

# IMPLEMENTASI METODE *TRIANGLE GEOMETRY* UNTUK PENGENALAN ARAH PERGERAKAN KEPALA

SKRIPSI

Untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun oleh:  
Irnayanti Dwi Kusuma  
NIM: 135150207111065



PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA  
MALANG  
2018

**PENGESAHAN**

**IMPLEMENTASI METODE TRIANGLE GEOMETRY UNTUK PENGENALAN ARAH  
PERGERAKAN KEPALA**

**SKRIPSI**

Untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun Oleh:  
Irnayanti Dwi Kusuma  
NIM: 135150207111065

Skrripsi ini telah diuji dan dinyatakan lulus pada  
29 Juni 2018

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

  
Fitri Utamingrum, Dr. Eng., S.T., M.T  
NIP. 19820710 200812 2 001

  
Randy Cahya Wihandika, S.ST., M.Kom  
NIK. 201405 880206 1 001

Mengetahui  
Ketua Jurusan Teknik Informatika



  
Tri Adhoro Karniawan, S.T., M.T., Ph.D  
NIP. 19710518 200312 1 001



## IDENTITAS TIM PENGUJI

### Majelis penguji ujian skripsi

Penguji I:

Dr.Eng Ahmad Afif Supianto, S.Si, M.Kom (ke I) \* ketua majelis  
NIK. 2012018206231001

Penguji II:

Yuita Arum Sari, S.Kom., M.Kom (ke II)  
NIK: 2016098807152001



### PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 29 Juni 2018



Inayanti Dwi Kusuma  
NIM: 135150207111065



## DAFTAR RIWAYAT HIDUP

Nama : Irnayanti Dwi Kusuma  
 NIM : 135150207111065  
 Tempat dan Tanggal Lahir : Lamongan, 10 Februari 1995  
 Jenis Kelamin : Perempuan  
 Status Perkawinan : Belum Kawin  
 Agama : Islam  
 Alamat Rumah : Perumda Deket Gg7/5 Lamongan  
 No Telp : 085646431222  
 Alamat e-mail : Irnayantid@gmail.com

RIWAYAT PENDIDIKAN		
Tahun	Sekolah	Alamat
1999 – 2001	TK Permata Bunda Perumda-Deket Permai Lamongan	Jl. Sawo – Deket Permai kec. Deket, kab. Lamongan
2001 – 2007	SDN Deket Kulon III Lamongan	Jl. Sawo – Deket Permai kec. Deket, kab. Lamongan
2007 – 2010	SMP Negeri 2 Lamongan	Jl. Veteran No. 3 Lamongan
2010 – 2013	SMA Negeri 2 Lamongan	Jl. Veteran No. 1 Lamongan
2013 – 2018	Teknik Informatika Universitas Brawijaya	Jl. Veteran No. 9 Malang

## KATA PENGANTAR

Puji syukur kehadiran Allah SWT, karena atas kehendak dan rahmat-Nya lah penulis dapat menyelesaikan skripsi yang berjudul “**Implementasi Metode *Triangle Geometry* Untuk Pengenalan Arah Pergerakan Kepala**”. Pada kesempatan kali ini, penulis ingin menyampaikan rasa hormat dan terima kasih yang sebesar-besarnya kepada semua pihak yang telah memberikan bantuan dan dukungan selama pengerjaan skripsi, diantaranya:

1. Kedua Orang tua penulis, Alm. Slamet dan Mas’anah yang tidak pernah berhenti mendoakan, mendukung penulis untuk selalu semangat dalam proses perkuliahan hingga tahap menyelesaikan skripsi ini.
2. Kakak kandung penulis, M. Awalludin Khabibi, Elly Nur Fajar, dan Muckhamad Khasanuddin Masykur yang selalu mendukung dan memberi saran penulis dalam proses perkuliahan dan menyelesaikan skripsi ini.
3. Bapak Wayan Firdaus Mahmudy, S.Si, M.T, Ph.D, selaku Dekan Fakultas Ilmu Komputer.
4. Bapak Tri Astoto Kurniawan, S.T, M.T, Ph.D, selaku Ketua Jurusan Teknik Informatika.
5. Agus Wahyu Widodo, S.T, M.Cs, selaku Ketua Program Studi Teknik Informatika
6. Ibu Fitri Utamingrum, Dr. Eng., S. T, M. T, sebagai Dosen Pembimbing pertama yang senantiasa meluangkan waktu untuk membimbing, mengarahkan dengan sabar, memberikan kritik dan saran selama penelitian skripsi.
7. Bapak Randy Cahya Wihandika, S.ST., M. Kom sebagai Dosen Pembimbing kedua, atas segala bimbingan, waktu, kritik dan saran yang diberikan kepada penulis selama penelitian skripsi.
8. Bapak Herman Tolle, Dr. Eng., S.T, M.T, selaku Ketua Grup Riset MGM yang senantiasa memberikan dukungan kepada penulis.
9. Dosen Fakultas Ilmu Komputer Universitas Brawijaya atas ilmu, nasehat dan motivasi yang diberikan kepada penulis.
10. Hafizhuddin Zul Fahmi atas motivasi dan dukungannya selama perkuliahan hingga proses pengerjaan skripsi.
11. Teman – teman Robotiik FILKOM sebagai tempat menimbah ilmu dan pengalaman selama perkuliahan hingga menyelesaikan skripsi ini, Yusuf yang bersedia menjadi *volunteer* untuk pengujian skripsi ini.
12. Teman-teman ‘Astagfirullah’: Pravita, Arinda, Farikh, Nasrul, Udin, Dhana, Ariz yang selalu memberikan motivasi, hiburan dan dukungan kepada penulis.
13. Teman-teman MGM Studio FILKOM UB: Imam Ghozali, S. Kom, Ivan Putra Pratama, S. Kom., Doni Putra Purbawa, Harits Abdurohman, Andi Hafidh, dan teman teman yang lain yang tidak bisa saya sebutkan satu persatu atas dukungan dan bantuan yang diberikan.
14. Teman - teman PPI Saga Jepang: Devi Oktabrianti, Erona Wavareta, Widya, Yulius Firantoko, Gabriel yang bersedia menjadi *volunteer* untuk

pengujian dalam skripsi ini dan anggota PPI Saga lainnya atas dukungan dan motivasinya.

