengan *Support Vector Machine* dan *Deep Convolutional Neural Network* yang nantinya akan disimpan disaat ada manusia dan mengirimkan notifikasi saat manusia tidak dikenali. Pada penelitian ini akurasi dari pengenalan wajah 83% dan penghematan media penyimpanan adalah 73,3 %.

Kata kunci : *Deep neural network*, Kamera Pengaman, *Tensorflow*, *Deep Convolutional Neural Network*, *Support Vector Machine*, *Face Recognition*, *Human Detection*.

*Detection and Recognition of Human Face for Improved Performance Cameras
Performance*

ABSTRACT

A security camera is a camera used to help improve the security that produces a video image and is stored into a storage medium. The stored image is the recording taken at the moment of the occurrence of the recording and can only be evidence for a crime. For that needed an approach that helps improve the performance of the security camera. The approach that can be done is to detect the human object that is in front of the camera and provide information about the human object is. The methods used for this approach are Support Vector Machine and Deep Convolutional Neural Network. The image inputted from the camera will experience the process of grayscaling, then threshold will be done to separate the background and object which will then be extracted value and classified with Support Vector Machine and Deep Convolutional Neural Network which will be stored when there is human and send notification when humans are not recognized. In this study the accuracy of face recognition reached 83% and savings of storage media was 73.3%.

Keyword : Deep neural network, Security Camera, Tensorflow, Deep Convolutional Neural Network, Support Vector Machine, Face Recognition, Human Detection.

DAFTAR ISI

	Halaman
PERSETUJUAN	iii
PERNYATAAN	iv
UCAPAN TERIMA KASIH	v
ABSTRAK	vii
ABSTRACT	viii
DAFTAR ISI	ix
DAFTAR TABEL	xii
DAFTAR GAMBAR	xiii
 BAB 1 PENDAHULUAN	 1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Tujuan Penelitian	3
1.4. Batasan Masalah	3
1.5. Manfaat Penelitian	3
1.6. Metodologi Penelitian	3
1.7. Sistematika Penulisan	4
 BAB 2 LANDASAN TEORI	 6
2.1. Citra	6
2.2. Pengolahan Citra	6
2.2.1. Resizing	7
2.2.2. Grayscale	8
2.2.3. Erosi	8
2.3. Ekstraksi Fitur	8

2.4. Jaringan Syaraf Tiruan	9
2.4.1. Komponen Jaringan Syaraf	9
2.4.2. Arsitektur Jaringan	10
2.4.3. Fungsi Aktivasi	12
2.5. Support Vector Machine	13
2.5.1. Struktural Risk Minimization	14
2.6. Convolutional Neural Network (CNN)	15
2.7. Deep Convolutional Neural Network	17
2.8. TensorFlow	18
2.9. Penelitian Terdahulu	19
 BAB 3 METODE PENELITIAN	 21
3.1. Data Penelitian	21
3.2. Arsitektur Umum	21
3.2.1. Human Detection	22
3.2.2. Pengenalan Wajah	26
3.3. Perancangan Aplikasi	32
3.4. Metode Pengujian	34
3.5. Metode Evaluasi	35
 BAB 4 IMPLEMENTASI HASIL PENGUJIAN SISTEM	 37
4.1. Implementasi Sistem	37
4.1.1. Spesifikasi Hardware dan Software	37
4.1.2. Implementasi	37
4.1.3. Implementasi Perancangan Antarmuka	39
4.2. Hasil Pengujian Sistem	40
4.2.1. Pendeteksian Manusia	40
4.2.2. Pengenalan Wajah	51
4.2.3. Effisiensi Sistem Kamera	62

BAB 5 KESIMPULAN DAN SARAN	64
5.1. Kesimpulan	64
5.2. Saran	64
 DAFTAR PUSTAKA	 66

DAFTAR TABEL

	Halaman
Tabel 2.1	Penelitian Terdahulu 19
Tabel 3.1	Proses Pelatihan Setiap Layer 29
Tabel 4.1	Pengujian Pendeteksian Manusia Dengan Cahaya 5 Lux 42
Tabel 4.2	Pengujian Pendeteksian Manusia Dengan Cahaya 200 Lux 44
Tabel 4.3	Pengujian Pendeteksian Manusia Dengan Cahaya 750 Lux 45
Tabel 4.4	Pengujian Pendeteksian Manusia Dengan Cahaya 1000 Lux 46
Tabel 4.5	Hasil Perhitungan F-Score Pendeteksian Manusia Dengan Perbedaan Intensitas Cahaya 47
Tabel 4.6	Pengujian Pendeteksian Manusia Pada Jarak 1 Meter 48
Tabel 4.7	Pengujian Pendeteksian Manusia Pada Jarak 2 Meter 49
Tabel 4.8	Pengujian Pendeteksian Manusia Pada Jarak 5 Meter 50
Tabel 4.9	Hasil Perhitungan F-Score Pendeteksian Manusia Dengan Perbedaan Jarak 51
Tabel 4.10	Pengujian Pengenalan Wajah Dengan Cahaya 5 Lux 52
Tabel 4.11	Pengujian Pengenalan Wajah Dengan Cahaya 200 Lux 53
Tabel 4.12	Pengujian Pengenalan Wajah Dengan Cahaya 750 Lux 53
Tabel 4.13	Pengujian Pengenalan Wajah Dengan Cahaya 1000 Lux 54
Tabel 4.14	Hasil Perhitungan F-Score Pengenalan Wajah Dengan Perbedaan Intensitas Cahaya 55
Tabel 4.15	Pengujian Pengenalan Wajah Manusia Pada Jarak 1 Meter 55
Tabel 4.16	Pengujian Pengenalan Wajah Manusia Pada Jarak 2 Meter 56
Tabel 4.17	Pengujian Pengenalan Wajah Manusia Pada Jarak 5 Meter 57
Tabel 4.18	Hasil Perhitungan F-Score Pengenalan Wajah Dengan Perbedaan Jarak 58
Tabel 4.19	Posisi Wajah Untuk Pengujian 59
Tabel 4.20	Pengujian Pengenalan Wajah Dengan Posisi Wajah Berbeda 60
Tabel 4.21	Hasil Perhitungan Akurasi Dari Berbagai Posisi Wajah 62
Tabel 4.22	Hasil Perhitungan F-Score Pengenalan Wajah Dengan Posisi Wajah Berbeda 62

DAFTAR GAMBAR

		Halaman
Gambar 2.1	Proses Erosi Dengan Bantuan Matriks 3X3	8
Gambar 2.2	Struktur Neuron Jaringan Syaraf	9
Gambar 2.3	Jaringan Syaraf Dengan 3 Lapisan	10
Gambar 2.4	Jaringan Syaraf Dengan Lapisan Tunggal	11
Gambar 2.5	Jaringan Syaraf Dengan Banyak Lapisan	11
Gambar 2.6	Jaringan Syaraf Dengan Lapisan Kompetitif	12
Gambar 2.7	Arsitektur Convolutional Neural Network	15
Gambar 2.8	Contoh Diagram Convolutiona Layer	16
Gambar 2.9	Contoh Diagram Max Pooling Layer	16
Gambar 2.10	Arsitektur Deep Convolutional Neural Network	18
Gambar 2.11	Arsitektur Umum TensorFlow	18
Gambar 3.1	Contoh Citra Grayscale	22
Gambar 3.2	Arsitektur Umum	23
Gambar 3.3	Contoh Citra Threshold	24
Gambar 3.4	Gambar Threshold dan Posisi Kontur	25
Gambar 3.5	Manusia Terdeteksi dan Posisi Wajah Pada Frame	25
Gambar 3.6	Proses Convolutional Layer Pertama	27
Gambar 3.7	Proses Convolutional Layer Kedua	27
Gambar 3.8	Proses Convolutional Layer Ketiga	28
Gambar 3.9	Proses Convolutional Layer Keempat	28
Gambar 3.10	Proses Convolutional Layer Kelima	29
Gambar 3.11	Titik Posisi Pada Wajah	31
Gambar 3.12	Posisi Wajah Manusia	31
Gambar 3.13	Contoh Landmark Point Pada Wajah	31
Gambar 3.14	Rancangan Halaman Awal Sistem	33
Gambar 3.15	Rancangan Halaman Utama	33
Gambar 3.16	Perancangan Isi Notifikasi Berupa SMS	34
Gambar 3.17	Variabel Perhitungan F-Score	35

Gambar 4.1	Halaman Utama Aplikasi	39
Gambar 4.2	Notifikasi SMS	40
Gambar 4.3	Perekaman Saat Terdeteksi Manusia	41
Gambar 4.4	Percobaan Dengan Tumbuhan Tinggi	41
Gambar 4.5	Percobaan Dengan Kucing	41
Gambar 4.6	Kondisi Lingkungan Intensitas Cahaya 5 Lux	42
Gambar 4.7	Kondisi Lingkungan Intensitas Cahaya 200 Lux	43
Gambar 4.8	Kondisi Lingkungan Intensitas Cahaya 750 Lux	45
Gambar 4.9	Kondisi Lingkungan Intensitas Cahaya 1000 Lux	46
Gambar 4.10	Objek Pada Jarak 1 Meter dari Kamera	48
Gambar 4.11	Objek Pada Jarak 2 Meter dari Kamera	49
Gambar 4.12	Objek Pada Jarak 5 Meter dari Kamera	50

BAB 1

PENDAHULUAN

1.1. Latar Belakang

Kamera pengaman adalah sebuah kamera yang digunakan untuk membantu pengamanan sesuatu, baik tempat, barang atau seseorang. Kamera pengaman akan memberikan hasil berupa sebuah citra digital baik dalam bentuk image atau sebuah video yang disimpan kedalam media penyimpanan. Citra yang disimpan kedalam media penyimpanan tersebut adalah semua hasil rekaman yang diambil oleh kamera pengaman tanpa memperdulikan adanya manusia atau tidak, dikenal atau tidaknya manusia yang ada didepan kamera dan adanya kejadian atau tidak. Dengan cara penyimpanan tersebut maka akan dibutuhkan sebuah *source* yang cukup besar untuk dapat menampung semua hasil rekaman kamera pengaman. Oleh karena itu dibutuhkan sebuah pendekatan yang nantinya akan meningkatkan kinerja dari kamera pengaman tersebut dari sisi efisiensi kamera pengaman tersebut .

Ada beberapa penelitian yang telah dilakukan sebelumnya yang memiliki keterkaitan dengan masalah tersebut. Jianpeng Zhou dan Jack Hoang dalam penelitiannya menggunakan *robust human detection* (Jianpeng Zhoun & Jack Hoang, 2005) untuk mendeteksi manusia dan mengikuti pergerakannya. Nishu sigla pada penelitiannya menggunakan metode *static background* dengan menganalisis perbedaan tiap frame (Nishu sigla, 2014) untuk mendeteksi adanya pergerakan objek dalam sebuah background. Sreedevi et al dalam penelitiannya memanfaatkan perbandingan frame (Sreedevi et al, 2012) untuk mendeteksi sebuah gerakan dalam video kemudian mendeteksi apakah gerakan tersebut dilakukan manusia dan kemudian menyimpan video tersebut. Naouar Belghini dalam penelitiannya mengenali wajah menggunakan *3D depth information* dimana ujung hidung untuk setiap objek yang dikenali sebagai

manusia sebagai dasar koordinat (Belghini et al, 2012). Farhan Kahn dalam penelitiannya menggunakan *robust mechanism* untuk mendeteksi gerakan dan menganalisisnya. Mereka meminta objek manusia menggunakan sarung tangan yang berbeda pada kedua tangannya. Sehingga mereka mendeteksi pergerakan manusia dari pergerakan tangan. Edwards et al dalam penelitiannya memanfaatkan wajah seseorang dengan berbagai macam ekspresi untuk menjadi data latih dan melakukan perbandingan terhadap data uji.

Pada penelitian ini, peneliti akan menggunakan teknik *motion detector* dimana kamera akan mulai merekam hanya saat ada gerakan ataupun kegiatan dari objek yang dideteksi sebagai manusia. Akan tetapi jika ada kegiatan ataupun aktifitas dari objek yang tidak dideteksi sebagai manusia, maka kamera tidak akan merekam. Tidak hanya disitu, akan dilakukan sebuah validasi ke pemilik rumah melalui notifikasi pada smartphone pengguna apakah objek yang dideteksi sebagai manusia dikenal atau tidak. Jika tidak maka pemilik rumah bisa menghubungi pihak yang berwajib untuk menindaklanjuti hal ini. Untuk pemrosesan image dan pengenalannya peneliti akan menggunakan metode *Convolutional Neural Network (CNN)* yang memiliki akurasi sampai 93,7%. Sementara untuk alat yang akan digunakan adalah webcam. Hal ini juga ditujukan untuk membantu bagi orang yang ingin memiliki keamanan rumah tapi dengan dana yang tidak tinggi. Karena webcam standar dibandingkan dengan cctv yang mampu mendeteksi gerakan akan sangat jauh perbandingan dananya.

Oleh karena itu penulis ingin mengajukan sebuah penelitian yang berjudul “PENDETEKSIAN DAN PENGENALAN WAJAH MANUSIA UNTUK PENINGKATAN KINERJA KAMERA PENGAMAN”. Tujuan peneliti adalah agar dapat mengefisiensikan kinerja dari pada kamera pengaman dan akan menghemat baik dari biaya ataupun *memory* penyimpanan.

1.2. Rumusan Masalah

Kamera pengaman akan merekam baik adanya sebuah kejadian ataupun tidak ada kejadian. Hal ini sangat kurang efisien karena akan memakan banyak *resources memory*. Selain itu kamera pengaman juga hanya menjadi bukti disaat terjadinya kejahatan. Oleh karena itu diperlukan sebuah pendekatan untuk dapat meningkatkan efisiensi dari pada kamera pengaman tersebut.

1.3. Tujuan Penelitian

Tujuan dari penelitian ini adalah meningkatkan efisiensi kamera pengaman dengan menyimpan rekaman yang hanya ada kejadian serta membantu mencegah terjadinya kejahatan.

1.4. Batasan Masalah

Guna mencegah meluasnya cakupan permasalahan yang akan dibahas dan untuk membuat studi ini lebih terarah maka dilakukan pembatasan masalah sebagai berikut :

1. Kamera akan menangkap semua objek yang bergerak.
2. Pengenalan wajah hanya dapat dilakukan apabila wajah tidak ditutupi seperti menggunakan topeng atau penutup wajah lainnya.
3. Notifikasi akan terkirim hanya apabila komputer pemrosesan terhubung ke jaringan internet.
4. Notifikasi akan diterima apabila pemilik smartphone menggunakan jaringan internet.
5. Pendeteksian hanya dilakukan dalam kondisi cahaya 5 lux, 200 lux, 750 lux, 1000 lux

1.5. Manfaat Penelitian

Manfaat yang diperoleh dari penelitian ini adalah :

1. Mengefisiensikan peralatan teknologi kamera pengaman dan media penyimpanan.
2. Menghemat biaya yang dikeluarkan untuk teknologi kamera pengaman.

1.6. Metodologi Penelitian

Metodologi penelitian yang digunakan pada penelitian ini adalah:

a. Studi Literatur

Pada tahap ini dilakukan studi kepustakaan yaitu proses mengumpulkan bahan referensi mengenai pengolahan citra digital, pendeteksian manusia, dan metode Support Vector Machine dan Deep Convolutional Neural Network dari berbagai buku, jurnal, artikel, dan beberapa referensi lainnya.

b. Analisis

Pada tahap ini dilakukan analisis terhadap studi literatur untuk mengetahui dan mendapatkan pemahaman mengenai pendeteksian manusia dan gerakan untuk menyelesaikan masalah tentang efisiensi kamera pengaman.

c. Perancangan

Pada tahap perancangan sistem dilakukan perancangan arsitektur dan antarmuka agar tampilan aplikasi yang akan dibangun *user friendly*. Proses perancangan dilakukan berdasarkan hasil analisis studi literatur yang telah didapatkan.

d. Implementasi

Pada tahap implementasi dilakukan pengkodean untuk membangun aplikasi yang dimaksud dengan menggunakan bahasa pemrograman yang telah dipilih dan ditentukan sebelumnya oleh penulis.

e. Pengujian

Pada tahap ini dilakukan pengujian terhadap pengkodean program untuk memastikan apakah program sudah berjalan sesuai dengan yang direncanakan.

f. Penyusunan Laporan

Penyusunan Laporan merupakan tahap pembuatan dokumentasi dari aplikasi yang sudah dibangun.

1.7. Sistematika Penulisan

Sistematika penulisan dari skripsi ini terdiri dari lima bagian utama sebagai berikut:

Bab 1 Pendahuluan

Bab ini berisi latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian, dan sistematika penulisan.

Bab 2 Landasan Teori

Bab ini berisi teori-teori yang digunakan untuk memahami permasalahan yang dibahas pada penelitian ini.

Bab 3 Analisis dan Perancangan

Bab ini berisi analisis dan penerapan metode *Support Vector Machine* untuk mendeteksi manusia dan Convolutional Neural Network untuk pengenalan wajah serta perancangan aplikasi seperti pemodelan dengan flowchart, use case dan sequence diagram.

Bab 4 Implementasi dan Pengujian

Bab ini berisi pembahasan tentang implementasi dari analisis dan perancangan yang disusun pada Bab 3 dan pengujian apakah hasil yang didapatkan sesuai dengan yang diharapkan.

Bab 5 Kesimpulan dan Saran

Bab ini berisi kesimpulan dari keseluruhan uraian bab-bab sebelumnya dan saransaran yang diajukan untuk pengembangan penelitian selanjutnya.

BAB 2

LANDASAN TEORI

Bab ini berisi tentang teori-teori yang dipakai dalam penelitian ini.