15. Peerakam C., dan Koharu, teman selama menjalani student exchange di Saga University, Japan yang bersedia menjadi *volunteer* untuk pengujian data skripsi.
16. Pravita, Erza, Ayuni, Chalid, Ramandha, Hafis, Cahyono, Ardi, Rere, Dimas, yang telah bersedia menjadi *volunteer* untuk pengujian data skripsi.
17. Semua pihak yang telah membantu kelancaran penelitian skripsi dan memberikan doa yang tidak dapat disebutkan penulis satu persatu.

Penulis menyadari bahwa dengan kekurangan dan keterbatasan yang dimiliki penulis, skripsi ini masih jauh dari kata sempurna dan masih banyak kekurangan, oleh karena itu penulis mengharapkan saran yang membangun agar skripsi ini bermanfaat bagi yang membutuhkan dan dapat digunakan untuk penelitian yang selanjutnya.

Malang, 29 Juni 2018

Penulis  
Irnayantid@gmail.com



## ABSTRAK

Saat ini semakin populer aplikasi deteksi kepala atau wajah yang digunakan sebagai pengamanan perangkat, *virtual and augmented reality*, *game* komputer dan kontrol perangkat. Namun, penyandang disabilitas fisik memiliki kendala untuk mengontrol sistem, disisi lain juga terdapat kendala pada pengguna normal yang menggunakan *headpiece* untuk mengontrol sistem yaitu pengalokasian biaya untuk pembelian *headpiece*. Pengenalan arah pergerakan kepala menggunakan metode *triangle geometry* dapat diimplementasikan ke dalam sistem yang memanfaatkan *input* sebagai kontrol. Pemanfaatan *input* sebagai kontrol dapat mempermudah penyandang difabel untuk menggunakan suatu sistem dengan mudah dan interaktif. Selain itu, topik ini dapat membantu mengurangi biaya kebutuhan sistem. Penelitian ini memiliki tahap deteksi warna kulit, mata dan hidung untuk dapat melakukan deteksi wajah dan arah pergerakan kepala. Penelitian ini berfokus pada keberhasilan sistem yang dapat mengenali arah pergerakan kepala sesuai dengan keadaan kepala secara *real-time*. Arah kepala yang digunakan yakni atas, bawah, kanan dan kiri. Sistem melakukan pengolahan nilai *yaw*, *pitch*, *roll* dengan metode *triangle geometry* sehingga didapatkan *range* nilai untuk masing – masing arah pergerakan kepala. Nilai akurasi masing – masing arah pergerakan kepala menunjukkan angka 88% untuk arah atas, 82,6% untuk arah bawah, 84,6% untuk arah kanan dan 73,5% untuk arah kiri sehingga sistem ini dapat diimplementasikan sebagai *input* kontrol.

**Kata Kunci:** *Triangle Geometry, Face Detection, Head Movement.*

## ABSTRACT

Nowadays, applications that detect head or face are popularly used as device securing, virtual and augmented reality, computer games, and device control. However, people with physical disabilities have constraints to control the system, on the other hand there are also constraints on normal users who use headpieces to control the system that is the cost allocation for headpiece purchases. Identification of head direction movement using Triangle Geometry Method could be implemented into a system that utilizes input as a control. Utilization of inputs as controls can make easier for people with disabilities to use a system and use it interactively. In addition, this topic can help reduce the cost of system requirements and does not complicate the user, such as using a special device that put on the user's head to control a system. Skin color detection, eyes, and nose are phases that used on this topic to build a system of face detection and head direction movement. This research focuses on the success of systems that can recognize the direction of head movement with the state of the head in real-time. The direction of the head is up, down, right and left. The system processes the value of yaw, pitch, roll with triangle geometry method so it can get a range values for each direction of head movement. The accuracy value of each direction is 88% for the upward direction, 82.6% for the downward direction, 84.6% for the right direction and 73.5% for the left direction. So, this system can be implemented as control input.

**Keyword:** Triangle Geometry, Face Detection, Head Movement.

## DAFTAR ISI

PENGESAHAN .....	Error! Bookmark not defined.ii
PERNYATAAN ORISINALITAS .....	iii
KATA PENGANTAR.....	vi
ABSTRAK.....	viii
ABSTRACT .....	ix
DAFTAR ISI.....	x
DAFTAR TABEL.....	xii
DAFTAR GAMBAR.....	xv
DAFTAR KODE PROGRAM .....	xvii
<b>BAB 1 PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar belakang.....	1
1.2 Rumusan masalah.....	2
1.3 Tujuan .....	Error! Bookmark not defined.2
1.4 Manfaat.....	Error! Bookmark not defined.3
1.5 Batasan masalah .....	Error! Bookmark not defined.
1.6 Sistematika Pembahasan.....	Error! Bookmark not defined.
<b>BAB 2 LANDASAN KEPUSTAKAAN .....</b>	<b>Error! Bookmark not defined.</b>
2.1 Kajian Pustaka .....	Error! Bookmark not defined.
2.2 <i>Light</i> Meter HS 1010 .....	Error! Bookmark not defined.
2.3 Kamera Webcam Logitech C920 .....	Error! Bookmark not defined.
2.4 Citra.....	7
2.4.1 Citra Biner.....	7
2.4.2 Pengolahan Citra .....	7
2.5 <i>Library</i> OpenCv .....	10
2.6 <i>Haar Cascade</i> .....	10
2.7 <i>Skin Detection</i> .....	15
2.8 <i>Triangle Geometry</i> .....	Error! Bookmark not defined.
<b>BAB 3 METODOLOGI PENELITIAN .....</b>	<b>18</b>
3.1 Analisis Masalah.....	19
3.2 Studi Literatur .....	19
3.3 Rekayasa Kebutuhan Sistem.....	19



3.3.1	Kebutuhan <i>Perangkat Keras</i> .....	19
3.3.2	kebutuhan <i>Perangkat lunak</i> .....	19
3.4	Perancangan Sistem.....	20
3.5	Implementasi Sistem .....	20
3.6	Pengujian Sistem.....	21
3.7	Analisis dan Pembahasan .....	21
3.8	Kesimpulan.....	21
BAB 4	PERANCANGAN.....	22
4.1	Perancangan Proses.....	22
4.1.1	<i>Pre-Processing</i> .....	23
4.1.2	Deteksi Warna Kulit ( <i>Skin Detection</i> ) .....	23
4.1.3	Deteksi Bagian Wajah .....	26
4.1.4	<i>Head Tracking</i> .....	28
4.2	Perhitungan Manual .....	31
4.2.1	Nilai <i>yaw, pitch, roll</i> .....	31
4.3	Perancangan Pengujian .....	35
BAB 5	IMPLEMENTASI .....	37
5.1	Implementasi Proses.....	37
5.1.1	Implementasi <i>Pre-processing</i> .....	37
5.1.2	Implementasi Deteksi Warna Kulit .....	37
5.1.3	Implementasi Deteksi Bagian Wajah <b>Error! Bookmark not defined.</b>	
5.1.4	Implementasi <i>HeadTracking</i> .....	41
5.1.5	Implementasi Klasifikasi Nilai Batas Atas dan Batas Bawah <i>Range Arah</i> .....	43
BAB 6	PENGUJIAN .....	48
6.1	Pengujian Deteksi Warna Kulit, Mata dan Hidung .....	48
6.2	Pengujian Analisis Nilai Batas Atas dan Batas Bawah.....	68
6.3	Pengujian Arah Pergerakan Kepala.....	83
BAB 7	PENUTUP .....	96
7.1	Simpulan .....	96
7.2	Saran .....	96
DAFTAR PUSTAKA	.....	98



## DAFTAR TABEL

Tabel 4.1 Tabel Pengujian Deteksi Warna Kulit, Mata dan Hidung	<b>Error! Bookmark not defined.</b>
Tabel 4.2 Tabel Pengujian Analisis Nilai Batas Atas dan Batas Bawah .....	35
Tabel 4.3 Pengujian Arah Pergerakan Kepala .....	36
Tabel 5.1 Klasifikasi Arah Atas.....	43
Tabel 5.2 Klasifikasi Arah Bawah.....	44
Tabel 5.3 Klasifikasi Arah Kanan.....	44
Tabel 5.4 Klasifikasi Arah Kiri .....	45
Tabel 6.1 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-1.....	48
Tabel 6.2 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-2.....	50
Tabel 6.3 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-2 (Lanjutan) .....	51
Tabel 6.4 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-3.....	52
Tabel 6.5 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-4.....	54
Tabel 6.6 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-4 (Lanjutan) .....	55
Tabel 6.7 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-4 (Lanjutan) .....	56
Tabel 6.8 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-5.....	56
Tabel 6.9 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-5 (Lanjutan) .....	57
Tabel 6.10 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-5 (Lanjutan) .....	58
Tabel 6.11 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-6.....	58
Tabel 6.12 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-6 (Lanjutan) .....	59
Tabel 6.13 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-6 (Lanjutan) .....	60
Tabel 6.14 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-7.....	61
Tabel 6.15 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-7 (Lanjutan) .....	62
Tabel 6.16 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-7 (Lanjutan) .....	62



Tabel 6.17 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-8.....	62
Tabel 6.18 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-8 (Lanjutan) .....	63
Tabel 6.19 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-9.....	64
Tabel 6.20 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-9 (Lanjutan) .....	65
Tabel 6.21 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-10.....	67
Tabel 6.22 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-10 (Lanjutan) .....	68
Tabel 6.23 Pengujian Analisis Nilai Batas Atas dan Batas Bawah Data Uji Ke-1 ...	69
Tabel 6.24 Pengujian Analisis Nilai Batas Atas dan Batas Bawah Data Uji Ke-2 ...	70
Tabel 6.25 Pengujian Analisis Nilai Batas Atas dan Batas Bawah Data Uji Ke-3 ...	72
Tabel 6.26 Pengujian Analisis Nilai Batas Atas dan Batas Bawah Data Uji Ke-4 ...	73
Tabel 6.27 Pengujian Analisis Nilai Batas Atas dan Batas Bawah Data Uji Ke-5 ...	74
Tabel 6.28 Pengujian Analisis Nilai Batas Atas dan Batas Bawah Data Uji Ke-6 ...	76
Tabel 6.29 Pengujian Analisis Nilai Batas Atas dan Batas Bawah Data Uji Ke-7 ...	66
Tabel 6.30 Pengujian Analisis Nilai Batas Atas dan Batas Bawah Data Uji Ke-8 ...	79
Tabel 6.31 Pengujian Analisis Nilai Batas Atas dan Batas Bawah Data Uji Ke-9 ...	80
Tabel 6.32 Pengujian Analisis Nilai Batas Atas dan Batas Bawah Data Uji Ke-10 .	82
Tabel 6.33 Pengujian Data Uji Arah Atas .....	84
Tabel 6.34 Pengujian Data Uji Arah Atas (Lanjutan) .....	85
Tabel 6.35 Pengujian Data Uji Arah Atas (Lanjutan) .....	86
Tabel 6.36 Pengujian Data Uji Arah Atas (Lanjutan) .....	87
Tabel 6.37 Pengujian Data Uji Arah Bawah.....	87
Tabel 6.38 Pengujian Data Uji Arah Bawah (Lanjutan) .....	88
Tabel 6.39 Pengujian Data Uji Arah Bawah (Lanjutan) .....	89
Tabel 6.40 Pengujian Data Uji Arah Bawah (Lanjutan) <b>Error! Bookmark not defined.</b>	
Tabel 6.41 Pengujian Data Uji Arah Kanan .....	91
Tabel 6.42 Pengujian Data Uji Arah Kanan (Lanjutan) .....	91
Tabel 6.43 Pengujian Data Uji Arah Kanan (Lanjutan) <b>Error! Bookmark not defined.</b>	
Tabel 6.44 Pengujian Data Uji Arah Kiri .....	93
Tabel 6.45 Pengujian Data Uji Arah Kiri (Lanjutan).....	94



Tabel 6.46 Pengujian Data Uji Arah Kiri (Lanjutan)..... 94  
Tabel 6.47 Akurasi Masing-Masing Arah Pergerakan KepalaError! Bookmark not defined.