2.1. Citra

Citra adalah representasi dari suatu objek. Secara harafiah citra adalah gambar pada bidang dua dimensi. Citra memiliki peranan penting sebagai bentuk informasi visual. Citra mempunyai karakteristik yang tidak dimiliki oleh teks, yaitu citra kaya akan informasi.

Citra dapat didefinisikan sebagai fungsi dua dimensi $f(x,y)$, dimana x dan y adalah koordinat bidang datar, dan harga f pada setiap koordinat (x,y) disebut intensitas atau level keabuan (*gray level*) dari citra di titik itu. Jika x , y , dan f semuanya berhingga (*finite*) dan nilainya diskrit maka citra itu disebut citra digital (Gonzalez & Woods, 2008).

Citra digital dapat dibedakan menjadi beberapa jenis yaitu citra biner (*binary image*), citra keabuan (*grayscale image*) dan citra warna (*color image*).

2.2. Pengolahan Citra

Seringkali citra mengalami penurunan mutu (degradasi), misalnya mengandung cacat (*noise*), warnanya terlalu kontras, kurang tajam, kabur, dan sebagainya. Citra seperti ini lebih sulit untuk diinterpretasi karena informasi yang disampaikan oleh citra tersebut berkurang. Agar lebih mudah diinterpretasi, maka citra tersebut perlu dimanipulasi menjadi citra baru yang memiliki kualitas lebih baik. Bidang studi yang menyangkut hal ini adalah pengolahan citra (Gonzalez & Woods, 2008).

Ada beberapa operasi pengolahan citra yang biasanya dilakukan, diantaranya (Jain, 1989) :

1. Representasi dan pemodelan citra

Bertujuan untuk memberikan gambaran suatu hal. Contohnya adalah bagaimana pencahayaan benda di suatu lokasi dari hasil foto kamera, karakteristik jaringan tubuh dari citra X-Ray, posisi target dari sebuah radar dari citra radar, suhu suatu daerah dari citra inframerah atau medan gravitasi suatu daerah dari citra geofisika.

2. Perbaikan citra

Bertujuan untuk menonjolkan fitur-fitur tertentu yang ada pada citra untuk kepentingan ekstraksi fitur, analisis citra atau menampilkan informasi citra tersebut. Contohnya adalah perbaikan kontras, perbaikan tepi objek, pemberian warna semu, penipisan derau, penajaman, dan sebagainya.

3. Pemugaran citra

Bertujuan untuk menghilangkan atau mengurangi degradasi citra. Mirip dengan perbaikan citra, hanya saja pada pemugaran citra sudah diketahui penyebab degradasi citra. Contohnya adalah menghilangkan kesamaran (*deblurring*) yang disebabkan oleh keterbatasan sensor, usia citra yang sudah tua atau banyaknya derau (*noise*) pada citra.

4. Analisis citra (*image analysis*)

Bertujuan untuk menghitung besaran kuantitatif dari citra untuk menghasilkan deskripsi dari citra tersebut. Teknik analisis citra memerlukan fitur-fitur tertentu untuk membantu dalam identifikasi objek. Proses segmentasi terkadang diperlukan untuk memisahkan objek yang diinginkan dari sekelilingnya sehingga selanjutnya dapat dilakukan pengukuran. Contohnya adalah mendeteksi tepi objek, ekstraksi batas (*boundary*), dan representasi daerah (*region*).

5. Rekonstruksi citra

Bertujuan untuk membentuk ulang objek dari beberapa citra hasil proyeksi. Contohnya adalah gambar 3D yang dihasilkan dari beberapa foto rontgen dengan sinar X.

6. Kompresi citra

Bertujuan untuk mengurangi jumlah bit yang dibutuhkan untuk menyimpan atau mengirim gambar tanpa kehilangan informasi yang diperlukan. Citra berukuran besar akan mempengaruhi penggunaan memori dan lamanya waktu pengiriman.

2.2.1. Resizing

Resizing adalah proses mengubah ukuran panjang dan lebar citra pada arah vertikal dan/atau horizontal.

2.2.2. Grayscale

Grayscale adalah proses mengubah citra warna menjadi citra keabuan. Cara mengubah citra warna ke citra keabuan adalah dengan mencari nilai rata-rata dari ketiga nilai warna.

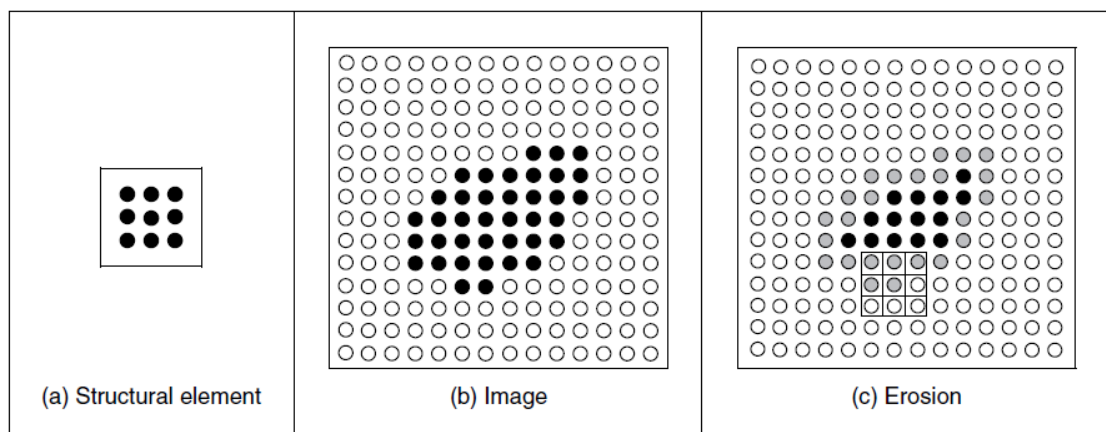
$$G = (R + G + B)/3 \quad (2.1)$$

Keterangan :

R = nilai warna merah, G = nilai warna hijau, B = nilai warna biru

2.2.3. Erosi

Erosi adalah proses penebalan objek citra biner. Erosi dilakukan dengan mengubah nilai dari piksel tetangga menjadi 0 berdasarkan matriks berukuran $m \times m$. Proses erosi dapat dilihat pada Gambar 2.1.



Gambar 2.1. Proses erosi dengan bantuan matriks 3x3 (Nixon & Aguado, 2008)

2.3. Ekstraksi Fitur

Ekstraksi fitur adalah suatu proses pengambilan ciri/fitur dari citra yang nilainya akan digunakan untuk proses selanjutnya. Ekstraksi fitur dapat dibedakan menjadi 3 jenis (Kalel, Pisal, & Bagawade, 2016) :

1. Ekstraksi Warna

Ruang warna mewakili warna dalam bentuk nilai intensitas. Kita bisa menentukan, memvisualisasikan dan menciptakan warna dengan menggunakan metode ruang warna. Metode yang sering digunakan untuk ekstraksi warna antara lain *Histogram Intersection*, *Zernike Chromaticity Distribution Moments* dan *Color Histogram*.

2. Ekstraksi Bentuk

Bentuk adalah sumber informasi utama yang bisa digunakan untuk pengenalan objek. Tanpa bentuk, objek konten visual tidak bisa dikenali dengan baik. Dua objek tidak bisa memiliki bentuk yang sama persis tapi dengan menggunakan berbagai algoritma kita bisa mengenali bentuk yang serupa dengan mudah. Metode yang sering digunakan untuk ekstraksi bentuk antara lain *Binary Image* dan segmentasi secara horizontal dan vertikal.

3. Ekstraksi Tekstur

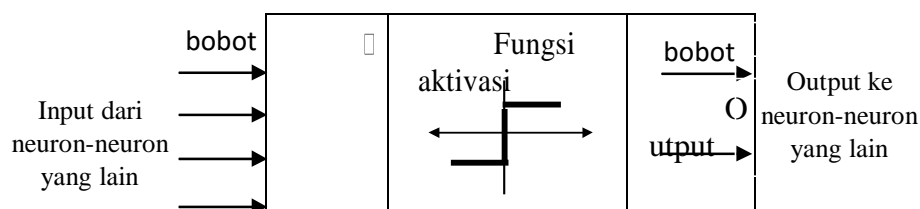
Tekstur mengandung informasi penting tentang susunan dasar permukaan objek. Metode yang sering digunakan untuk ekstraksi tekstur antara lain *Gray Level Co-Occurrence Matrix* (GLCM), *Gray Level Run Length Matrix* (GLRLM), dan *Edge Detection*.

2.4. Jaringan Syaraf Tiruan

Jaringan syaraf tiruan (*artificial neural network*) adalah suatu proses pembelajaran buatan pada komputer yang meniru cara kerja otak manusia. Jaringan syaraf tiruan digunakan untuk memecahkan masalah tertentu seperti pengenalan pola atau klasifikasi.

2.4.1. Komponen Jaringan Syaraf

Ada beberapa tipe jaringan syaraf, tetapi hampir semuanya memiliki komponen-komponen yang sama. Seperti otak manusia, jaringan syaraf juga terdiri dari beberapa neuron, dan ada hubungan antara neuron-neuron tersebut. Neuron-neuron tersebut akan mentransformasikan informasi yang diterima melalui sambungan keluarnya menuju ke neuron-neuron yang lain. Pada jaringan syaraf, hubungan ini dikenal dengan nama bobot. Informasi tersebut disimpan pada suatu nilai tertentu pada bobot tersebut.

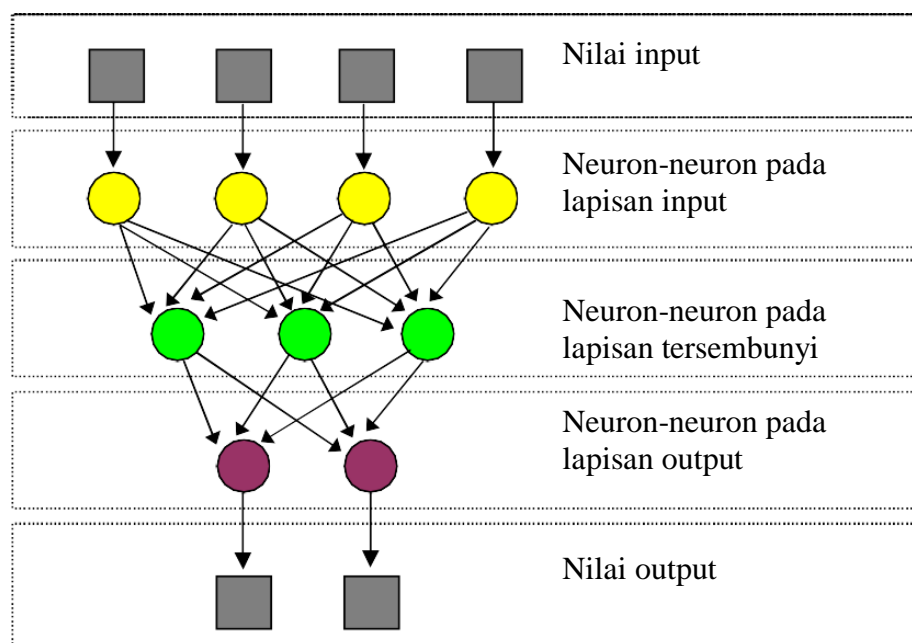


Gambar 2.2. Struktur neuron jaringan syaraf

Input akan dikirim ke neuron dengan bobot kedatangan tertentu. Input akan diproses oleh suatu fungsi perambatan yang akan menjumlahkan nilai-nilai semua bobot yang datang. Hasil penjumlahan ini kemudian dibandingkan dengan suatu nilai ambang

(*threshold*) tertentu melalui fungsi aktivasi (*activation function*) setiap neuron. Apabila input tersebut melewati suatu nilai ambang tertentu, maka neuron tersebut akan diaktifkan. Jika tidak, maka neuron tersebut tidak akan diaktifkan. Apabila neuron tersebut diaktifkan, maka neuron tersebut akan mengirimkan output melalui bobot-bobot outputnya ke semua neuron yang berhubungan dengannya. Demikian seterusnya.

Pada jaringan syaraf, neuron-neuron akan dikumpulkan dalam lapisan-lapisan (*layer*) yang disebut dengan lapisan neuron (*neuron layers*). Biasanya neuron-neuron pada satu lapisan akan dihubungkan dengan lapisan-lapisan sebelum dan sesudahnya (kecuali lapisan input dan lapisan output). Informasi yang diberikan pada jaringan syaraf akan dirambatkan dari lapisan input sampai ke lapisan output melalui lapisan yang lainnya, yang sering dikenal dengan nama lapisan tersembunyi (*hidden layer*).



Gambar 2.3. Jaringan syaraf dengan 3 lapisan

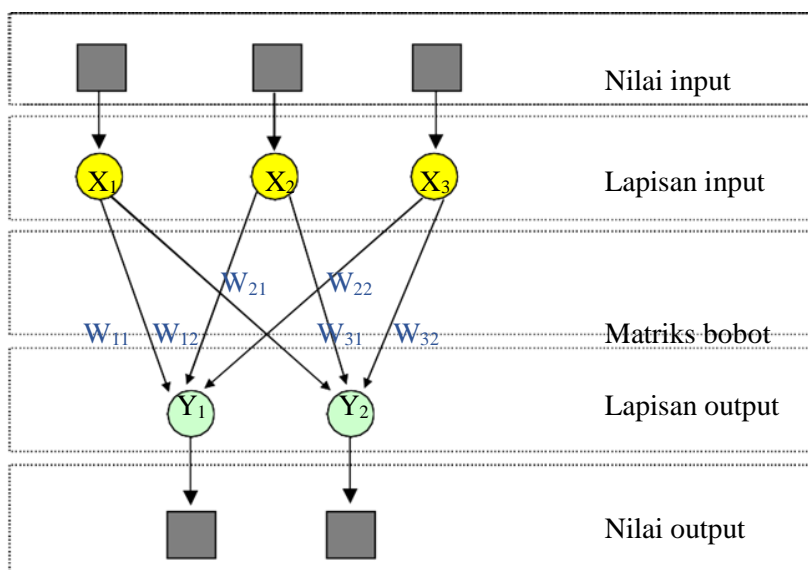
2.4.2. Arsitektur Jaringan

Ada beberapa arsitektur jaringan syaraf yaitu :

1. Jaringan dengan lapisan tunggal (*single layer net*)

Jaringan dengan lapisan tunggal hanya memiliki satu lapisan dengan bobot-bobot terhubung. Jaringan ini hanya menerima input kemudian secara langsung akan mengolahnya menjadi output tanpa harus melalui lapisan tersembunyi. Lapisan input memiliki 3 neuron, yaitu X_1 , X_2 dan X_3 . Sedangkan pada lapisan output memiliki 2 neuron yaitu Y_1 dan Y_2 . Neuron-neuron pada kedua lapisan saling berhubungan.

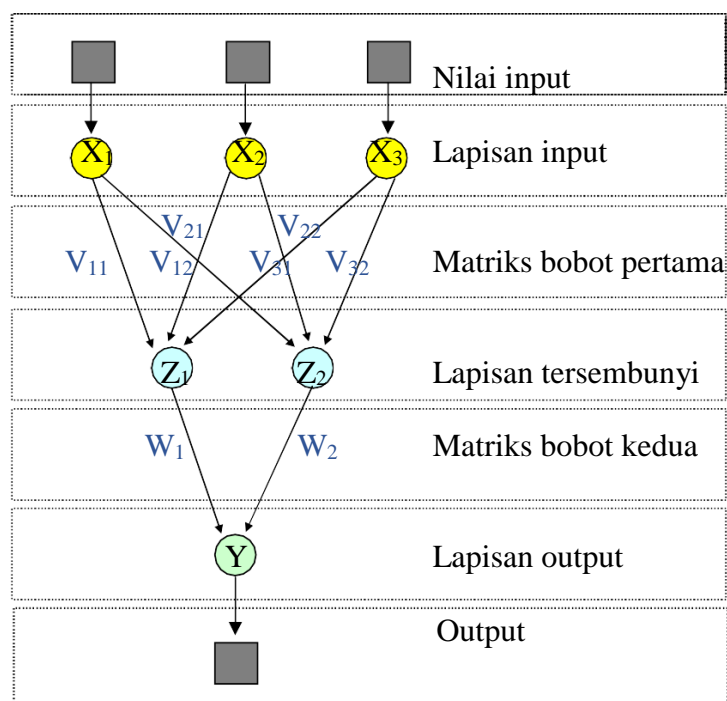
Seberapa besar hubungan antara 2 neuron ditentukan oleh bobot yang bersesuaian. Semua unit input akan dihubungkan dengan setiap unit output.



Gambar 2.4. Jaringan syaraf dengan lapisan tunggal

2. Jaringan dengan banyak lapisan (*multilayer net*)

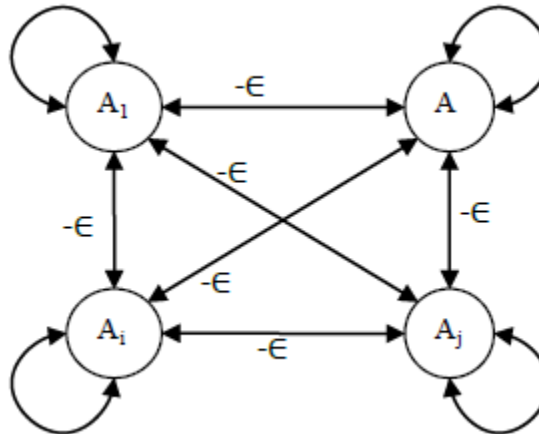
Jaringan dengan banyak lapisan memiliki satu atau lebih lapisan tersembunyi di antara lapisan input dan lapisan output. Umumnya, ada lapisan bobot- bobot yang terletak antara 2 lapisan yang bersebelahan. Jaringan ini dapat menyelesaikan permasalahan yang lebih sulit daripada jaringan dengan lapisan tunggal.



Gambar 2.5. Jaringan syaraf dengan banyak lapisan

3. Jaringan dengan lapisan kompetitif (*competitive layer net*)

Jaringan dengan lapisan kompetitif merupakan jaringan saraf tiruan yang sangat besar. Jaringan ini disebut juga *feedback loop* karena ada unit output yang memberikan informasi terhadap unit input.



Gambar 2.6. Jaringan syaraf dengan lapisan kompetitif

2.4.3. Fungsi Aktivasi

Ada beberapa fungsi aktivasi yang dapat digunakan dalam jaringan syaraf tiruan :

1. Fungsi Undak Biner (*Hard Limit*)

Jaringan syaraf dengan lapisan tunggal sering menggunakan fungsi undak untuk mengkonversikan input dari suatu variabel yang bernilai kontinu ke suatu output biner (0 atau 1).

$$f(x) = \begin{cases} 0, & x \leq 0 \\ 1, & x > 0 \end{cases} \quad (2.2)$$

2. Fungsi Undak Biner (*Threshold*)

Fungsi undak biner dengan menggunakan nilai ambang sering juga disebut dengan nama fungsi nilai ambang (*threshold*).

$$f(x) = \begin{cases} 0, & x < \theta \\ 1, & x \geq \theta \end{cases} \quad (2.3)$$

3. Fungsi Bipolar (*Symetric Hard Limit*)

Fungsi ini hampir sama dengan fungsi undak biner, hanya saja output yang dihasilkan berupa 1, 0 atau -1.

$$f(x) = \begin{cases} 1, & x > 0 \\ 0, & x = 0 \\ -1, & x < 0 \end{cases} \quad (2.4)$$

4. Fungsi Bipolar (dengan *Threshold*)

Fungsi ini hampir sama dengan fungsi undak biner dengan *threshold*, hanya saja output yang dihasilkan berupa 1, 0 atau -1.

$$f(x) = \begin{cases} 1, & x > 0 \\ 0, & x = 0 \\ -1, & x < 0 \end{cases} \quad (2.5)$$

5. Fungsi Linear

Fungsi linear memiliki nilai output yang sama dengan nilai inputnya.

$$f(x) = x \quad (2.6)$$

6. Fungsi Sigmoid Biner

Fungsi ini digunakan untuk jaringan syaraf yang dilatih dengan menggunakan metode *backpropagation*. Fungsi ini memiliki nilai pada range 0 sampai 1.

$$f(x) = \frac{1}{1+e^{-x}} \quad (2.7)$$

7. Fungsi Sigmoid Bipolar

Fungsi ini hampir sama dengan fungsi sigmoid biner, hanya saja output dari fungsi ini memiliki range antara -1 sampai 1.

$$f(x) = \frac{1-e^{-x}}{1+e^{-x}} \quad (2.8)$$

2.5. Support Vector Machine

Support Vector Machine(SVM) adalah sistem pembelajaran yang menggunakan ruang hipotesis berupa fungsi-fungsi linier dalam sebuah ruang fitur (*feature space*) berdimensi tinggi, dilatih dengan algoritma pembelajaran yang didasarkan pada teori optimasi dengan mengimplementasikan learning biasyang berasal dari teori pembelajaran statistik. Teori yang mendasari SVM sendiri sudah berkembang sejak 1960-an, tetapi baru diperkenalkan oleh Vapnik, Boser dan Guyon pada tahun 1992 dan sejak itu SVM berkembang dengan pesat. SVM adalah salah satu teknik yang relatif baru dibandingkan dengan teknik lain, tetapi memiliki performansi yang lebih baik di berbagai bidang aplikasi seperti bioinformatics, pengenalan tulisan tangan, klasifikasi teksdan lain sebagainya.

2.5.1. Struktural Risk Minimization(SRM)

Proses pembelajaran pada SVM bertujuan untuk mendapatkan hipotesis berupa bidang pemisah terbaik yang tidak hanya meminimalkan *empirical risk* yaitu rata-rata error pada data pelatihan, tetapi juga memiliki generalisasi yang baik. Generalisasi adalah kemampuan sebuah hipotesis untuk mengklasifikasikan data yang tidak terdapat dalam data pelatihan dengan benar. Untuk menjamin generalisasi ini, SVM bekerja berdasarkan prinsip SRM.

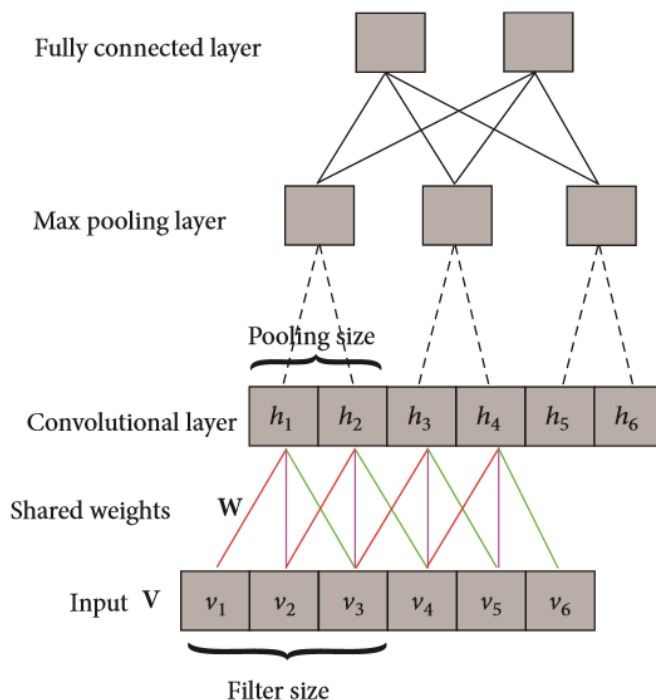
SRM bertujuan untuk menjamin batas atas dari generalisasi pada data pengujian dengan cara mengontrol "kapasitas" (*fleksibilitas*) dari hipotesis hasil pembelajaran. Untuk mengukur kapasitas ini digunakan dimensi *Vapnik-Chervonenkis (VC)* yang merupakan properti dari ruang hipotesis $\{ \alpha \in \mathcal{H} \}$. Nilai dari dimensi VC ini, berdasarkan teori pembelajaran statistik akan menentukan besarnya nilai kesalahan hipotesis pada data pengujian. Lebih jelasnya, besar kesalahan pada data pengujian/ actual risk $R(\alpha)$ dengan probabilitas sebesar $1 - \eta$, $0 < \eta < 1$, pada dataset yang terdiri dari n data dapat dilihat pada persamaan (2.9). $R_{emp}(\alpha)$ adalah kesalahan pada data pelatihan dan h adalah dimensi VC.

$$R(\alpha) \leq R_{emp}(\alpha) + \sqrt{\frac{h \left(\log\left(\frac{2l}{h}\right) + 1 \right) - \log\left(\frac{\eta}{4}\right)}{n}} \quad (2.9)$$

Nilai VC confidence (nilai elemen kedua pada ruas kanan (2.9)), ditentukan oleh hipotesis/ fungsi hasil pembelajaran [BUR98]. Jadi, prinsip SRM adalah menemukan subset dari ruang hipotesis yang dipilih sehingga batas atas *actual risk* dengan menggunakan subset tersebut diminimumkan. SRM bertujuan untuk meminimumkan *actual risk* dengan cara meminimumkan kesalahan pada data pelatihan dan juga VC confidence. Namun, implementasi SRM tidak dilakukan dengan meminimumkan persamaan (2.9) karena dimensi VC dari ruang hipotesis $\{ \alpha \in \mathcal{H} \}$ sulit untuk dihitung dan hanya terdapat sedikit model hipotesis yang diketahui bagaimana cara menghitung dimensi VC-nya [OSU97]. Selain itu, walaupun dimensi VC dapat dihitung, tidak mudah meminimumkan persamaan (2.9). Implementasi SRM pada SVM menggunakan fungsi linier dan akan dijelaskan pada bagian selanjutnya.

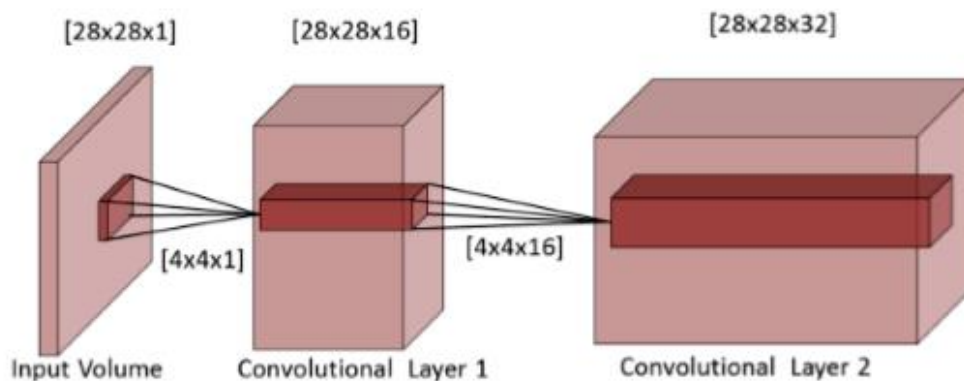
2.6. Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) adalah merupakan salah satu jenis *neural network* yang berisi kombinasi beberapa *layer* yaitu *convolutional layer*, *pooling layer*, dan *fully connected layer* (Hu, et al. 2015). *Convolutional Layer* memproses data dengan topologi *grid* (Goodfellow, et al. 2016). *Convolutional Neural Network* menggunakan operasi *convolution* pada perkalian matriks di setiap *layer*. Arsitektur *Convolutional Neural Network (CNN)* dapat dilihat pada Gambar 2.2. dimana jaringan ini terdiri dari beberapa *layer*, yakni *convolutional layer*, *pooling layer* dan *fully-connected layer*.



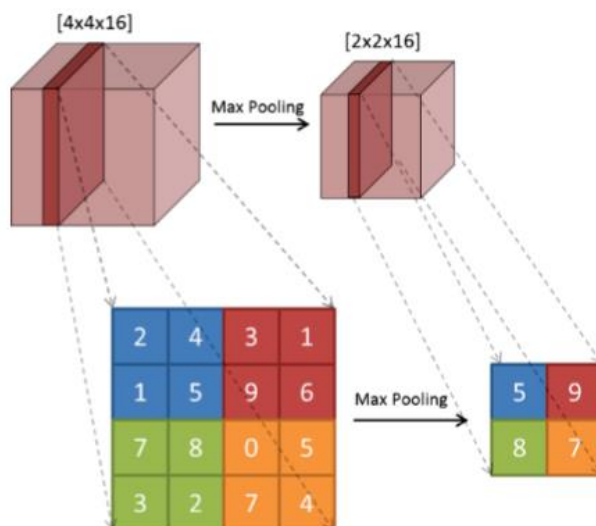
Gambar 2.7. Arsitektur Convolutional Neural Network (Hu, et al. 2015)

Convolutional Layer adalah sebuah inti utama dari CNN, dimana *layer* ini memiliki sebuah kumpulan *filter* yang dapat digunakan untuk mempelajari citra masukan. Melalui *layer* ini, fitur akan di ekstraksi dan kemudian di lanjutkan ke *layer* berikutnya dengan tujuan untuk mengekstraksi fitur yang lebih kompleks (Bui & Chang, 2016). Contoh diagram *Convolutional Layer* dapat dilihat pada Gambar 2.8. dimana ukuran citra masukan yang diberikan adalah 28x28 dan filter atau *kernel* 4x4.



Gambar 2.8. Contoh Diagram Convolutional Layer (Bui & Chang, 2016)

Pooling Layer merupakan proses *resizing* yaitu proses untuk mengubah ukuran citra *input* yang berbeda, salah satunya dengan menggunakan operasi *MAX*. Hal ini bertujuan untuk membantu mengurangi jumlah parameter dan waktu perhitungan yang dibutuhkan saat melatih *network* (Bui & Chang, 2016). Contoh diagram *Pooling Layer* dapat dilihat pada Gambar 2.9.



Gambar 2.9. Contoh Diagram MAX Pooling Layer (Bui & Chang, 2016)

Pada Gambar 2.9. citra yang di masukkan berukuran 4×4 kemudian di *resize* menjadi citra berukuran 2×2 dengan kedalaman masing-masing bernilai 16. Pada Max Pooling, untuk setiap 4 pixels akan diambil satu nilai maksimum. Terlihat pada Gambar 2.9. pada 4 pixels berwarna biru, nilai maksimum yang akan di ambil adalah 5. Pada 4 pixels berwarna merah, nilai maksimum yang akan di ambil adalah 9. Pada pixels

berwarna hijau, nilai maksimum yang akan di ambil adalah 8. Pada pixels berwarna orange, nilai maksimum yang akan di ambil adalah 7. Sehingga menghasilkan sebuah citra yang telah diperkecil.

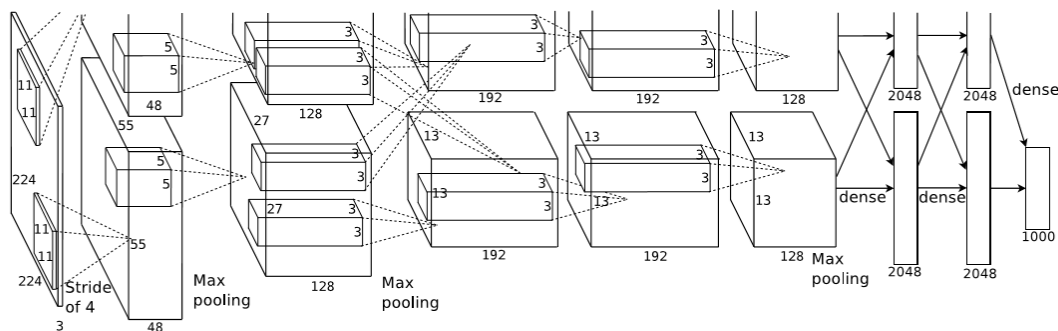
Layer ketiga pada CNN adalah *Fully Connected Layer*, dimana *layer* ini mengambil seluruh neuron pada *layer* sebelumnya (*Convolutional Layer* dan *MAX Pooling Layer*) dan menghubungkannya ke setiap *single neuron* yang ada (Devikar, 2016).

2.7. Deep Convolutional Neural Network (DCNN)

Deep Convolutional Neural Network (DCNN) menunjukan performa yang luar biasa dalam bidang *image recognition*, hal ini didukung dari performa DCNN yang sangat baik dalam mengekstraksi *high-level features*. Selain itu, *Convolutional Layer* dan *MAX Pooling Layer* yang digunakan pada DCNN terbukti sangat efektif dalam mengenali bentuk yang bervariasi. DCNN mampu melakukan seluruh tahap pada pengenalan citra yaitu tahap *feature extraction* dan *classifier* secara bersamaan karena DCNN menerima *raw image* sebagai *input*, sehingga tidak membutuhkan tahap ekstraksi fitur dan *pre-processing* secara terpisah seperti pada *Conventional Classifier* (Kim & Xie, 2015).

Arsitektur *Deep Convolutional Neural Network* dapat dilihat pada Gambar 2.10. dimana terdapat lima *Convolutional Layer* pada *layer* awal dan tiga *layer* berikutnya adalah *Fully Connected Layer* di akhir (Krizhevsky, et al. 2012). *Output* dari *Fully Connected Layer* adalah 1000-way-softmax dan menghasilkan distribusi 1000 kelas label. *Convolutional Layer* yang pertama melakukan *filter* pada citra masukkan yang memiliki ukuran 224x224x3 dengan 96 *kernel* yang memiliki ukuran 11x11x3 dengan *stride* 4 pixels, yaitu jarak antara *receptive field* dari neuron tetangga dalam *kernel map*. Hasil dari *Convolutional Layer* yang pertama menjadi masukkan pada *Convolutional Layer* kedua. Pada *Convolutional Layer* kedua, dilakukan *filter* dengan 256 *kernel* yang memiliki ukuran 5x5x48. *Convolutional Layer* ketiga, keempat dan kelima terhubung satu sama lain tanpa *intervensi pooling* atau normalisasi *layer*. *Convolutional Layer* ketiga memiliki 384 *kernels* berukuran 3x3x256. *Convolutional Layer* keempat memiliki 384 *kernels* berukuran 3x3x192. *Convolutional Layer* kelima memiliki 256 *kernels* berukuran 3x3x192. Setelah proses *Convolutional Layer* selesai,

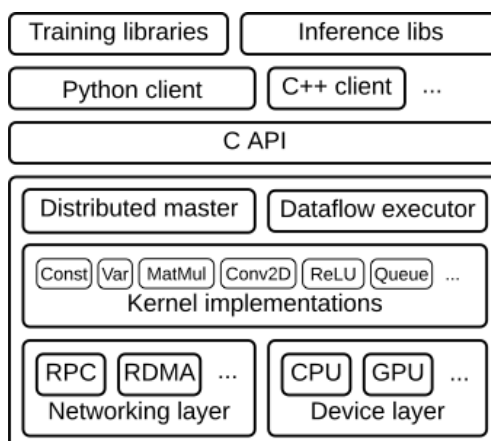
dihasilkan *Fully Connected Layer* yang memiliki 4096 neuron (Krizhevsky, et al. 2012).