## DAFTAR GAMBAR

Gambar 2.1 <i>Light Meter</i> .....	6
Gambar 2.2 Kamera Webcam Logitech C920.....	7
Gambar 2.3 <i>Equalize Histogram</i> .....	9
Gambar 2.4 <i>Operasi Pencerminan (Flipping)</i> .....	9
Gambar 2.5 Penghalusan ( <i>Blur</i> ) .....	10
Gambar 2.6 Fitur <i>Haar Cascade</i> .....	10
Gambar 2.7 Metode <i>AdaBoost</i> .....	14
Gambar 2.8 Implementasi Fitur <i>Haar</i> pada citra <i>input</i> <b>Error! Bookmark not defined.</b>	<b>Error! Bookmark not defined.</b>
Gambar 2.9 Estimasi Posisi Kepala.....	16
Gambar 2.10 Ilustrasi <i>Triangle geometry</i> pada kepala <b>Error! Bookmark not defined.</b>	<b>Error! Bookmark not defined.</b>
Gambar 2.11 Ilustrasi <i>Triangle Geometry</i> pada posisi <i>roll</i> <b>Error! Bookmark not defined.</b>	<b>Error! Bookmark not defined.</b>
Gambar 3.1 Alur Metode Penelitian.....	18
Gambar 3.2 Blok Diagram Sistem .....	20
Gambar 4.1 Flowchart Utama Sistem .....	22
Gambar 4.2 FlowChart <i>Pre-Processing</i> .....	23
Gambar 4.3 Flowchart Deteksi Warna Kulit.....	24
Gambar 4.4 Flowchart Aturan Warna RGB .....	25
Gambar 4.5 Flowchart Aturan Warna YCbCr .....	26
Gambar 4.6 Flowchart Aturan Warna HSV .....	26
Gambar 4.7 Flowchart Deteksi Wajah .....	28
Gambar 4.8 Flowchart <i>Head Tracking</i> .....	29
Gambar 4.9 Flowchart Klasifikasi <i>Range</i> Arah .....	30
Gambar 4.10 Orientasi Gerakan Kepala.....	31
Gambar 4.11 Citra Arah Atas .....	31
Gambar 4.12 Citra Arah Bawah.....	32
Gambar 4.13 Citra Arah Kanan .....	33
Gambar 4.14 Citra Arah Kiri .....	34
Gambar 5.1 Ilustrasi Klasifikasi <i>range</i> arah berdasarkan nilai <i>Yaw</i> <b>Error! Bookmark not defined.</b>	<b>Error! Bookmark not defined.</b>

Gambar 5.2 Ilustrasi Klasifikasi *range* arah berdasarkan nilai *Pitch* ..... 46  
Gambar 5.3 Ilustrasi Klasifikasi *range* arah berdasarkan nilai *Roll* ..... 47



## DAFTAR KODE PROGRAM

Kode Program 5.1 Implementasi <i>Pre-Processing</i> .....	37
Kode Program 5.2 Implementasi Deteksi Warna Kulit .....	38
Kode Program 5.3 Implementasi Penerapan aturan warna RGB, HSV dan YCbCr	39
Kode Program 5.4 Implementasi Deteksi Bagian Wajah <b>Error! Bookmark not defined.</b>	
Kode Program 5.5 Implementasi Perhitungan Nilai Variable Persamaan Metode .....	41
Kode Program 5.6 Implementasi Klasifikasi <i>Range</i> Arah Pergerakan Kepala.....	42





## BAB 1 PENDAHULUAN

### 1.1 Latar belakang

Saat ini semakin populer aplikasi deteksi kepala atau wajah yang digunakan sebagai pengamanan perangkat, *virtual and augmented reality*, *game* komputer dan kontrol perangkat. Namun, penyandang disabilitas fisik memiliki kendala untuk mengontrol sistem, disisi lain juga terdapat kendala pada pengguna normal yang menggunakan *headpiece* untuk mengontrol sistem yaitu pengalokasian biaya untuk pembelian *headpiece*. Hasil Riskesdas tahun 2013 mendapatkan bahwa prevalensi disabilitas tertinggi adalah pada kelompok orang yang tidak bekerja, yaitu sebesar 14,4% dengan usia produktif. Hal tersebut dapat menghambat produktifitas mereka akan dunia yang semakin bergantung pada teknologi dan rentan akan kondisi kemiskinan. Pemanfaatan kontrol perangkat dapat membantu penyandang difabel untuk mempermudah penggunaan pada suatu sistem dan dapat menggunakannya dengan interaktif. Dengan adanya fitur deteksi kepala atau wajah yang dimasukkan menjadi bagian dari interaksi manusia dengan komputer bertujuan agar pengguna dapat merasa lebih interaktif menggunakan aplikasi. Menyediakan interaksi manusia dengan komputer yang baik adalah poin penting dalam pengembangan perangkat lunak interaktif (Barr, 2006).

Interaksi klasik menggunakan *mouse*, *keyboard* atau *gamepad*, sangat terbatas dibandingkan dengan realita grafis yang akan ditampilkan aplikasi. Seperti pada beberapa *game* komputer melakukan pendekatan menggunakan perangkat *headpiece* untuk mendeteksi pergerakan kepala dan mengubah arah permainan. Beberapa motivasi telah memicu para peneliti untuk memberikan solusi visi komputer ke masalah *real-time* deteksi gerakan kepala yakni membuat interaksi komputer lebih murah dan perangkat keras komputer lebih cepat yang dapat menyajikan *real-time image processing*, meningkatkan keterjangkauan perangkat video seperti kamera dan video. Visi komputer merupakan solusi yang menarik karena memecahkan masalah deteksi gerakan kepala secara *real-time* tanpa menggunakan biaya yang mahal dan menggunakan alat pada kepala secara fisik. Selain itu, visi komputer menawarkan fleksibilitas karena dapat digunakan dalam lingkup yang lebih luas.

Beberapa penelitian yang terkait dengan visi komputer telah banyak dilakukan sebelumnya untuk berbagai macam tujuan, antara lain: untuk mengurangi tingkat kecelakaan lalu lintas akibat kelelahan dan mengantuk penelitian yang dilakukan oleh Churiwala (2012) menggunakan visi komputer untuk mengenali kondisi pengemudi. Dengan hanya menggunakan *Haar like-features* untuk melakukan deteksi bagian wajah (mata) sebagai parameter pengukur pada penelitiannya. Penelitian selanjutnya yang dilakukan oleh Yu & Gang (2011) adalah memprediksi pergerakan kepala dengan menggunakan metode AAM (*Active Appearance Models*) untuk mengenali bagian depan wajah dan memanfaatkan metode Kalman Filter untuk mendapatkan hasil estimasi arah gerakan kepala. Sedangkan penelitian yang dilakukan oleh Perreira Da Silvia (2009) menerapkan pergerakan



kepala sebagai masukan *game* pada komputer. Beberapa penelitian ini menunjukkan bahwa visi komputer dapat dikembangkan untuk lingkup yang luas dan dapat membantu pengguna untuk lebih interaktif dengan interaksi komputer menggunakan visi komputer dibandingkan dengan menggunakan interaksi klasik.

Berbagai metode yang telah diterapkan pada penelitian tersebut memiliki kelebihan dan kekurangan masing-masing, seperti metode *Haar like-features* yang diimplementasikan oleh Churiwala (2012) yang mudah untuk diimplementasikan. Implementasi metode *AAM (Active Appearance Model)* yang dilakukan oleh Yu & Gang (2011) untuk menjawab permasalahan pada metode *AdaBoost* yang mengenali wajah bagian depan dan tidak menunjukkan hasil yang bagus ketika sudut wajah sedikit diputar. Sedangkan metode *Triangle Geometry* yang diimplementasikan oleh Perreira Da Silvia (2009) dapat membuat sistem pelacakan kepala secara *real-time* dengan memanfaatkan sumber daya yang rendah. Sistem ini menyediakan solusi cepat dan murah untuk mengintegrasikan suatu pola pengguna ke aplikasi interaktif.