Gambar 2.10. Arsitektur Deep Convolutional Neural Network
(Krizhevsky, et al. 2012)

2.8. TensorFlow

TensorFlow adalah *open source library* untuk *machine learning* yang di release oleh Google yang mendukung beberapa bahasa pemrograman (Devikar, 2016). Dalam proses *Transfer Learning*, *Tensorflow* berperan untuk memproses *Inception-v3 Model* untuk di training ulang menggunakan data yang baru dan kemudian menghasilkan *classifier* dengan komputasi yang cepat dan akurasi yang baik. *Tensorflow* dapat digunakan pada semua sistem operasi. Arsitektur umum dari *Tensorflow* dapat dilihat pada Gambar 2.11.



Gambar 2.11. Arsitektur Umum Tensorflow (www.tensorflow.org)

2.9. Penelitian Terdahulu

Telah dilakukan beberapa penelitian yang berkaitan dengan masalah yang diteliti oleh peneliti dengan metode yang berbeda-beda. Penelitian tentang pendeteksian manusia telah dilakukan (*Jianpeng Zhoun & Jack Hoang, 2005*), dengan melakukan pengurangan *background* untuk memisahkan antara manusia dan objek lainnya dan mengikuti pergerakan manusia tersebut.

Penelitian yang dilakukan oleh (*Nishu singla, 2014*) menggunakan metode static background dengan menganalisis perbedaan tiap frame untuk mendeteksi adanya pergerakan objek dalam sebuah background.

Sreedevi et al dalam penelitiannya memanfaatkan perbandingan frame (Sreedevi et al, 2012) untuk mendeteksi sebuah gerakan dalam video kemudian mendeteksi apakah gerakan tersebut dilakukan manusia dan kemudian menyimpan video tersebut.

Naouar Belghini dalam penelitiannya mengenali wajah menggunakan *3D depth information* dimana ujung hidung untuk setiap objek yang dikenali sebagai manusia sebagai dasar koordinat (Belghini et al, 2012).

Farhan Kahn dalam penelitiannya menggunakan *robust mechanism* untuk mendeteksi gerakan dan menganalisisnya. Mereka meminta objek manusia menggunakan sarung tangan yang berbeda pada kedua tangannya. Sehingga mereka mendeteksi pergerakan manusia dari pergerakan tangan.

Edwards et al dalam penelitiannya memanfaatkan wajah seseorang dengan berbagai macam ekspresi untuk menjadi data latih dan melakukan perbandingan terhadap data uji.

Tabel 2.1. Penelitian Terdahulu

No	Nama Peneliti	Tahun	Metode Penelitian	Akurasi
1	Jianpeng Zhoun & Jack Hoang	2005	Robust Human Detection	-
2	Nishu Singla	2014	Frame Diffrence Method	-
3	Sreedevi M, Yaswanth Kumar Avulapati, Anjan Babu G, Sendhil Kumar R	2012	Digital Signal Processor	-

4	Naouar Belghini, Arsalane Zarghilli, Jamal Kharroubi	2012	Gaussian Hermit Momments	82%
5	Farhan S. Khan, Salman A. Baset	2002	Robust Mechanism	Hanya tangan 90%, Tangan dan Kaki 65%
6	G.J. Edwards, T .F . Cootes, and C.J. T aylor	2006	Active Appearance Models	88%
7	Taigman et al.	2014	Deep Neural Network	Akurasi pengenalan wajah: 97% (LFW dataset)
8	Schroff et al.	2015	Deep Convolutional Network	Akurasi pengenalan wajah: 98,87% (LFW dataset)
9	Amos et al.	2016	Convolutional Neural Network	Akurasi pengenalan wajah: 92,92% (LFW dataset)

BAB 3

METODE PENELITIAN

Bab ini akan membahas tentang metode penelitian yang akan diterapkan pada sistem pendeteksian manusia dan pengenalan wajah untuk peningkatan kinerja kamera pengaman. Pada bab ini akan dibahas tentang data, metode pendekatan serta evaluasi yang digunakan pada penelitian ini.

3.1. Data Penelitian

Data yang digunakan pada penelitian ini adalah berupa data citra digital. Pada pendeteksian manusia data diperoleh melalui pengambilan video secara realtime dengan menggunakan webcam. Sebagai bahan pembandingan untuk penelitian pendeteksian manusia diambil juga data video menggunakan sebuah kamera konvensional. Sementara untuk data pengenalan wajah diambil citra digital dari orang-orang yang berkaitan dengan kepemilikan kamera pengaman tersebut. Data pembandingan untuk pengenalan wajah tersebut diperoleh melalui pengambilan citra digital secara realtime dengan menggunakan webcam. Dalam penelitian ini penulis menggunakan webcam dengan kualitas 3 MP dalam kondisi cahaya 960 lux.

Setelah data video tersebut diperoleh akan dilakukan proses pembandingan antara waktu mulai dari video kamera konvensional dengan webcam dimana video yang waktu perekamannya lebih dahulu dilakukan dibandingkan dengan kamera konvensional akan dihapuskan. Begitu juga dengan kamera konvensional. Sehingga diperoleh data yang akan dibandingkan efisiensi media penyimpanan.

3.2. Arsitektur Umum

Metode yang diajukan pada penelitian ini dapat dilihat pada Gambar 3.2. yang menunjukkan arsitektur umum dan rangkaian langkah yang dilakukan untuk melakukan pendeteksian manusia dan pengenalan wajah dari kamera secara real-

time. Rangkaian langkah yang dimaksud adalah sebagai berikut : inisialisasi kamera yang digunakan sebagai alat untuk pengambilan data, mendeteksi wajah pada *video stream*, ekstraksi fitur menggunakan landmark point dan classification menggunakan *Deep Neural Network*. Sementara untuk pendeteksian manusia, video stream yang digunakan untuk pengenalan wajah akan diproses juga dengan *Support Vector Machine (SVM)* dan akan mendeteksi apakah ada manusia atau tidak. Setelah rangkaian langkah di atas maka akan dilakukan perekaman video apabila dideteksi manusia dan jika tidak terdeteksi maka akan berhenti.

3.2.1. Human Detection

Human Detection adalah proses pendeteksian manusia yang ada didepan dari pada kamera yang nantinya akan dipakai untuk membantu perekaman oleh kamera. Ada beberapa sub proses didalam human detection yang akan dijelaskan sebagai berikut :

1. Grayscale

Video yang diterima dari camera mengandung banyak warna yang nantinya akan sulit untuk mendeteksi tiap piksel-pikselnya karena memiliki warna yang berbeda-beda. Maka dari itu diperlukan proses grayscale agar nilai dari intensitas komponen merah, hijau dan biru menjadi sama. Sehingga mendeteksi tiap piksel hanya dengan menyatakan nilai intensitasnya sebagai nilai tunggal. Untuk rumus pemrosesan *grayscale* yang akan digunakan adalah berikut :

$$grayscale = 0.299R + 0.587G + 0.114B \quad (3.1)$$

$$grayscale = (0,299 \times 155) + (0,587 \times 120) + (0,114 \times 135)$$

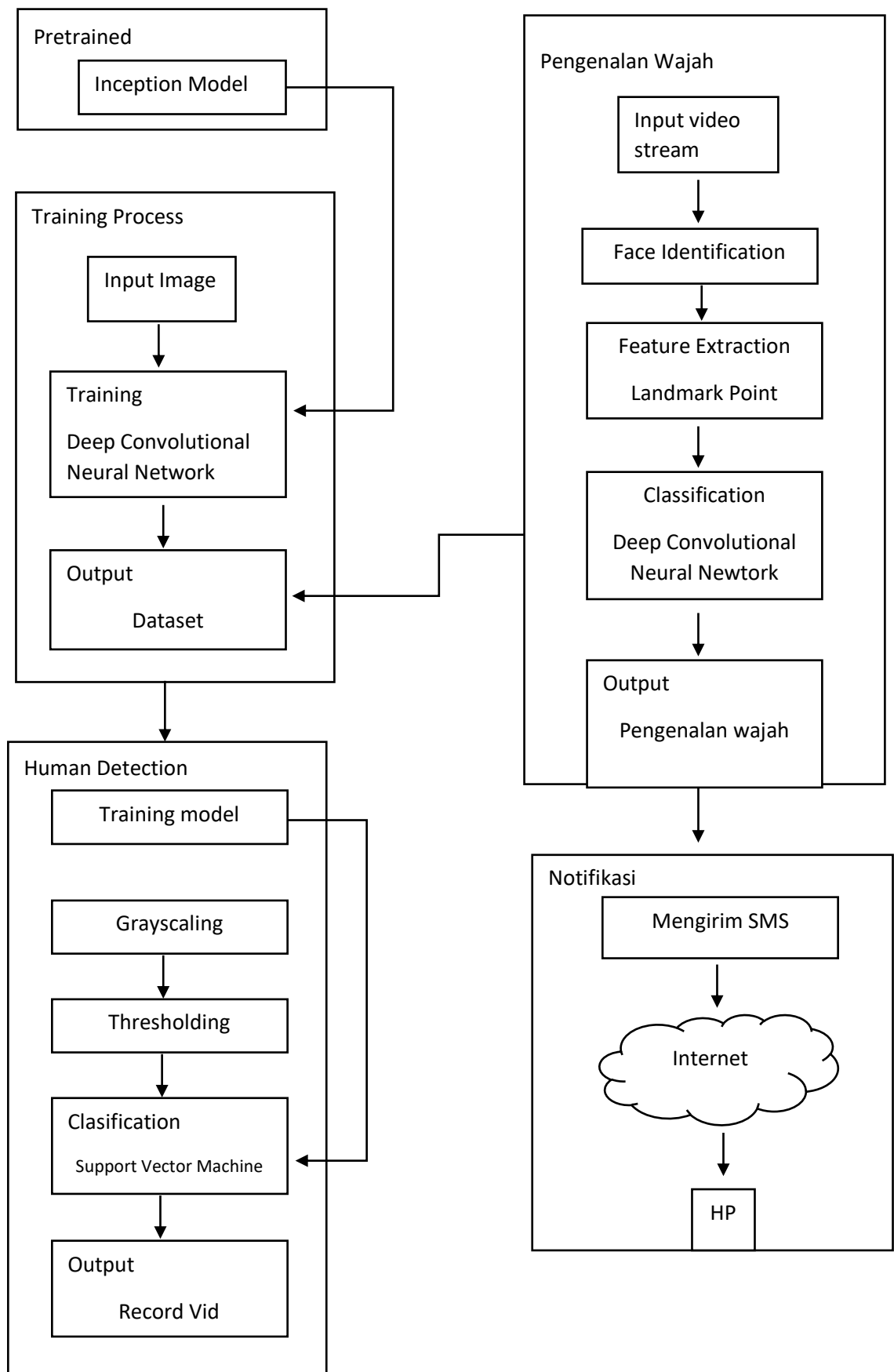
$$grayscale = 46,345 + 70,44 + 15,39$$

$$grayscale = 132,175$$

Contoh citra hasil grayscale bisa dilihat pada Gambar 3.1.



Gambar 3.1. Contoh Citra Grayscale



Gambar 3.2. Arsitektur Umum

2. Thresholding

Thresholding adalah proses merubah citra abu-abu menjadi citra biner. Pada proses thresholding citra objek akan dibedakan dengan backgroundnya dengan cara menandai nilai piksel milik objek dan membandingkan dengan nilai thresholdnya. Dengan cara ini maka akan bisa dideteksi apakah objek ada pada frame saat itu atau tidak. Contoh citra hasil thresholding bisa dilihat pada Gambar 3.3



Gambar 3.3 Contoh Citra Threshold

Citra yang telah dithreshold akan melalui proses dilatasi untuk mendapatkan posisi objek yang tepat. Setelah proses tersebut maka akan dicari kontur dari object dengan dengan fungsi dan akan dihasilkan nilai dari titik yang menjadi lokasi kontur.

$$C(T) = M_1(T) \cdot \sigma_1^2(T) + M_2(T) \cdot \sigma_2^2(T) \quad (3.2)$$

Dimana M_1 adalah jumlah pixel

M_2 adalah sisa pixel dari image

σ_1^2 dan σ_2^2 adalah nilai varian dari pixel

$$C = ((16 \cdot 16) \cdot 0.4637) + ((16 \cdot 16) \cdot 0.5152)$$

$$C = 118.7072 + 131.8912$$

$$C = 250.5894$$

Maka nilai dari threshold untuk proses tersebut adalah 250. Dimana tiap nilai piksel yang lebih besar dari C akan dirubah jadi 255 sementara untuk lebih kecil dari C akan menjadi 0.



```
[array([[515, 86]], [[514, 87]], [[514, 92]], [[517, 95]], [[523, 95]], [[524, 94]], [[524, 93]], [[525, 92]], [[525, 87]], [[524, 86]], dtype=int32), array([[568, 67]], [[567, 68]], [[567, 74]], [[568, 75]], [[569, 75]], [[570, 76]], [[574, 76]], [[574, 74]], [[576, 72]], [[576, 68]], [[575, 67]], dtype=int32)]
```

Gambar 3.4 Gambar Threshold dan Posisi Kontur

Pada Gambar 3.4 terdeteksi hasil dari threshold gambar dimana ditampilkan titik dari nilai hitam atau area gelap dari hasil threshold.

3. Classification

Setelah citra melalui proses-proses sebelumnya proses berikutnya adalah *classification* dimana pada poses ini akan dihasilkan nilai apakah pada citra tersebut terdapat objek manusia atau tidak. Pada *classification* akan digunakan metode support vector machine(SVM). Metode SVM ini yang nantinya akan melakukan proses terhadap nilai dari hasil thresholding. Nilai kontur yang telah diperoleh maka dilakukan proses terhadap nilai tersebut untuk mengetahui keberadaan manusia. Apabila terdapat manusia maka akan dilakukan proses perhitungan lokasi terletak manusia pada frame.



Gambar 3.5 Manusia Terdeteksi dan Posisi pada Frame

Pada Gambar 3.5 terdapat manusia dengan posisi hasil perhitungan dari nilai kontur yang telah didapat. Setelah manusia tersebut terdeteksi maka akan dilakukan perekaman dengan waktu hasil perekaman menjadi 25 detik. Kemudian setelah itu akan dilakukan pengecekan kembali apakah manusia tetap ada atau tidak.

3.2.2. Pengenalan Wajah

Proses pengenalan wajah adalah proses yang digunakan untuk mengenali manusia yang berada didepan kamera. Untuk subproses dari pengenalan wajah dapat dijelaskan sebagai berikut :

1. Training

Pelatihan yang dilakukan menggunakan *Deep Convolutional Neural Network* dengan tahapan pelatihan berjumlah 4000 *steps*. Dimana setiap tahapan akan menampilkan *training accuracy*, *validation accuracy* dan *cross entropy*. Pada proses pelatihan, citra masukan berukuran 224x224x3 pertama kali akan diproses oleh *Convolutional Layer* Pertama dengan filter 96 kernel berukuran 11x11x3. Proses *Convolutional Layer* Pertama dapat dilihat pada Gambar 3.6 dan untuk perhitungan bisa dilihat pada persamaan 3.3

$$S_{(i,j)} = (K * I)(i, j) = \sum \sum I(i - m, j - n)K(m, n) \quad (3.3)$$

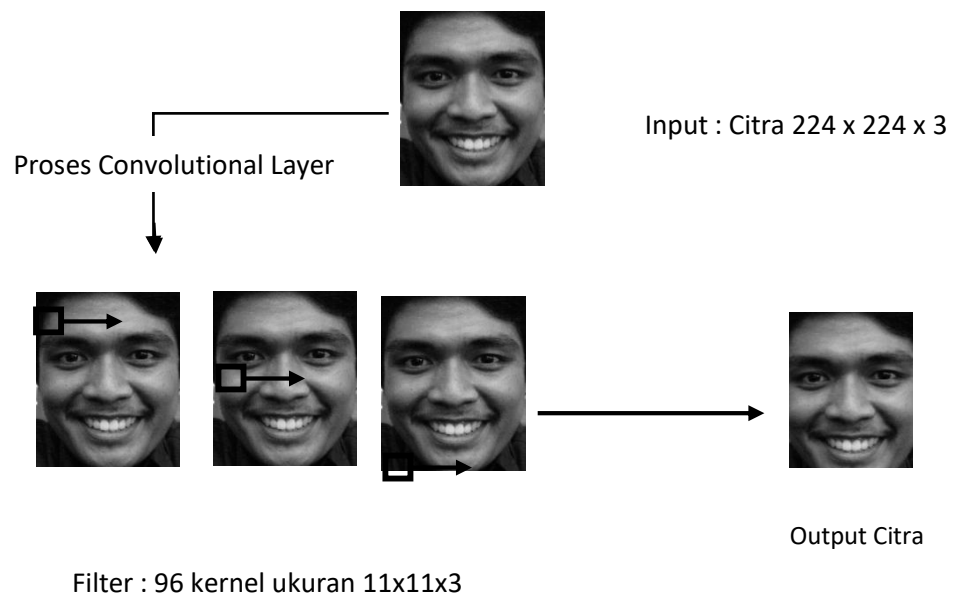
$$S_{(i,j)} = \begin{bmatrix} 128 & 245 \\ 255 & 190 \end{bmatrix} \begin{bmatrix} 3 & 3 \\ 3 & 3 \end{bmatrix}$$

$$S_{(i,j)} = (128 * 3) + (245 * 3) + (255 * 3) + (190 * 3)$$

$$S_{(i,j)} = 384 + 735 + 765 + 570$$

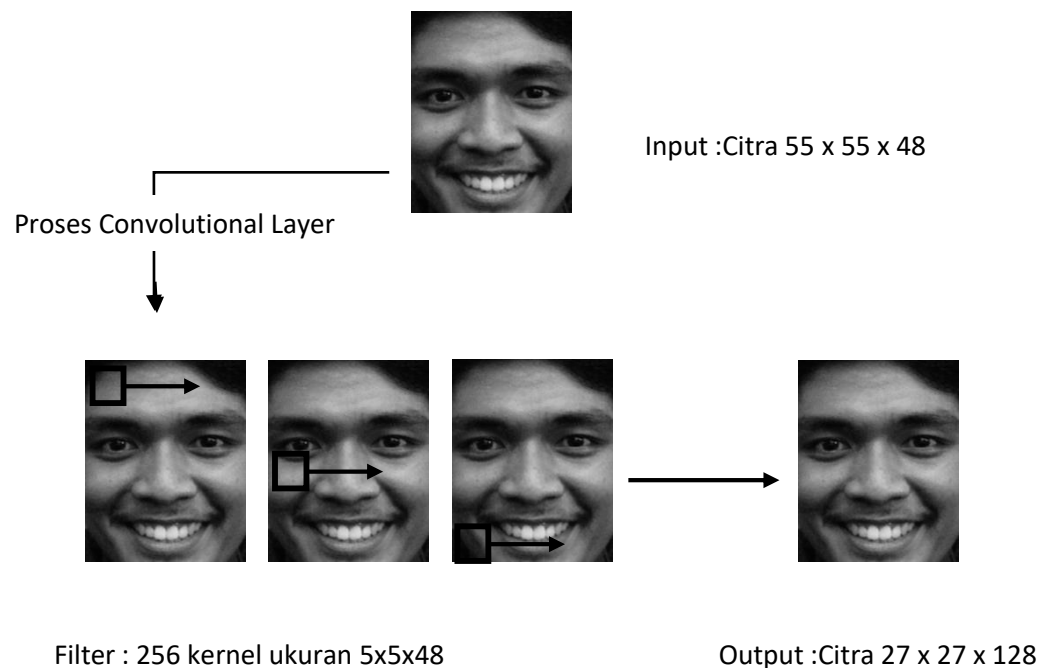
$$S_{(i,j)} = 2454$$

Maka nilai hasil convulasi salah satu pixel pada layer pertama adalah 2454 yang akan menjadi pixel pada convulasi pada layer kedua.