Oleh karena itu, penulis mengembangkan interaksi menggunakan visi komputer tanpa menggunakan biaya yang mahal dan dapat di aplikasikan ke lingkup yang luas dengan cara mengenali arah pergerakan kepala user. Metode yang digunakan adalah *Triangle Geometry* untuk mengenali pergerakan kepala dan penggunaan *Haar Cascade* untuk deteksi bagian pada wajah. Hasil yang diinginkan penulis dalam penelitian ini berupa arah pergerakan kepala dalam bentuk kalimat yang sesuai dengan keadaan kepala secara *real-time* dengan menunjukkan nilai *yaw, pitch, roll*. Diharapkan dengan aplikasi implementasi metode ini dapat di aplikasikan ke lingkup yang lebih luas dan sebagai pengganti interaktif klasik dan membuat kemudahan pada pengguna.

## 1.2 Rumusan masalah

Berdasarkan latar belakang yang di atas, maka rumusan masalah yang dapat dibuat adalah sebagai berikut:

1. Bagaimana menerapkan metode *triangle geometry* untuk deteksi arah pergerakan kepala?
2. Bagaimana analisis nilai batas atas dan batas bawah *yaw, pitch, roll* pada arah pergerakan kepala (kanan, kiri, atas, bawah)?
3. Bagaimana tingkat akurasi masing-masing arah pergerakan kepala dengan menggunakan Implementasi metode *triangle geometry*?

## 1.3 Tujuan

Dari rumusan masalah di atas, maka tujuan yang ingin dicapai dalam penelitian ini adalah sebagai berikut:

1. Menerapkan metode *triangle geometry* untuk deteksi arah pergerakan kepala.
2. Menganalisis nilai batas atas dan batas bawah *yaw, pitch, roll* pada arah pergerakan kepala (kanan, kiri, atas, bawah).

3. Menguji tingkat akurasi masing – masing arah pergerakan kepala dengan menggunakan Implementasi metode *triangle geometry*.

#### 1.4 Manfaat

Manfaat penelitian ini adalah mampu mengimplementasikan *metode triangle geometry* untuk deteksi arah pergerakan kepala dengan tujuan dapat dimanfaatkan sebagai kontrol dengan biaya yang murah menggunakan visi komputer sehingga membantu pengguna untuk merasa lebih interaktif dibandingkan dengan menggunakan interaksi klasik seperti *mouse*, *keyboard* atau *gamepad* serta dapat dimanfaatkan pada lingkup yang lebih luas, sebagai contoh mendeteksi kondisi mengantuk sopir truk.

#### 1.5 Batasan masalah

Pada penelitian ini, permasalahan dibatasi oleh hal – hal sebagai berikut:

1. Hanya satu wajah pengguna yang dapat dikenali dalam *frame*.
2. Hasil implementasi metode berupa arah pergerakan kepala (kanan, kiri, atas, bawah) dalam bentuk kalimat yang sesuai dengan keadaan kepala secara nyata dengan menunjukkan nilai *yaw*, *pitch*, *roll*.
3. Kamera yang digunakan adalah kamera Logitech C920 dengan resolusi HD 1080p.

#### 1.6 Sistematika Pembahasan

Uraian singkat mengenai struktur penulisan pada masing-masing bab adalah sebagai berikut:

##### **BAB I PENDAHULUAN**

Pada bab ini memuat latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan penelitian, sistematika pembahasan mengenai “Implementasi Metode Triangle Geometry Untuk Pengenalan Arah Pergerakan Kepala”.

##### **BAB II LANDASAN KEPUSTAKAAN**

Bab ini menjelaskan tentang landasan teori yang terkait dengan penelitian. Serta menjelaskan tentang penelitian serupa yang pernah dilakukan sebagai referensi penelitian.

##### **BAB III METODOLOGI PENELITIAN**

Bab ini membahas mengenai langkah-langkah penerapan metode *triangle geometry* untuk mengenali pergerakan kepala, *RGB-H-CbCr skin color* dan *Haar Cascade* bagian dari *Library OpenCv* untuk mendeteksi bagian pada wajah (Mata dan hidung). Pada bab ini dijelaskan pula mengenai langkah pengujian sistem yang di rancang.

## **BAB IV PERANCANGAN**

Bab ini menjelaskan secara rinci detail perancangan dari “Implementasi Metode Triangle Geometry Untuk Pengenalan Arah Pergerakan Kepala” yang meliputi flowchart sistem dan penentuan batas atas dan batas bawah *range* arah.

## **BAB V IMPLEMENTASI**

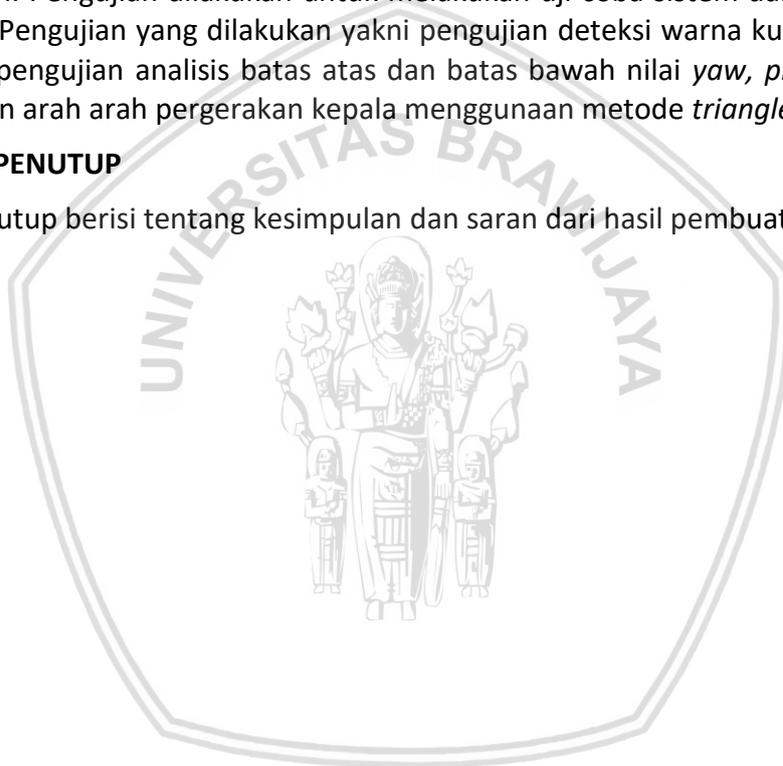
Bab ini memuat detail implementasi yang digunakan untuk menyelesaikan permasalahan dalam penelitian, dalam bab ini ditunjukkan potongan *source code* dan hasil proses yang telah dilakukan.