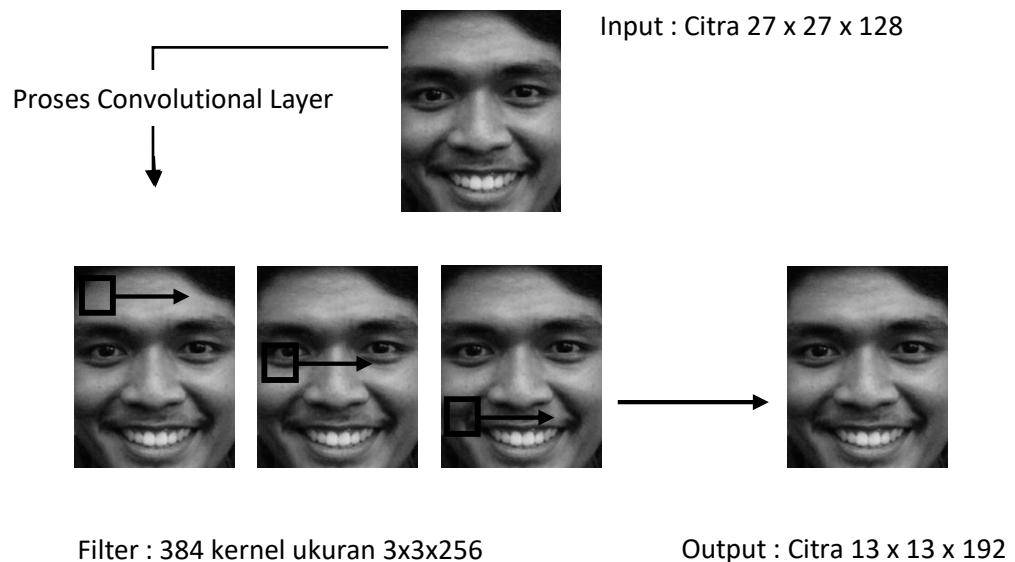
Gambar 3.6. Proses *Convolutional Layer* Pertama

Kemudian, hasil dari *Convolutional Layer* Pertama berukuran 55x55x48 akan di proses oleh *Convolutional Layer* Kedua dengan filter 256 kernel berukuran 5x5x48. Proses *Convolutional Layer* Kedua dapat dilihat pada Gambar 3.7.



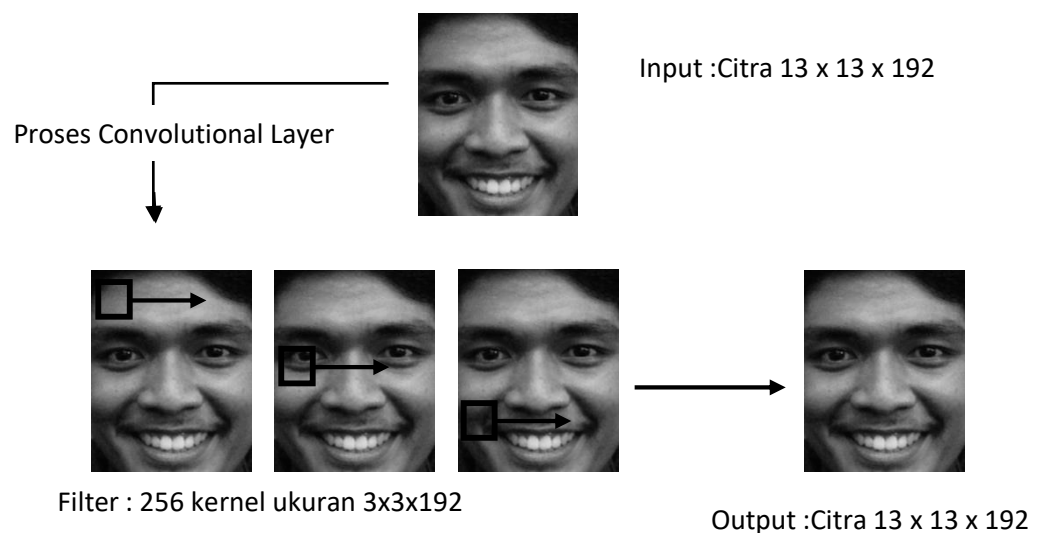
Gambar 3.7. Proses *Convolutional Layer* Kedua

Convolutional Layer Ketiga, Keempat dan Kelima terhubung satu sama lain tanpa *intervensi pooling* atau *normalisasi layer*. Hasil dari *Convolutional Layer* Kedua akan di proses oleh *Convolutional Layer* Ketiga dengan filter 256 kernel berukuran $3 \times 3 \times 192$. Proses *Convolutional Layer* Ketiga dapat dilihat pada Gambar 3.8.



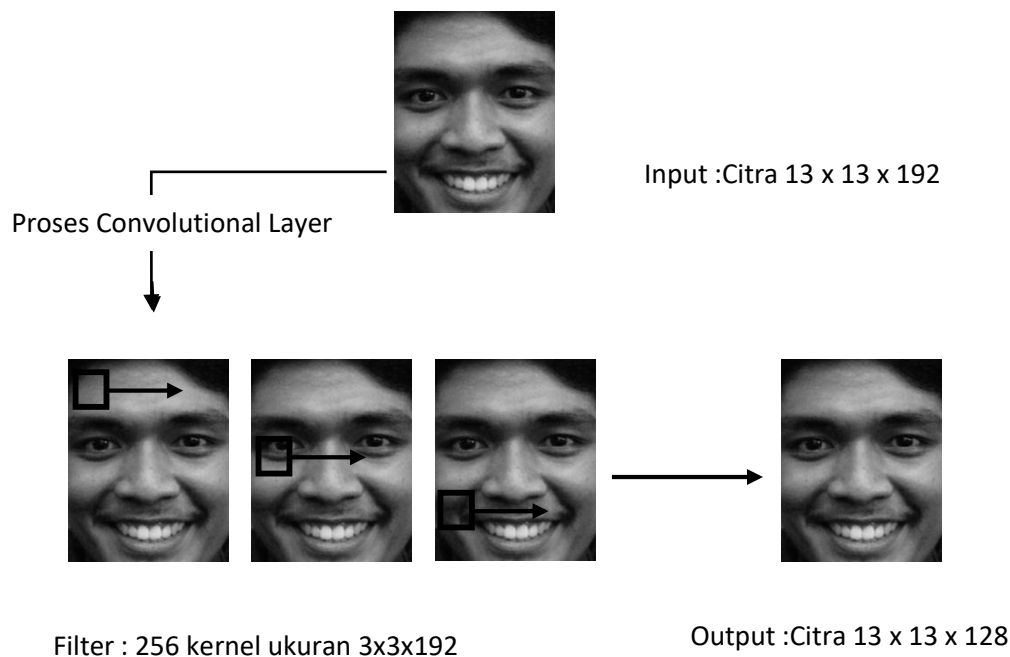
Gambar 3.8. Proses *Convolutional Layer* Ketiga

Citra masukan pada *Convolutional Layer* Keempat berukuran $13 \times 13 \times 192$. Akan di proses dengan filter 256 kernel berukuran $3 \times 3 \times 192$ seperti pada Gambar 3.9.



Gambar 3.9. Proses *Convolutional Layer* Keempat

Citra masukan pada *Convolutional Layer* Kelima memiliki ukuran $13 \times 13 \times 192$. Akan di proses dengan filter 256 kernel berukuran $3 \times 3 \times 192$. Proses *Convolutional Layer* Kelima dapat dilihat pada Gambar 3.10.



Gambar 3.10. Proses *Convolutional Layer* Kelima

Setelah proses Convolutional Layer, maka dihasilkan 3 *fully connected layers* yang memiliki 4096 neuron pada tiap *layer*. Hasil dari *fully connected layer* terakhir adalah 1000-way-softmax. Proses Pelatihan Setiap Layer menggunakan *Deep Convolutional Neural Network* dapat dilihat pada Tabel 3.1.

Tabel 3.1. Tabel Proses Pelatihan Setiap Layer

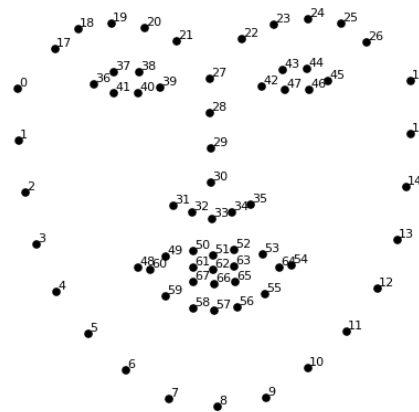
Layer	Kernel	Ukuran	Jumlah Neuron
Citra Masukan	-	224 x 224	150.528
		x 3 x 1	
<i>Convolutional Layer</i> Pertama	96	55 x 55 x 48 x 2	290.400
<i>Convolutional Layer</i> Kedua	256	27 x 27 x 128 x 2	186.624

<i>Convolutional</i>	384	13 x 13 x	64.896
<i>Layer Ketiga</i>		192 x 2	
<i>Convolutional</i>	384	13 x 13 x	64.896
<i>Layer Keempat</i>		192 x 2	
<i>Convolutional</i>	256	13 x 13 x	43.264
<i>Layer Kelima</i>		128 x 2	
<i>Fully-Connected</i>	-	2048 x 2	4.096
<i>Layer</i>			
<i>Fully-Connected</i>	-	2048 x 2	4.096
<i>Layer</i>			
<i>Fully-Connected</i>	-	1000 x 1	1000
<i>Layer (Softmax</i>			
<i>Output)</i>			

Output yang dihasilkan dari proses training ini adalah berupa graph dengan ekstensi file (.pb) yang nantinya akan dipakai untuk melakukan pengenalan terhadap wajah manusia yang ada didepan kamera.

2. Ekstraksi Fitur

Pada pengenalan wajah tahapan yang awal adalah mengambil nilai ekstraksi dari gambar. Untuk ekstraksi fitur digunakan metode *landmark point*. Dengan landmark point maka akan diambil titik-titik posisi dari hidung, rahang, pelipis dan mulut . Titik-titik tersebut berupa nilai x dan y dimana wajah akan menjadi bidangnya. Posisi point pada wajah dapat dilihat pada Gambar 3.11. Dari gambar tersebut dapat dilihat terdapat 64 titik posisi di wajah manusia.



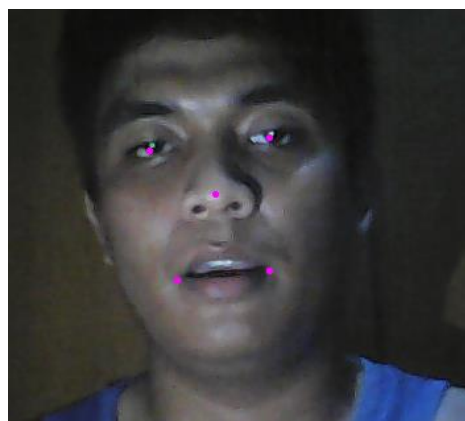
Gambar 3.11. Titik Posisi pada Wajah

Citra yang diinput akan dideteksi posisi wajah manusia dari citra tersebut seperti terlihat pada Gambar 3.12.



Gambar 3.12. Posisi Wajah Manusia

Setelah wajah terdeteksi maka akan dilakukan proses landmark point dimana sistem akan mendeteksi titik-titik pada wajah yang nantinya akan dipakai untuk menjadi pembanding terhadap dataset yang telah disimpan pada saat training seperti pada Gambar 3.13.



Gambar 3.13 Contoh Landmark Point Pada Wajah

3. Classification

Pada classification nilai yang didapat dari pada hasil ekstraksi fitur akan di klasifikasikan untuk menyatakan objek tersebut dikenal atau tidak. Untuk klasifikasi digunakan metode deep convolutional neural network(DCNN). Pada DCNN hasil ekstraksi fitur tadi akan dilakukan perbandingan dengan data set yang merupakan hasil dari training sebelumnya. Hasil ekstraksi fitur sebelumnya berupa posisi titik wajah akan dibandingkan kedalam dataset yang telah diperoleh dari hasil training.

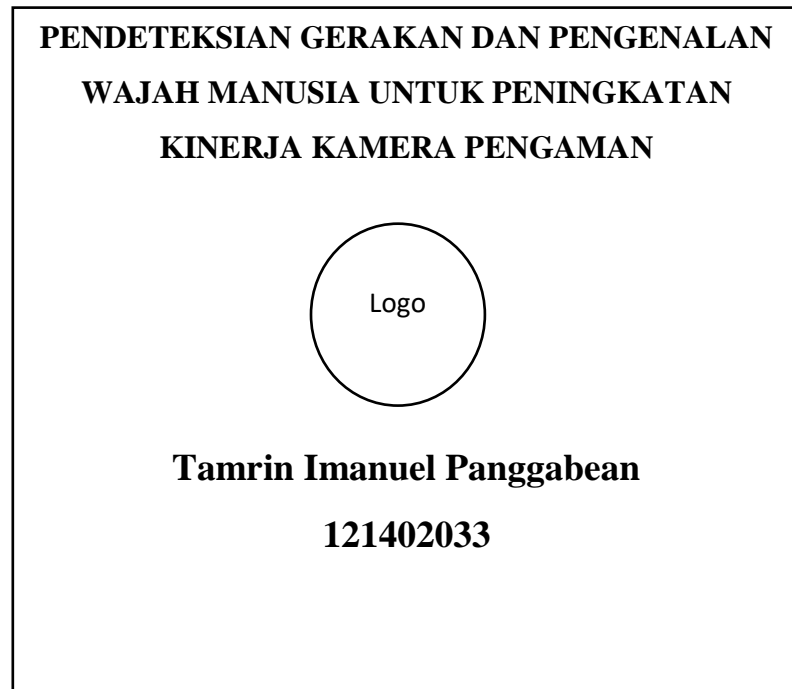
Dataset pada tensorflow disimpan dalam bentuk *(.pb)* yang akan menyimpan informasi dari hasil ekstraksi fitur sebelumnya. Hasil dari proses ini adalah berupa nama dari pemilik wajah yang telah disimpan dan apabila wajah tersebut tidak dikenali maka di layar akan ditampilkan *unknown*. Proses klasifikasi dapat dilakukan terhadap lebih dari 1 wajah sekaligus dalam 1 frame video dimana data hasil ekstraksi tadi disimpan dalam sebuah data *array* yang nantinya akan dicek bersamaan kedalam dataset.

Disaat sistem mengembalikan nilai dikenali maka sistem tidak melakukan proses apapun sementara disaat sistem mengembalikan nilai unknown maka sistem secara otomatis akan mengirimkan notifikasi ke pemilik rumah berupa sms. Pengiriman SMS dibantu dengan sebuah provider penyedia *sms gateway*.