## **BAB VI PENGUJIAN**

Pengujian dilakukan apabila perancangan dan implementasi sistem telah dilakukan. Pengujian dilakukan untuk melakukan uji coba sistem dan melakukan analisis. Pengujian yang dilakukan yakni pengujian deteksi warna kulit, mata dan hidung, pengujian analisis batas atas dan batas bawah nilai *yaw*, *pitch*, *roll* dan pengujian arah arah pergerakan kepala menggunakan metode *triangle geometry*.

## **BAB VII PENUTUP**

Penutup berisi tentang kesimpulan dan saran dari hasil pembuatan sistem.



## BAB 2 LANDASAN KEPUSTAKAAN

Bab ini terdiri dari kajian pustaka dan dasar teori. Kajian pustaka membahas tentang penelitian sebelumnya yang telah dilakukan oleh Churiwala (2012), Yu & Gang (2011) dan Perreira Da Silvia (2009). Sedangkan dasar teori membahas mengenai teori-teori dasar yang digunakan untuk menunjang penulisan penelitian ini yang berjudul “Implementasi Metode Triangle Geometry Untuk Pengenalan Arah Pergerakan Kepala”.

### 2.1 Kajian Pustaka

Dalam penelitian ini kajian pustaka yang akan dibahas yaitu mengenai penelitian sebelumnya dengan metode yang berbeda. Referensi yang ditulis oleh Churiwala dkk pada tahun 2012, menjelaskan bahwa dengan sederhana, mengantuk dapat dikenali dengan parameter perpindahan mata, menguap, perputaran kepala dengan menggunakan *Haar like-features*. *Haar like-features* digunakan untuk mengenali bagian dari wajah yang kemudian akan dideteksi sesuai dengan kondisi *input*. *Haar like-features* menggunakan integral untuk menentukan ada atau tidaknya fitur pada suatu citra dan dengan menggunakan perhitungan integral waktu komputasi yang dibutuhkan rendah sehingga dapat diimplementasikan pada sistem yang menggunakan video sebagai *input*. Berbeda dengan penelitian yang dilakukan oleh Yu & Wang tahun 2011, menjelaskan mengenai pergerakan kepala dengan menggunakan *AAM (Active Appearance Model)* dan Kalman filter. *AAM (Active Appearance Model)* digunakan untuk mendeteksi wajah bagian depan dan mengenali bagian pada wajah (Mata dan Bibir) sedangkan Kalman filter untuk memprediksi pergerakan kepala. *AAM (Active Appearance Model)* adalah Pendekatan berbasis model yang menganalisis variasi yang berbeda dari objek menggunakan beberapa data uji objek dalam satu set pelatihan dan akhirnya menghitung model berdasarkan variasi objek (Rizvandi, et al., 2008). Metode ini memiliki waktu komputasi yang tinggi sehingga metode ini kurang baik untuk diimplementasikan pada sistem yang digunakan untuk deteksi secara *real-time*. Sedangkan referensi yang ditulis oleh Perreira Da Silvia dkk di tahun 2009, menjelaskan mengenai *triangle geometry* sebagai deteksi pergerakan kepala. Pada penelitian ini juga di jelaskan bahwa *triangle geometry* dibantu dengan tahapan cara untuk mengenali bagian dari wajah seperti *pre-processing* yang terdiri dari *image scaling*, *histogram backprojection*, *light correction* yang kemudian hasilnya diproses dengan *hough transform* dan *gaussian (DoG) filter*. Kelemahan dari proses deteksi bagian wajah pada penelitian ini adalah tidak menggunakan *skin detection* yang tujuannya dapat mengenali berbagai macam warna kulit pengguna yang akan menggunakan mengaplikasikan dari sistem ini dan membantu untuk mendeteksi bagian wajah.

Mengacu pada beberapa referensi yang telah dijelaskan di atas, maka penulis menghindari kelemahan yang ada dan menggabungkan kelebihan dari beberapa referensi tersebut dalam penelitian yang berjudul “Implementasi Metode Triangle

Geometry Untuk Pengenalan Arah Pergerakan Kepala”. Sehingga, tercapai tujuan dari penelitian ini.

Pada penelitian ini penulis melakukan mengembangkan deteksi bagian pada wajah (Mata dan Hidung) menggunakan metode *RGB-H-CbCr skin colour* dan *Haar Cascade* bagian dari *Library OpenCv*, deteksi arah pergerakan kepala yang dibuat dengan metode *triangle geometry* dengan tujuan untuk menciptakan suatu kontrol yang mudah digunakan sebagai interaksi pada suatu sistem dan mengetahui tingkat akurasi untuk dapat dibandingkan dengan metode-metode sebelumnya.

## 2.2 Light Meter HS 1010

Light Meter HS 1010 adalah alat ukur yang digunakan untuk mengukur nilai intensitas cahaya di suatu tempat. Alat ini dapat mengukur *range* intensitas cahaya 1 lux ~ 200000 lux. Tingkat akurasi yang digunakan pada alat ini menggunakan standar *incandescent lamp* dengan kondisi temperatur warna 2856K sehingga  $\pm 4\% \pm 10$  digit (<10000 lux) dan  $\pm 5\% \pm 10$  digit ( $\geq 10000$  lux).



## 2.3 Kamera Webcam Logitech C920

Kamera Webcam Logitech C920 adalah kamera beresolusi HD 1080p yang dapat digunakan untuk mengambil gambar atau merekam video dengan bentuk kabel berupa USB. Sehingga pada sistem ini dapat dimanfaatkan sebagai alat untuk membantu mengambil *input* untuk diproses kedalam sistem.



**Gambar 2.2 Kamera Webcam Logitech C920**

Sumber: (Amazon.com)

## 2.4 Citra

Citra adalah suatu representasi (gambaran), kemiripan atau mitasi dari suatu objek. Citra sebagai keluaran suatu sistem perekaman data dapat bersifat optik berupa foto, bersifat analog, berupa sinyal-signal video seperti gambar pada monitor televisi atau bersifat digital yang dapat langsung disimpan pada suatu media penyimpanan (Sutoyo, et al., 2009).