3.3. Perancangan Aplikasi

a. Rancangan Halaman Awal Sistem

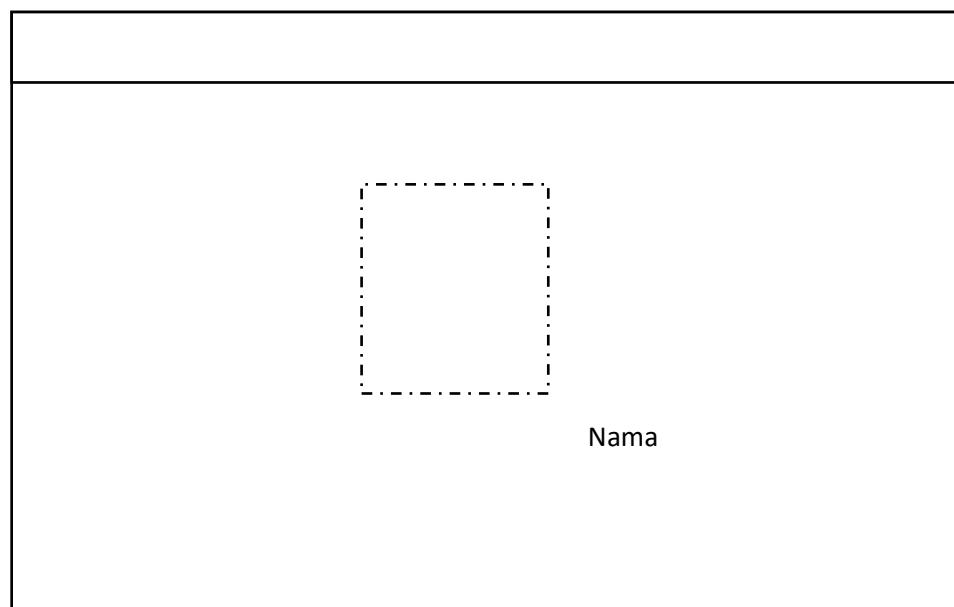
Rancangan halaman awal sistem menampilkan nama sistem pada bagian atas, logo, nama dan NIM pada bagian tengah. Rancangan tampilan awal sistem akan ditunjukkan pada Gambar 3.14.



Gambar 3.14. Rancangan halaman awal sistem

b. Rancangan Halaman Utama

Halaman utama berguna untuk menampilkan video stream dari camera. Pada halaman ini akan ditampilkan hasil deteksi wajah dan juga hasil deteksi manusia



Gambar 3.15. Rancangan Halaman Utama

c. Rancangan Pemberitahuan

Disaat terdeteksi ada orang yang tidak dikenal pada video stream maka akan dikirimkan notifikasi berupa sms ke orang tertentu.

Pengirim : 08xxx
Terdapat orang tidak dikenal didepan rumah anda

Gambar 3.16. Perancangan Isi Notifikasi berupa SMS

3.4. Metode Pengujian

Tahapan uji coba berfungsi untuk menguji apakah sistem yang dibangun sudah berjalan dengan baik atau belum dengan membandingkan hasil yang diperoleh dari sistem dengan beberapa parameter. Pada penelitian ini untuk menguji sistem pendeteksian manusia dan pengenalan wajah untuk peningkatan kinerja kamera pengaman, dilakukan beberapa percobaan dengan membandingkan hasil dari sistem dengan beberapa kondisi percobaan, diantaranya :

- A. Untuk pengefisiensian media penyimpanan dilakukan pengujian pembandingan kamera konvensional dengan webcam dengan tambahan sistem selama 6 jam perekaman.
- B. Pengujian untuk pendeteksian manusia dan pengenalan wajah dengan perbedaan intensitas cahaya mulai dari 5 lux, 200 lux, 750 lux, 1000 lux.
- C. Pengujian untuk pendeteksian manusia dan pengenalan wajah dengan perbedaan jarak ke posisi kamera dan dibandingkan dengan objek bukan manusia.
- D. Pengujian untuk pengenalan wajah dengan sudut wajah yang berbeda. Kondisi tersebut dimaksudkan untuk mengukur efisiensi dari media penyimpanan kamera serta akurasi dari sistem yang dibuat.

3.5. Metode Evaluasi

Setelah uji coba dilakukan maka akan tahap berikutnya adalah melakukan evaluasi terhadap sistem yang telah dibuat. Untuk melakukan evaluasi ada beberapa aspek yang diperhatikan oleh penulis. Evaluasi pertama adalah terhadap akurasi yang diberikan oleh sistem. Untuk perhitungan akurasi dapat menggunakan persamaan 3.2.

$$\text{Akurasi} = \frac{\text{Jumlah Hasil Uji Coba Yang Benar}}{\text{Total Uji Coba}} \times 100\% \quad (3.2)$$

Sementara untuk efisiensi media penyimpanan bisa dihitung dengan menggunakan perhitungan:

$$\text{Eff} = \frac{\text{ESS} - \text{NSS}}{\text{ESS}} \times 100\% \quad (3.3)$$

Eff adalah hasil efisiensi dari kamera konvensional ke webcam dengan sistem.

ESS adalah ukuran media penyimpanan yang diperlukan oleh kamera konvensional.

NSS adalah ukuran media penyimpanan yang diperlukan oleh webcam dengan sistem.

Metode untuk evaluasi sistem yang dibangun pada penelitian adalah dengan menggunakan rumus umum perhitungan *precision*, *recall*, dan *F-score* pada persamaan 3.8, 3.9, dan 3.10 dengan menerapkan aturan variabel pada Gambar 3.6.

		Kondisi Aktual	
		<i>Dikenal</i>	<i>Tidak Dikenal</i>
Hasil identifikasi	<i>Dikenal</i>	<i>True Positive (TP)</i>	<i>False Positive (FP)</i>
	<i>Tidak Dikenal</i>	<i>False Negative (FN)</i>	<i>True Negative (TN)</i>

Gambar 3.17. Variabel perhitungan *F-score*

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (3.4)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (3.5)$$

$$\text{F-score} = \frac{2 \times (\text{Precision} \times \text{Recall})}{\text{Precision} + \text{Recall}} \quad (3.16)$$

Keterangan :

- *True Positive* (TP) adalah kondisi dimana kondisi aktual dikenal berhasil diidentifikasi oleh sistem sebagai *dikenal*.
- *False Positive* (FP) adalah kondisi dimana kondisi aktual *tidak dikenal* gagal diidentifikasi oleh sistem sebagai *tidak dikenal*.
- *False Negative* (FN) adalah kondisi dimana kondisi aktual *dikenal* gagal diidentifikasi oleh sistem sebagai *dikenal*.
- *True Negative* (TN) adalah kondisi dimana kondisi aktual *tidak dikenal* berhasil diidentifikasi oleh sistem sebagai *tidak dikenal*.

BAB 4

IMPLEMENTASI DAN HASIL PENGUJIAN SISTEM

Bab ini akan membahas tentang hasil yang didapat dari implementasi pendeteksian dan pengenalan wajah manusia untuk peningkatan kinerja dari kamera pengaman yang telah dibahas pada Bab 3.

4.1. Implementasi Sistem

Pada tahapan ini pendeteksian dan pengenalan wajah akan diterapkan pada kamera pengaman. Dalam pembuatan sistem menggunakan bahasa pemrograman Python yang diintegrasikan dengan *Webcam*.

4.1.1. Spesifikasi hardware dan software

Pada penelitian ini spesifikasi hardware (perangkat keras) dan software(perangkat lunak) yang dipakai dalam pembangunan sistem ini adalah :

- a. *Processor Intel(R) Core(TM) i5-A456U CPU @ 2.8 GHz*
- b. *Memori RAM 4 GB DDR4*
- c. *Kapasitas hard disk 1TB HDD*
- d. *Sistem operasi Windows 10 Pro 64-bit*
- e. *JetBrains PyCharm Community Edition 2017.1.2 x64*
- f. *Python 3.5*
- g. *Webcam 3MP*
- h. *CCTV VIVOTEK 10 MP*

4.1.2. Implementasi Sistem

a. Pendeteksian Manusia

Hasil implementasi untuk pendeteksian manusia dapat dilihat pada bagian dibawah ini:

1. Thresholding

Pada pendeteksian manusia sistem akan melakukan thresholding kepada video stream inputan. Dengan membandingkan daerah terang dan gelap maka daerah yang nilai

thresholdnya lebih dari 1,1 maka akan dirubah warna pikselnya menjadi hitam. Video yang diinput tersebut maka di threshold kedalam fungsi threshold setelah itu maka hasil dari threshold dikirim untuk mendeteksi kontur dari manusia yang ada.

2. Classification

Setelah video stream melalui proses threshold dan deteksi kontur maka sistem akan memberikan nilai dimana lokasi manusia tersebut. Sistem akan membandingkan dengan area yang ada dan apabila terdeteksi manusia maka akan diberikan kotak yang menunjukkan manusia.

b. Pengenalan wajah

1. Training

Sistem akan menerima data training berupa foto wajah dari orang yang akan dikenali sebagai anggota pemilik rumah. Dengan metode tensorflow training dilakukan dengan sebuah model yang nantinya akan dipakai sebagai pendeteksi wajah dari manusia. Model tersebut dibuka dan dibaca untuk struktur wajah manusia. Setelah itu dibaca image yang menjadi data training. Kemudian data perwajah akan diekstraksi dengan landmark point yang kemudian dilakukan penyimpanan kedalam graph. Graph ini yang nantinya akan menjadi data pembanding antara wajah yang ada didepan kamera dengan yang sudah ditraining.

2. Feature Extraction

Feature extraction yang dipakai pada aplikasi adalah landmark point. Landmark point akan melakukan pemrosesan terhadap pixel wajah kemudian akan mendeteksi lokasi wajah. Wajah yang terdeteksi didalam frame video kemudian dikirim kedalam face_patches setelah itu akan dilakukan pendeteksian terhadap aligned dari wajah tersebut.

3. Classification

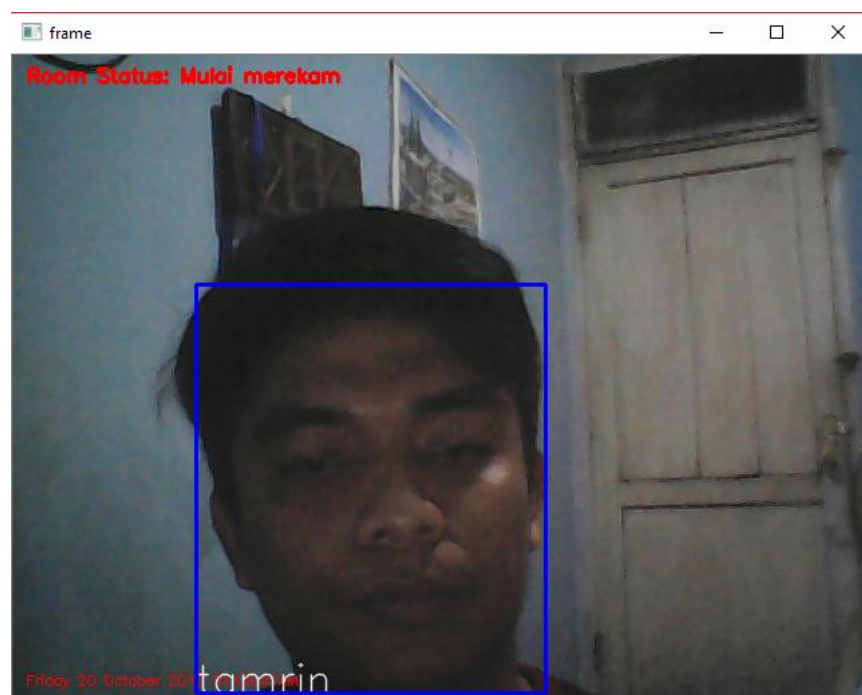
Hasil dari ekstraksi akan dikirim untuk dilakukan pengenalan wajah dari dataset. Data dari hasil feature extraction dicek kedalam dataset yang telah didapat dari training sebelumnya. Kemudian jika dideteksi data tersebut matching maka sistem tersebut maka akan diset matching_id ke id dari pemilik wajah tersebut.

4.1.3. Implementasi Perancangan Antarmuka

Antarmuka yang dirancang terdiri dari 2 tampilan desktop dan 1 tampilan notifikasi, yaitu:

1. Tampilan Halaman Utama

Halaman utama dari aplikasi ini adalah halaman untuk menampilkan hasil capture video secara realtime dari webcam. Pada halaman ini akan ditampilkan apabila terdapat manusia dan pengenalan wajah dari manusia. Untuk halaman utama bisa dilihat pada Gambar 4.1.



Gambar 4.1 Halaman Utama Aplikasi

Dimana sistem akan melakukan pendeteksian wajah dan manusia kemudian dilakukan ekstraksi dan setelah itu dilakukan perbandingan terhadap *data training* yang telah disimpan sebelumnya. Setelah terdeteksi manusia lalu akan disimpan sebagai video dan jika terdeteksi wajah dan dikenali maka wajah tersebut akan dikotaki dengan warna biru kemudian akan ditampilkan hasil nama yang terdeteksi. Apabila tidak dikenali maka akan keluar *Unknown* dan akan dikirimkan sms notifikasi.

2. Isi Notifikasi Berupa SMS

Disaat terdapat orang tidak dikenal didepan rumah maka akan ada notifikasi ke pemilik rumah bahwa terdapat orang tidak dikenal didepan rumahnya. Berikut contoh notifikasi yang masuk pada Gambar 4.2.



Gambar 4.2. Isi Notifikasi SMS

Setelah terdeteksi orang yang tidak dikenal maka akan dikirim notifikasi berupa sms kepada pemilik rumah.

4.2. Hasil Pengujian Sistem

Pengujian sistem telah dilakukan dengan memperhitungan beberapa kondisi yang berbeda sesuai pada bagian 3.4 untuk mengetahui akurasi dan efisiensi dari sistem yang telah dibuat. Percobaan dilakukan dengan terpisah antara Pendeteksian Manusia dan Pengenalan Wajah. Berikut hasil percobaan tersebut :

4.2.1. Pendeteksian Manusia

Pada proses ini akan dilakukan perbandingan kinerja dari kamera CCTV konvensional dengan sistem yang dibuat. Dilakukan perekaman selama 8 jam dengan sudut, waktu dan kondisi lingkungan yang sama. Contoh pendeteksian manusia bisa dilihat pada Gambar 4.3. Pengujian tidak hanya dilakukan pada objek manusia tetapi juga terhadap tumbuhan dan juga hewan. Untuk contoh pengujian objek selain manusia dapat dilihat pada Gambar 4.4 dan Gambar 4.5.



Gambar 4.3. Perekaman Saat Terdeteksi Manusia



Gambar 4.4. Percobaan dengan Tumbuhan Tinggi



Gambar 4.5. Percobaan dengan Kucing

Sesuai dengan Gambar 4.4 dan Gambar 4.5 sistem berhasil untuk tidak mendeteksi bahwa yang ada didepan kamera adalah bukan manusia. Dimana Gambar 4.4 tumbuhan yang memiliki tinggi hampir mirip dengan manusia tidak dideteksi sebagai manusia dan untuk Gambar 4.5 objek yang bergerak dan adalah kucing dideteksi sebagai bukan manusia.

1. Pengujian dengan kondisi Intensitas Cahaya Berbeda

Pada proses ini dilakukan pengujian terhadap beberapa kondisi cahaya yang berbeda. Dimulai dari 5 lux – 1000 lux.

a. Intensitas Cahaya 5 Lux

Pengujian pada intensitas cahaya 5 lux diambil pada saat jam 18.00 dan untuk hasil pengujian bisa dilihat pada tabel 4.1. Untuk kondisi lingkungan pada intensitas cahaya 5 lux dapat dilihat pada Gambar 4.6.



Gambar 4.6. Kondisi Lingkungan Intensitas Cahaya 5 Lux

Tabel 4.1. Pengujian Pendeteksian Manusia Dengan Cahaya 5 Lux

Nomor Percobaan	Output Aktual	Output Sistem	Time Detection(s)
1	Terdapat Manusia	Tidak Terdapat Manusia	0.0003
2	Tidak Terdapat Manusia	Tidak Terdapat Manusia	0.0004
3	Tidak Terdapat Manusia	Terdapat Manusia	0.0003
4	Tidak Terdapat Manusia	Terdapat Manusia	0.0003
5	Terdapat Manusia	Tidak Terdapat Manusia	0.0003
6	Terdapat Manusia	Terdapat Manusia	0.0004

7	Tumbuhan	Tidak Terdapat Manusia	0.0003
8	Kucing	Tidak Terdapat Manusia	0.0004
9	Tumbuhan	Terdapat Manusia	0.0005
10	Kucing	Tidak Terdapat Manusia	0.0003

$$\text{Akurasi} = \frac{5}{10} \times 100 \%$$

Akurasi Sistem Pada Kondisi 5 Lux = 50 % dengan rata-rata waktu deteksi 0.0003 detik

Kesalahan pada pengujian ini kemungkinan besar disebabkan oleh kurangnya cahaya pada lingkungan. Sehingga membuat ketepatan sistem mendeteksi objek terganggu. Dan juga ini bisa disebabkan karena kamera yang dipakai adalah webcam dengan pixel 3MP.

b. Intensitas cahaya 200 Lux

Pengujian pada intensitas cahaya 200 lux diambil pada saat jam 15.30 dan untuk hasil pengujian bisa dilihat pada tabel 4.2. Untuk kondisi lingkungan dengan intensitas cahaya 200 Lux dapat dilihat pada Gambar 4.7.



Gambar 4.7. Kondisi Lingkungan Intensitas Cahaya 200 Lux

Tabel 4.2. Pengujian Pendeteksian Manusia Dengan Cahaya 200 Lux

Nomor Percobaan	Output Aktual	Output Sistem	Time Detection(s)
1	Terdapat Manusia	Tidak Terdapat Manusia	0.0003
2	Tidak Terdapat Manusia	Tidak Terdapat Manusia	0.0004
3	Tidak Terdapat Manusia	Terdapat Manusia	0.0003
4	Tidak Terdapat Manusia	Tidak Terdapat Manusia	0.0003
5	Terdapat Manusia	Terdapat Manusia	0.0003
6	Terdapat Manusia	Terdapat Manusia	0.0004
7	Tumbuhan	Tidak Terdapat Manusia	0.0003
8	Kucing	Tidak Terdapat Manusia	0.0004
9	Tumbuhan	Tidak Terdapat Manusia	0.0005
10	Kucing	Tidak Terdapat Manusia	0.0003

$$\text{Akurasi} = \frac{8}{10} \times 100 \%$$

Akurasi Sistem Pada Kondisi 200 Lux = 80 % dengan rata-rata waktu deteksi 0.0004 detik. Pada percobaan dengan kondisi cahaya 200 Lux menghasilkan akurasi pendeteksian manusia 80% dengan 10 percobaan.

c. Intensitas cahaya 750 Lux

Pengujian pada intensitas cahaya 750 lux diambil pada saat jam 13.00 dan untuk hasil pengujian bisa dilihat pada tabel 4.3. Untuk lingkungan dengan kondisi intensitas cahaya 750 Lux dapat dilihat pada Gambar 4.8.



Gambar 4.8. Kondisi Lingkungan Intensitas Cahaya 750 Lux

Table 4.3. Pengujian Pendeteksian Manusia Dengan Cahaya 750 Lux

Nomor Percobaan	Output Aktual	Output Sistem	Time Detection(s)
1	Terdapat Manusia	Terdapat Manusia	0.0003
2	Tidak Terdapat Manusia	Tidak Terdapat Manusia	0.0004
3	Tidak Terdapat Manusia	Terdapat Manusia	0.0003
4	Tidak Terdapat Manusia	Tidak Terdapat Manusia	0.0003
5	Terdapat Manusia	Terdapat Manusia	0.0003
6	Terdapat Manusia	Terdapat Manusia	0.0004
7	Tumbuhan	Tidak Terdapat Manusia	0.0003
8	Kucing	Tidak Terdapat Manusia	0.0004
9	Tumbuhan	Tidak Terdapat Manusia	0.0005
10	Kucing	Tidak Terdapat Manusia	0.0003

$$\text{Akurasi} = \frac{9}{10} \times 100 \%$$

Akurasi Sistem Pada Kondisi 750 Lux = 90 % dengan rata-rata waktu deteksi 0.0004 detik. Pada percobaan dengan kondisi cahaya 750 Lux menghasilkan akurasi pendeteksian manusia 90% dengan 10 percobaan.

d. Intensitas cahaya 1000 Lux

Pengujian pada intensitas cahaya 1000 lux diambil pada saat jam 11.00 dan untuk hasil pengujian bisa dilihat pada tabel 4.4. Untuk kondisi lingkungan dengan intensitas cahaya 1000 Lux dapat dilihat pada Gambar 4.9.



Gambar 4.9. Kondisi Lingkungan Intensitas Cahaya 1000 Lux

Tabel 4.4. Pengujian Pendeteksian Manusia Dengan Cahaya 1000Lux

Nomor Percobaan	Output Aktual	Output Sistem	Time Detection(s)
1	Terdapat Manusia	Tidak Terdapat Manusia	0.0003
2	Tidak Terdapat Manusia	Tidak Terdapat Manusia	0.0004
3	Tidak Terdapat Manusia	Tidak Terdapat Manusia	0.0003
4	Tidak Terdapat Manusia	Tidak Terdapat Manusia	0.0003
5	Terdapat Manusia	Terdapat Manusia	0.0003
6	Terdapat Manusia	Terdapat Manusia	0.0004
7	Tumbuhan	Tidak Terdapat Manusia	0.0003
8	Kucing	Tidak Terdapat Manusia	0.0004
9	Tumbuhan	Tidak Terdapat Manusia	0.0005
10	Kucing	Tidak Terdapat Manusia	0.0003

$$\text{Akurasi} = \frac{9}{10} \times 100 \%$$

Akurasi Sistem Pada Kondisi 1000 Lux = 90 % dengan rata-rata waktu deteksi 0.0004 detik. Pada percobaan dengan kondisi cahaya 1000 Lux menghasilkan akurasi pendeteksian manusia 90% dengan 10 percobaan.

Tabel 4.5. Hasil Perhitungan F-Score Pendeteksian Manusia Dengan Perbedaan Intensitas Cahaya

	Kondisi Percobaan Cahaya			
	5 Lux	200 Lux	750 Lux	1000 Lux
TP	1	2	3	2
FN	2	1	0	1
FP	3	1	1	0
TN	4	6	6	7
<i>Precision</i>	25	66,6	75	100
<i>Recall</i>	33,3	66,6	100	66,6
F-score	28,5	66,6	85,7	79.9

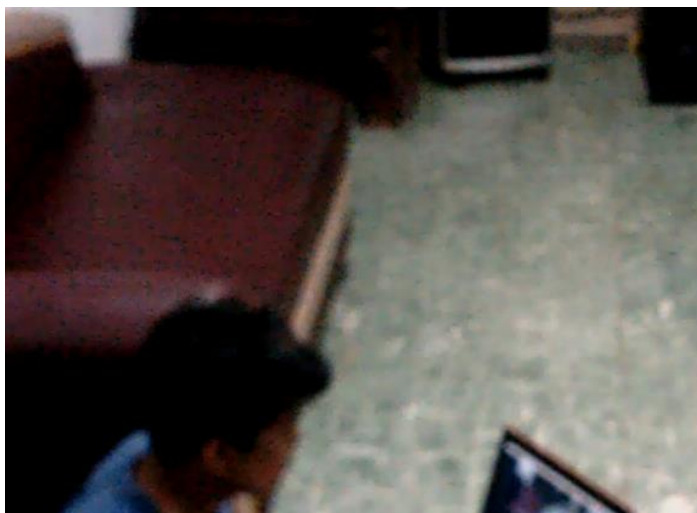
Maka dari hasil uji coba dengan intensitas cahaya yang berbeda proses pendeteksian manusia menghasilkan akurasi yang berbeda dengan kondisi 5 lux sebagai akurasi yang terendah dan pada akurasi 750 lux sudah bisa mendeteksi dengan akurasi yang sama dengan 1000 lux. Untuk penilaian F-Score dari tabel 4.5 nilai yang terendah berada pada posisi intensitas cahaya 5 lux dan untuk F-Score tertinggi berada pada kondisi intensitas cahaya 750 lux.

2. Pengujian dengan jarak yang berbeda

Pada proses ini sistem akan diuji dengan posisi jarak manusia yang berbeda mulai dari 1 meter, 2 meter dan 5 meter.

a. Jarak 1 meter

Pada pengujian ini akan dilakukan dengan meletakkan objek dengan jarak 1 meter dari kamera. Hasil ujicoba dapat dilihat pada tabel 4.6 dan untuk contoh objek pada jarak 1 meter dapat dilihat pada Gambar 4.10.



Gambar 4.10. Objek pada Jarak 1 Meter dari Kamera

Tabel 4.6. Pengujian Pendeteksian Manusia Pada Kondisi 1 Meter

Nomor Percobaan	Output Aktual	Output Sistem	Time Detection(s)
1	Terdapat Manusia	Terdapat Manusia	0.0003
2	Tidak Terdapat Manusia	Tidak Terdapat Manusia	0.0004
3	Tidak Terdapat Manusia	Terdapat Manusia	0.0003
4	Tidak Terdapat Manusia	Tidak Terdapat Manusia	0.0003
5	Terdapat Manusia	Terdapat Manusia	0.0003
6	Terdapat Manusia	Terdapat Manusia	0.0004
7	Tumbuhan	Tidak Terdapat Manusia	0.0003
8	Kucing	Tidak Terdapat Manusia	0.0004
9	Tumbuhan	Tidak Terdapat Manusia	0.0005
10	Kucing	Tidak Terdapat Manusia	0.0003

$$\text{Akurasi} = \frac{9}{10} \times 100 \%$$

Akurasi Sistem Pada Kondisi jarak 1 meter = 90 % dengan rata-rata waktu 0.0004 detik. Pada jarak pengujian dengan jarak objek dari kamera 1 meter didapatkan akurasi dari sistem adalah 90 %.

b. Jarak 2 meter

Pada pengujian ini akan dilakukan dengan meletakkan objek dengan jarak 2 meter dari kamera. Hasil ujicoba dapat dilihat pada tabel 4.7 dan untuk contoh objek pada jarak 2 meter dapat dilihat pada Gambar 4.11.



Gambar 4.11. Objek pada Jarak 2 Meter dari Kamera

Tabel 4.7. Pengujian Pendeteksian Manusia Pada Kondisi 2 Meter

Nomor Percobaan	Output Aktual	Output Sistem	Time Detection(s)
1	Terdapat Manusia	Terdapat Manusia	0.0003
2	Tidak Terdapat Manusia	Tidak Terdapat Manusia	0.0004
3	Tidak Terdapat Manusia	Terdapat Manusia	0.0003
4	Tidak Terdapat Manusia	Tidak Terdapat Manusia	0.0003
5	Terdapat Manusia	Terdapat Manusia	0.0003
6	Terdapat Manusia	Terdapat Manusia	0.0004
7	Tumbuhan	Tidak Terdapat Manusia	0.0003
8	Kucing	Tidak Terdapat Manusia	0.0004
9	Tumbuhan	Tidak Terdapat Manusia	0.0005
10	Kucing	Tidak Terdapat Manusia	0.0003

$$\text{Akurasi} = \frac{9}{10} \times 100 \%$$

Akurasi Sistem Pada Kondisi jarak 2 meter = 90 % dengan rata-rata waktu deteksi 0.0003 detik. Pada jarak pengujian dengan jarak objek dari kamera 2 meter didapatkan akurasi dari sistem adalah 90 %.

c. Jarak 5 meter

Pada pengujian ini akan dilakukan dengan meletakkan objek dengan jarak 5 meter dari kamera. Hasil ujicoba dapat dilihat pada tabel 4.8 dan untuk contoh objek pada jarak 5 meter dapat dilihat pada Gambar 4.12.



Gambar 4.12. Objek pada Jarak 5 Meter dari Kamera

Tabel 4.8. Pengujian Pendeteksian Manusia Pada Kondisi 5 Meter

Nomor Percobaan	Output Aktual	Output Sistem	Time Detection(s)
1	Terdapat Manusia	Tidak Terdapat Manusia	0.0003
2	Tidak Terdapat Manusia	Terdapat Manusia	0.0004
3	Tidak Terdapat Manusia	Terdapat Manusia	0.0003
4	Tidak Terdapat Manusia	Tidak Terdapat Manusia	0.0003
5	Terdapat Manusia	Terdapat Manusia	0.0003
6	Terdapat Manusia	Terdapat Manusia	0.0004
7	Tumbuhan	Tidak Terdapat Manusia	0.0003
8	Kucing	Tidak Terdapat Manusia	0.0004
9	Tumbuhan	Tidak Terdapat Manusia	0.0005
10	Kucing	Tidak Terdapat Manusia	0.0003

$$\text{Akurasi} = \frac{7}{10} \times 100 \%$$

Akurasi Sistem Pada Kondisi jarak 5 meter = 70 % dengan rata-rata waktu deteksi 0.0003 detik. Pada jarak pengujian dengan jarak objek dari kamera 5 meter didapatkan akurasi dari sistem adalah 70 %.

Tabel 4.9. Hasil Perhitungan F-Score Pendeteksian Manusia Dengan Perbedaan Jarak

	Kondisi Percobaan Jarak		
	1 Meter	2 Meter	5 Meter
TP	3	3	2
FN	0	0	1
FP	1	1	2
TN	6	6	5
<i>Precision</i>	75	75	50
<i>Recall</i>	100	100	66,6
F-score	85,7	85,7	57,1

Dari hasil ujicoba dengan jarak yang berbeda tersebut maka dapat disimpulkan pada saat jarak 2 meter akurasi masih menjadi yang terbaik. Penurunan akurasi pada jarak 5 meter diakibatkan karena jarak yang semakin jauh antara objek dan kamera. Untuk penilaian F-Score sesuai tabel 4.9 maka nilai tertinggi berada pada jarak 1 meter dan 2 meter sementara terendah pada jarak 5 meter.

4.2.2. Pengenalan Wajah

Pada proses ini akan dilakukan pengujian sistem untuk mengenali wajah manusia yang berada didepan kamera dengan kondisi yang berbeda-beda.

1. Pengujian dengan kondisi intensitas cahaya berbeda

Pada proses ini dilakukan pengujian pengenalan wajah manusia pada beberapa kondisi cahaya yang berbeda. Dimulai dari 5 lux – 1000 lux.

a. Intensitas Cahaya 5 Lux

Pengujian pada intensitas cahaya 5 lux diambil pada saat jam 18.00 dan untuk hasil pengujian bisa dilihat pada tabel 4.10. Untuk kondisi lingkungan pada intensitas cahaya 5 lux dapat dilihat pada Gambar 4.6.

Tabel 4.10. Pengujian Pengenalan Wajah Dengan Cahaya 5 Lux

Nomor Percobaan	Output Aktual	Output Sistem	Time Detection(s)
1	Dikenal	Dikenal	0.0003
2	Tidak Dikenal	Tidak Dikenal	0.0004
3	Tidak Dikenal	Tidak Dikenal	0.0003
4	Dikenal	Tidak Dikenal	0.0003
5	Dikenal	Dikenal	0.0003
6	Dikenal	Tidak Dikenal	0.0004
7	Tumbuhan	Tidak Terdeteksi	0.0003
8	Kucing	Tidak Terdeteksi	0.0004
9	Tumbuhan	Tidak Terdeteksi	0.0005
10	Kucing	Tidak Terdeteksi	0.0003

$$\text{Akurasi} = \frac{8}{10} \times 100 \%$$

Akurasi Sistem Pada Kondisi 5 Lux = 80 % dengan rata-rata waktu 0.0004 detik

Pada pengujian pengenalan wajah dengan kondisi intensitas cahaya 5 Lux akurasi ketepatan sistem adalah 80% dengan 10 percobaan.

b. Intensitas cahaya 200 Lux

Pengujian pada intensitas cahaya 200 lux diambil pada saat jam 15.30 dan untuk hasil pengujian bisa dilihat pada tabel 4.11. Untuk kondisi lingkungan dengan intensitas cahaya 200 Lux dapat dilihat pada Gambar 4.7.

Tabel 4.11. Pengujian Pengenalan Wajah Pada Kondisi 200 Lux

Nomor Percobaan	Output Aktual	Output Sistem	Time Detection(s)
1	Dikenal	Dikenal	0.0003
2	Tidak Dikenal	Tidak Dikenal	0.0004
3	Tidak Dikenal	Tidak Dikenal	0.0003
4	Dikenal	Dikenal	0.0003
5	Dikenal	Tidak Dikenal	0.0003
6	Dikenal	Tidak Dikenal	0.0004
7	Tumbuhan	Tidak Terdeteksi	0.0003
8	Kucing	Tidak Terdeteksi	0.0004
9	Tumbuhan	Tidak Terdeteksi	0.0005
10	Kucing	Tidak Terdeteksi	0.0003

$$\text{Akurasi} = \frac{8}{10} \times 100 \%$$

Akurasi Sistem Pada Kondisi 200 Lux = 80 % dengan rata-rata waktu 0.0004 detik.

Pada pengujian pengenalan wajah dengan intensitas cahaya 200 Lux memiliki akurasi 80% dari 10 percobaan.

c. Intensitas cahaya 750 Lux

Pengujian pada intensitas cahaya 750 lux diambil pada saat jam 13.00 dan untuk hasil pengujian bisa dilihat pada tabel 4.12. Untuk lingkungan dengan kondisi intensitas cahaya 750 Lux dapat dilihat pada Gambar 4.8.

Table 4.12. Pengujian Pengenalan Wajah Pada Kondisi 750 Lux

Nomor Percobaan	Output Aktual	Output Sistem	Time Detection(s)
1	Dikenal	Dikenal	0.0003
2	Tidak Dikenal	Tidak Dikenal	0.0004
3	Tidak Dikenal	Tidak Dikenal	0.0003
4	Dikenal	Tidak Dikenal	0.0003

5	Dikenal	Dikenal	0.0003
6	Dikenal	Dikenal	0.0004
7	Tumbuhan	Tidak Terdeteksi	0.0003
8	Kucing	Tidak Terdeteksi	0.0004
9	Tumbuhan	Tidak Terdeteksi	0.0005
10	Kucing	Tidak Terdeteksi	0.0003

$$\text{Akurasi} = \frac{9}{10} \times 100 \%$$

Akurasi Sistem Pada Kondisi 750 Lux = 90 % dengan rata-rata waktu 0.0004 detik.

Pada pengujian pengenalan wajah dengan intensitas cahaya 750 Lux memiliki akurasi 90% dari 10 percobaan.

d. Intensitas cahaya 1000 Lux

Pengujian pada intensitas cahaya 1000 lux diambil pada saat jam 11.00 dan untuk hasil pengujian bisa dilihat pada tabel 4.13. Untuk kondisi lingkungan dengan intensitas cahaya 1000 Lux dapat dilihat pada Gambar 4.9.

Tabel 4.13. Pengujian Pengenalan Wajah Pada Kondisi 1000Lux

Nomor Percobaan	Output Aktual	Output Sistem	Time Detection(s)
1	Dikenal	Dikenal	0.0003
2	Tidak Dikenal	Tidak Dikenal	0.0004
3	Tidak Dikenal	Tidak Dikenal	0.0003
4	Dikenal	Dikenal	0.0003
5	Dikenal	Dikenal	0.0003
6	Dikenal	Dikenal	0.0004
7	Tumbuhan	Tidak Terdeteksi	0.0003
8	Kucing	Tidak Terdeteksi	0.0004
9	Tumbuhan	Tidak Terdeteksi	0.0005
10	Kucing	Tidak Terdeteksi	0.0003

$$\text{Akurasi} = \frac{10}{10} \times 100 \%$$

Akurasi Sistem Pada Kondisi 1000 Lux = 100 % dengan rata-rata waktu 0.0003 detik

Tabel 4.14. Hasil Perhitungan F-Score Pengenalan Wajah Dengan Perbedaan Intensitas Cahaya

	Kondisi Percobaan Cahaya			
	5 Lux	200 Lux	750 Lux	1000 Lux
TP	2	2	3	4
FN	2	2	1	0
FP	0	0	0	0
TN	6	6	6	6
<i>Precision</i>	100	100	100	100
<i>Recall</i>	50	50	75	100
F-score	66,6	66,6	85,7	100

Maka dari hasil uji coba dengan intensitas cahaya yang berbeda proses pengenalan wajah manusia menghasilkan akurasi yang hampir sama dengan akurasi minimal 80%. Untuk pengenalan wajah metode yang digunakan mampu mengenali wajah bahkan dalam kondisi cahaya 5 lux. Untuk penilaian F-Score dari tabel 4.14 maka didapatkan bahwa nilai F-Score tertinggi berada pada kondisi cahaya 1000 lux dan terendah pada kondisi cahaya 5 lux dan 200 lux.