### 2.4.1 Citra Biner

Citra Biner adalah citra yang hanya mempunyai dua warna yakni hitam dan putih. Dua warna citra biner disimbolkan dengan angka 0 dan 1. Angka 0 merepresentasikan warna hitam dan angka 1 merepresentasikan warna putih.

### 2.4.2 Pengolahan Citra

Pengolahan Citra adalah suatu metode yang berfungsi untuk memproses citra atau gambar dan memanipulasinya menjadi data gambar untuk mendapat informasi tertentu.

#### 2.4.2.1 Grayscale

*Grayscale* adalah warna citra yang terdiri atas gradasi warna hitam ke putih. Hitam terdapat pada bagian yang mempunyai intensitas terlemah dan warna putih terdapat pada intensitas terkuat. Persamaan untuk mengubah citra RGB ke *grayscale* dapat menggunakan Persamaan 2.1.

$$\text{Grayscale} = (0.299 \times R) + (0.587 \times G) + (0.144 \times B) \quad (2.1)$$

#### 2.4.2.2 RGB

Model warna *RGB* terdiri atas tiga buah kombinasi warna primer yaitu merah (*Red*), hijau (*Green*), dan biru (*Blue*). *RGB* Ketiga warna primer tersebut membentuk sistem koordinat cartesian tiga dimensi. Model warna ini berorientasi *hardware*, terutama untuk warna monitor dan warna pada kamera video.

#### 2.4.2.3 YCbCr

Model warna *YCbCr* terdiri atas tiga komponen warna yaitu *Y*, *Cb* dan *Cr*. *Y* merupakan luminansi dan *Cb*, *Cr* merupakan tempat yang digunakan untuk

menyimpan informasi warna yang berbeda. Persamaan konversi warna *RGB* ke *YCbCr* pada Persamaan 2.2.

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 65,481 & 128,553 & 24,966 \\ -37,797 & -74,203 & 112,000 \\ 112,000 & -93,786 & -18,214 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (2.2)$$

#### 2.4.2.4 HSV

*HSV* merupakan gabungan dari tiga komponen warna yaitu *Hue*, *Saturation* dan *Value* atau *Brightness*. *Hue* menyatakan warna sebenarnya, seperti merah, violet, kemerahan, dsb. *Saturation* atau terkadang disebut *chroma*, adalah kemurnian atau kekuatan warna dari suatu warna. *Value* adalah kecerahan dari warna. Nilainya berkisar antara 0-100 %. Apabila nilainya 0 maka warnanya akan menjadi hitam, semakin besar nilai maka semakin cerah dan muncul berbagai variasi baru dari warna-warna tersebut. Persamaan konversi *RGB* ke *HSV* pada Persamaan 2.3.

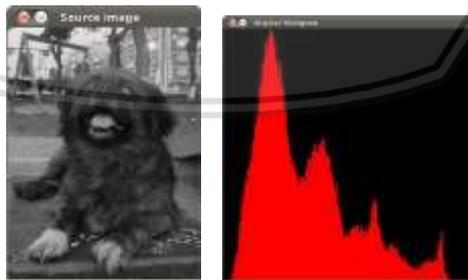
$$H = \tan\left(\frac{3(G-B)}{(R-G)+(R-B)}\right) \quad (2.3)$$

$$S = 1 - \frac{\min(R,G,B)}{V}$$

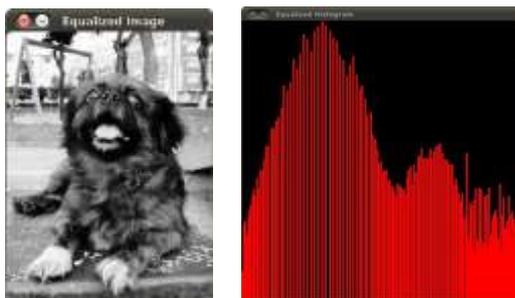
$$V = \frac{R+G+B}{3}$$

#### 2.4.2.5 Equalize Histogram

*Equalize Histogram* atau dapat disebut *equalizHist* adalah fungsi yang digunakan untuk menormalisasikan nilai *brightness* dan meningkatkan nilai *contrast* pada gambar *grayscale* agar penyebaran histogram merata sehingga derajat keabuan memiliki jumlah pixel yang relative sama. Ilustrasi penggunaan *Equalize Histogram* dapat dilihat pada Gambar 2.3.



(b) Original Image (a) Original Histogram



(d) *New Image*      (c) *New Histogram*

**Gambar 2.3 Equalize Histogram**

Sumber: (docs.opencv.org)

**2.4.2.6 Operasi Pencerminkan (*Flipping*)**

*Flipping* adalah operasi geometri yang tidak merubah ukuran suatu citra. Operasi ini hanya mengakibatkan adanya perubahan orientasi citra, baik secara horizontal, vertical maupun keduanya. Ilustrasi penggunaan operasi pencerminan dapat dilihat pada Gambar 2.4.



(a) *Original Image*      (c) *Horizontal Flip*      (b) *Vertical Flip*

**Gambar 2.4 Operasi Pencerminkan (*Flipping*)**

**2.4.2.7 Penghalusan (*Blur*)**

*Blur* adalah cara membuat efek untuk mengurai *noise* pada citra dengan menggunakan filter rata-rata. Filter rata-rata adalah melakukan proses pemfilteran citra  $f(x, y)$  dengan filter rata-rata  $g(x, y)$  untuk berbagai ukuran filter dari ukuran 3x3, 5x5 7x7 dan seterusnya. Dari ukuran filter tersebut akan didapatkan ukuran filter yang paling berpengaruh dan terhadap penghalusan citra. Ilustrasi penggunaan *blur* dapat dilihat pada Gambar 2.5.





**Gambar 2.5 Penghalusan (*Blur*)**

Sumber: (docs.opencv.org)

## 2.5 *Library* OpenCv

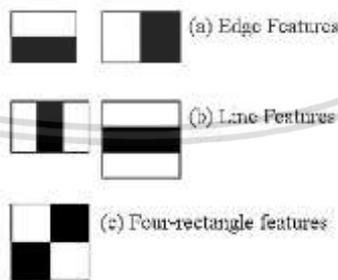
OpenCV dirilis di bawah lisensi BSD oleh karena itu gratis baik untuk penggunaan akademis dan komersial. OpenCv dapat di implementasikan dengan menggunakan berbagai bahasa pemrograman seperti C ++, C, Python dan Java interface dan mendukung Windows, Linux, Mac OS, iOS dan Android. OpenCV digunakan sebagai pendukung pengolahan citra dan memanfaatkan fungsi yang terdapat pada *library* tersebut.