2. Pengujian dengan jarak yang berbeda

Pada proses ini sistem akan diuji dengan posisi jarak manusia yang berbeda mulai dari 1 meter, 2 meter dan 5 meter.

a. Jarak 1 meter

Pada pengujian ini akan dilakukan dengan meletakkan objek dengan jarak 1 meter dari kamera. Hasil ujicoba dapat dilihat pada tabel 4.15 dan untuk contoh objek pada jarak 1 meter dapat dilihat pada Gambar 4.10.

Tabel 4.15. Pengujian Pengenalan Manusia Pada Kondisi 1 Meter

Nomor Percobaan	Output Aktual	Output Sistem	Time Detection(s)
1	Dikenal	Dikenal	0.0003
2	Tidak Dikenal	Tidak Dikenal	0.0004

3	Tidak Dikenal	Tidak Dikenal	0.0003
4	Dikenal	Dikenal	0.0003
5	Dikenal	Dikenal	0.0003
6	Dikenal	Dikenal	0.0004
7	Tumbuhan	Tidak Terdeteksi	0.0003
8	Kucing	Tidak Terdeteksi	0.0004
9	Tumbuhan	Tidak Terdeteksi	0.0005
10	Kucing	Tidak Terdeteksi	0.0003

$$\text{Akurasi} = \frac{10}{10} \times 100 \%$$

Akurasi Sistem Pada Kondisi jarak 1 meter = 100 % dengan rata-rata waktu 0.0004 detik. Pada pengujian pengenalan wajah dengan jarak objek 1 meter dari kamera menghasilkan akurasi 100% dari 10 percobaan.

b. Jarak 2 meter

Pada pengujian ini akan dilakukan dengan meletakkan objek dengan jarak 2 meter dari kamera. Hasil ujicoba dapat dilihat pada tabel 4.16 dan untuk contoh objek pada jarak 2 meter dapat dilihat pada Gambar 4.11.

Tabel 4.16. Pengujian Pengenalan Manusia Pada Kondisi 2 Meter

Nomor Percobaan	Output Aktual	Output Sistem	Time Detection(s)
1	Dikenal	Dikenal	0.0003
2	Tidak Dikenal	Tidak Dikenal	0.0004
3	Tidak Dikenal	Tidak Dikenal	0.0003
4	Dikenal	Dikenal	0.0003
5	Dikenal	Dikenal	0.0003
6	Dikenal	Dikenal	0.0004
7	Tumbuhan	Tidak Terdeteksi	0.0003
8	Kucing	Tidak Terdeteksi	0.0004
9	Tumbuhan	Tidak Terdeteksi	0.0005
10	Kucing	Tidak Terdeteksi	0.0003

$$\text{Akurasi} = \frac{10}{10} \times 100 \%$$

Akurasi Sistem Pada Kondisi jarak 2 meter = 100 % dengan rata-rata waktu 0.0003 detik. Pada pengujian pengenalan wajah dengan jarak objek 2 meter dari kamera menghasilkan akurasi 100% dari 10 percobaan.

c. Jarak 5 meter

Pada pengujian ini akan dilakukan dengan meletakkan objek dengan jarak 5 meter dari kamera. Hasil ujicoba dapat dilihat pada tabel 4.17 dan untuk contoh objek pada jarak 5 meter dapat dilihat pada Gambar 4.12.

Tabel 4.17. Pengujian Penngenalannya Manusia Pada Kondisi 5 Meter

Nomor Percobaan	Output Aktual	Output Sistem	Time Detection(s)
1	Dikenal	Dikenal	0.0003
2	Tidak Dikenal	Tidak Dikenal	0.0004
3	Tidak Dikenal	Tidak Dikenal	0.0003
4	Dikenal	Tidak Dikenal	0.0003
5	Dikenal	Dikenal	0.0003
6	Dikenal	Dikenal	0.0004
7	Tumbuhan	Tidak Terdeteksi	0.0003
8	Kucing	Tidak Terdeteksi	0.0004
9	Tumbuhan	Tidak Terdeteksi	0.0005
10	Kucing	Tidak Terdeteksi	0.0003

$$\text{Akurasi} = \frac{9}{10} \times 100 \%$$

Akurasi Sistem Pada Kondisi jarak 5 meter = 90 % dengan rata-rata waktu deteksi 0.0003 detik.

Tabel 4.18. Hasil Perhitungan F-Score Pengenalan Wajah Dengan Perbedaan Jarak




	Kondisi Percobaan Jarak		
	1 Meter	2 Meter	5 Meter
TP	4	4	3
FN	0	0	1
FP	0	0	0
TN	6	6	6
<i>Precision</i>	100	100	100
<i>Recall</i>	100	100	75
F-score	100	100	85,7



Dari hasil ujicoba dengan jarak yang berbeda tersebut maka dapat disimpulkan bahwa hasil akurasi dari metode ini sangat tinggi sekalipun dengan jarak 5 meter. Hasil penilaian F-Score pada tabel 4.18 menunjukkan bahwa jarak 1 meter dan 2 meter menghasilkan F-Score 100 %.

3. Pengujian Dengan Posisi Wajah Berbeda

Pada pengujian ini akan dilakukan pengenalan terhadap wajah manusia dengan beberapa posisi yang berbeda. Untuk posisi wajah yang akan diuji dapat dilihat pada tabel 4.19 dan untuk hasil pengujian masing-masing posisi dapat dilihat pada tabel 4.16.

Tabel 4.19. Posisi Wajah untuk Pengujian

No	Kondisi	Image
1	Menghadap Kamera	 A frontal view of a person's face. A blue bounding box is drawn around the face, and the word 'tamrin' is written in white at the bottom left of the box.
2	Menghadap Kanan	 A right profile view of a person's face. A blue bounding box is drawn around the face, and the word 'tamrin' is written in white at the bottom left of the box.
3	Menghadap Kekiri	 A left profile view of a person's face. A blue bounding box is drawn around the face, and the word 'tamrin' is written in white at the bottom left of the box.

4	Menghadap Keatas	
5	Menunduk	

Tabel 4.20. Pengujian Pengenalan Wajah Dengan Posisi Wajah Berbeda

Nomor Percobaan	Output Aktual	Output Sistem	Posisi Wajah	Time Detection(s)
1	Dikenal	Dikenal	Menghadap Kamera	0.0003
2	Tidak Dikenal	Tidak Dikenal	Menghadap Kamera	0.0004
3	Tidak Dikenal	Tidak Dikenal	Menghadap Kamera	0.0003
4	Dikenal	Dikenal	Menghadap Kamera	0.0003
5	Dikenal	Dikenal	Menghadap Kamera	0.0003
6	Tidak Dikenal	Tidak Dikenal	Menghadap Kamera	0.0004
7	Dikenal	Dikenal	Menghadap Kekan	0.0003
8	Tidak Dikenal	Tidak Dikenal	Menghadap Kekan	0.0004
9	Tidak Dikenal	Tidak Dikenal	Menghadap Kekan	0.0005
10	Dikenal	Tidak Dideteksi	Menghadap Kekan	0.0003
11	Dikenal	Dikenal	Menghadap Kekan	0.0004

12	Tidak Dikenal	Tidak Dikenal	Menghadap Kekanan	0.0005
13	Dikenal	Dikenal	Menghadap Kekiri	0.0004
14	Tidak Dikenal	Tidak Dideteksi	Menghadap Kekiri	0.0003
15	Tidak Dikenal	Tidak Dikenal	Menghadap Kekiri	0.0003
16	Dikenal	Dikenal	Menghadap Kekiri	0.0003
17	Tidak Dikenal	Tidak Dikenal	Menghadap Kekiri	0.0003
18	Dikenal	Dikenal	Menghadap Kekiri	0.0004
19	Dikenal	Dikenal	Menghadap Keatas	0.0003
20	Dikenal	Dikenal	Menghadap Keatas	0.0004
21	Tidak Dikenal	Tidak Dideteksi	Menghadap Keatas	0.0005
22	Tidak Dikenal	Tidak Dikenal	Menghadap Keatas	0.0004
23	Dikenal	Dikenal	Menghadap Keatas	0.0003
24	Tidak Dikenal	Tidak Dikenal	Menghadap Keatas	0.0004
25	Dikenal	Tidak Dideteksi	Menunduk	0.0003
26	Tidak Dikenal	Tidak Dikenal	Menunduk	0.0004
27	Tidak Dikenal	Tidak Dikenal	Menunduk	0.0005
28	Dikenal	Dikenal	Menunduk	0.0004
29	Dikenal	Dikenal	Menunduk	0.0004
30	Tidak Dikenal	Tidak Dideteksi	Menunduk	0.0005

Pada tabel 4.20 dilakukan pengujian dengan berbagai posisi wajah yang berbeda dan untuk hasil akurasinya dapat dilihat pada tabel 4.21 dan untuk hasil perhitungan F-Score dapat dilihat pada tabel 4.22.

Tabel 4.21. Hasil Perhitungan Akurasi Dari Berbagai Posisi Wajah

No.	Posisi Wajah	Output Sistem yang Sesuai dengan Output Aktual	Akurasi	Time Detection
1	Menghadap Kamera	6	100 %	0.0003
2	Menghadap Kekan	5	83.3 %	0.0004
3	Menghadap Kekiri	5	83.3 %	0.0003
4	Menghadap Keatas	5	83.3 %	0.0004
5	Menunduk	4	66.6%	0.0004

Tabel 4.22. Hasil Perhitungan F-Score

	Kondisi Percobaan				
	A	B	C	D	E
TP	3	2	3	3	2
FN	0	1	0	0	1
FP	0	0	0	0	0
TN	3	3	3	3	3
<i>Precision</i>	100	100	100	100	100
<i>Recall</i>	100	66,6	100	100	66,6
F-score	100	79,9	100	100	79,9

Berdasarkan hasil pada tabel 4.21 maka dapat dihitung akurasi untuk pengenalan wajah berdasarkan posisi wajah yang berbeda sebagai berikut :

$$\text{Akurasi} = \frac{6+5+5+5+4}{30} \times 100 \% = 83.3\%$$

Untuk F-Score rata-rata dari tabel 4.22 maka didapat adalah 91.96% dengan rata-rata waktu deteksi 0.0004 detik.

4.2.3. *Effisiensi Sistem Terhadap Kamera Konvensional*

Salah satu tujuan penelitian ini adalah mengefisiensikan kamera pengaman dari sisi media penyimpanan. Untuk menghitung efisiensi kamera pengaman dapat

menggunakan persamaan 3.3. Dengan menggunakan kamera cctv konvensional dalam tempo waktu pengujian 9 jam didapatkan ukuran video *5.85 GB*. Maka penghematan memory penyimpanan dengan sistem sebagai berikut :

$$\text{Penghematan Memory} = \frac{5.85 \text{ GB} - 1.56 \text{ GB}}{5.85 \text{ GB}} \times 100 \% = 73.3\%$$

Dimana hasil perekaman dengan sistem yang dibuat adalah *1.56 GB* sehingga penghematan media penyimpanan kamera pengaman dengan menggunakan sistem ini adalah 73.3 %.

BAB 5

KESIMPULAN DAN SARAN

Bab ini berisi tentang kesimpulan dari keseluruhan rancangan dan hasil penelitian sistem pendeteksian manusia dan pengenalan wajah untuk peningkatan kinerja kamera pengaman, serta saran-saran yang diperlukan untuk pengembangan untuk penelitian berikutnya.

5.1. Kesimpulan

Berdasarkan pengujian sistem pendeteksian manusia dan pengenalan wajah untuk peningkatan kinerja kamera pengaman didapatkan kesimpulan sebagai berikut:

1. Dengan sistem ini penghematan media penyimpanan untuk kamera pengaman adalah sebesar 73,3 %.
2. Pada proses pendeteksian manusia sebagai bagian dari peningkatan kinerja kamera pengaman sangat dipengaruhi oleh intensitas cahaya, dimana pada intensitas cahaya 5 lux proses pendeteksian berjalan dengan akurasi terendah pada penelitian ini.
3. Pada proses pengenalan wajah dengan metode Deep Convolutional Neural Network proses pengenalan wajah memiliki akurasi 83.3% dan F-Score 91.96%. Pada proses pengenalan wajah posisi kemiringan wajah dan intensitas cahaya sangat berpengaruh terhadap akurasi metode.

5.2. Saran

Adapun saran yang dapat diberikan oleh penulis terkait dengan penelitian ini untuk pengembangan selanjutnya adalah sebagai berikut :

1. Merubah lingkungan penelitian dengan intensitas cahaya yang kurang untuk lebih meningkatkan kinerja dari pada kamera pengaman.
2. Meningkatkan pengenalan wajah dengan dapat membedakan antara gambar dan manusia asli.
3. Merubah bentuk notifikasi ke pemilik rumah dengan notifikasi yang lebih modern.

DAFTAR PUSTAKA

- Burges, C. J. 1998. A Tutorial on Support Vector Machines for Pattern Recognition. Kluwer Academic Publishers.
- Belghini, N., Zarghili, A. & Kharroubi, J. 2012. 3D Face Recognition using Gaussian Hermite Moments. *International Journal of Computer Applications*.
- Chen, M., Ma, G. & Kee, S. 2005. Multi-view Human Head Detection in Static Images. *Conference on Machine Vision Applications* 3(21):100-103.
- Geithey, A. 2016. Machine Learning is Fun! Part 4: Modern Face Recognition with Deep Learning. *Medium.com*, 24 Juli 2016 (diakses 17 Januari 2017).
- Goodfellow, I., Bengio, Y. & Courville, A. 2016. Deep Learning (Adaptive Computation and Machine Learning series). The MIT Press: Cambridge.
- Jaikumar, K. & Jaiganesh, B. 2014. An Economical Car Security Authentication System Based On Face Recognition Structure. *International Journal Of Technology Enhancements And Emerging Engineering Research* 2(8): 28-31.
- Khan, I., Abdullah, H. & Zainal, M, S, B. 2012. Efficient Eyes and Mouth Detection Algoritihm using Combination of Viola Jones and Skin Color Pixel Detection. *International Journal of Engineering and Applied Sciences* 3 (4):51-60.
- Khan, S., Khoduskar, A. & Koli, N, A. 2011. Home Automation System. *International Journal of Advanced Engineering Technology* :129-132.
- Khan, S, F. & Baset, S, A. 2002. Real Time Human Motion Detection And Classification. *IEEE* :135-139.
- Krizhevsky, A., Sutskever, I. & Hinton, G.E. 2012. ImageNet Classification with Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems (NIPS)* 12(25): pp. 1097-1105.
- Li, F., Johnson, J. & Yeung, S. 2015. CS231n Convolutional Neural Networks for Visual Recognition (*Online*) <http://cs231n.github.io/convolutional-networks/> (10 April 2017)

- Lienhart, R. & Maydt, J. 2002. An Extended Set of Haar-Like Features for Rapid. *IEEE ICIP* (1):900-903.
- Mahdi, O, A. & Alankar, B. 2014. Wireless Controlling Of Remote Electrical Device Using Android Smartphone. *IOSR Journal of Computer Engineering*. 23-27.
- Mitra, S. 2012. Gaussian Mixture Models for Human Face Recognition under Illumination Variations. *Scientific Research*.
- Satria, A., Priadi, M. L., Wulandhari. L. A. & Budhiarto, W. 2015. The Framework of Home Remote Automation System Based on Smartphone. *International Journal of Smart Home*:9(1) :53-60.
- Schroff, F., Kalenichenko, D. & Philbin, J. 2015. FaceNet: A Unified Embedding for Face Recognition and Clustering. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015.
- Sianturi, J. 2014. Sistem pendeteksian manusia untuk keamanan ruangan menggunakan *viola jones*. Skripsi. Universitas Sumatera Utara.
- Sigla, N. 2014. Motion Detection Based on Frame Difference Method. *International Journal of Information & Computation Technology*.
- Sreedevi, M., Yaswanth, K. A., Anja. B. G. & Shendil. K. R. 2012. Real Time Human Detection For Human Recognition. *Proceedings of the World Congress on Engineering and Computer Science*.
- Stanford University. 2013. Feature extraction using convolution. (Online) http://deeplearning.stanford.edu/wiki/index.php/Feature_extraction_using_convolution (11 Maret 2017).
- Triatmoko, B.D., Pramono, S.H. & Dahlan, H. S. 2014. Penggunaan Metode *Viola-Jones* dan Algoritma Eigen Eyes dalam Sistem Kehadiran Pegawai. *Jurnal EECCIS*.
- Viola, P & Jones, M. 2001. Rapid Object Detection using a Boosted Cascade of Simple Features. *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition* 1 : 511-518.
- Viola, P. & Jones, M. 2004. Robust Real-Time Face Detection. *International Journal of Computer Vision* 57(2):137–154.
- Y.-W. Bai & Y.-T. Ku. 2008. Automatic Room Light Intensity Detection and Control Using a Microprocessor and Light Sensors. *IEEE Transactions on Consumer Electronics*.