## 2.6 *Haar Cascade*

*Haar Cascade* adalah metode untuk mengenali wajah dan bagiannya yang disediakan oleh *library* OpenCv yang diperkenalkan pertama kali oleh Viola and Jones pada tahun 2001. Metode ini menggunakan konsep fitur untuk membantu mengenali wajah dan bagiannya. Pada setiap fitur memiliki daerah gelap dan daerah terang.

*Haar Casacade* memiliki empat tahapan:

1. Deteksi fitur persegi yang disebut dengan fitur *Haar*, Gambar 2.6.



**Gambar 2.6 Fitur *Haar Cascade***

Sumber: (docs.opencv.org)

Cara untuk menentukan ada atau tidaknya fitur haar dengan cara mengurangi nilai rata-rata piksel daerah gelap dengan nilai rata-rata piksel daerah terang. Jika hasilnya diatas ambang maka fitur tersebut dikatakan ada.

Sebagai contoh terdapat citra masukan dengan dimensi 25x25 dengan nilai-nilai pixel sebagai berikut:

2	4	7	5	8
1	5	1	7	7
5	6	9	5	6
8	9	10	6	7
10	12	8	3	6

Hitam = 60

Putih = 47

Fitur Haar = 60 – 47  
= 13

Pada contoh ini menghasilkan nilai fitur 13, maka angka tersebut dapat ditentukan sebagai tingkat *luminance/haar like feature* citra untuk menentukan atau mencirikan bagian citra yang memiliki fitur haar.

2. *Integral Image* untuk menentukan ada atau tidaknya fitur *haar* pada citra yang sedang di deteksi.

*Integral image* merupakan teknik untuk mendapatkan nilai komputasi yang rendah. Teknik ini mendeteksi suatu objek dengan menggabungkan unit terkecil dari citra yaitu nilai-nilai piksel menjadi suatu representasi citra baru, caranya dengan menjumlahkan piksel yang ada di sebelah kiri dan atas titik tersebut.

Ilustrasi perhitungan *Integral Image*

1	1	1	→	1	2	3
1	1	1		2	4	6
1	1	1		3	6	9

Sebagai contoh terdapat citra masukan dengan dimensi 25x25 dengan nilai-nilai pixel sebagai berikut:

2	4	7	5	8
1	5	1	7	7
5	6	9	5	6
8	9	10	6	7
10	12	8	3	6



Maka *Integral Image* dari citra *input* diatas adalah:

2	6	13	18	26
3	12	20	32	47
8	23	40	57	78
16	40	67	90	118
26	62	97	123	157

Nilai piksel pada *integral image* ini didapatkan dengan cara menjumlahkan nilai suatu piksel dengan nilai piksel disebelah kiri, kiri atas dan atas dari piksel tersebut. Sebagai contoh, nilai dari piksel pada baris ke-2 dan kolom ke-2 pada ilustrasi di atas diperoleh dengan cara menjumlahkan nilai piksel baris ke-1 kolom ke-1, piksel baris ke-1 kolom ke-2, piksel baris ke-2 kolom ke-1 dengan piksel baris ke-2 kolom ke-2, maka didapatkan piksel dengan nilai 12 (2+4+1+5).

Cara menghitung selisih kotak hitam dan kotak putih sebagai berikut:

KONDISI SATU

A		C		
B		D		

KONDISI DUA

		A		C
		B		D

Maka nilai fitur *haar* adalah

$$Haar = |D+A-(B+C)| \tag{2.4}$$

Sehingga dapat diselesaikan dengan cara berikut:

*Integral Image*

2	6	13	18	26
3	12	20	32	47
8	23	40	57	78
16	40	67	90	118
26	62	97	123	157

*Haar* =  $|D+A-(B+C)|$

Hitam =  $97+2-(26+13)$   
 = 60

Putih =  $157+13-(97+26)$   
 = 47

Hitam - Putih =  $|47-60|$   
 = 13

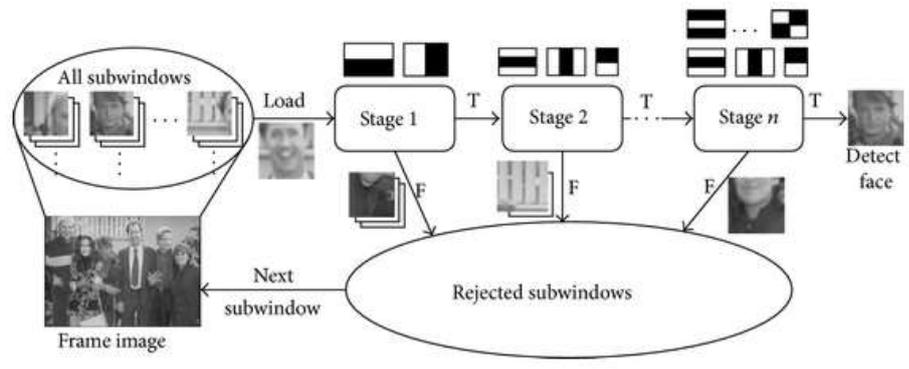
Hasil yang dihasilkan adalah 13. Hasil perhitungan pixel dengan atau tidak menggunakan *integral image* mendapatkan nilai yang sama sehingga perhitungan menggunakan *Integral Image* dapat digunakan sebagai perhitungan yang efektif untuk menghitung nilai *luminance/haar like feature*. Dalam metode *viola jones* tidak hanya menggunakan satu perhitungan seperti diatas namun menggunakan banyak sekali kemungkinan fitur yang diterapkan untuk menghitung nilai fitur *haar*.

3. *Adaptive-Boost (AdaBoost) Machine Learning*

Tahap ini adalah tahap selanjutnya setelah perhitungan nilai fitur didapatkan. Metode *AdaBoost* menggabungkan banyak *Classifier* lemah menjadi satu *Classifier* kuat. *Classifier* adalah suatu ciri yang menandakan adanya objek wajah dalam suatu citra. Sedangkan *Classifier* lemah adalah suatu jawaban benar namun memiliki tingkat kebenaran yang kurang akurat, jika digabung maka *Classifier* lemah tersebut akan menghasilkan suatu *Classifier* kuat.

Deteksi fitur *haar* menggunakan cara *sliding window* (membaginya ke beberapa *subwindows* dengan ukuran pixel tertentu) terhadap citra *input*. Sehingga dapat diketahui suatu fitur mempresentasikan tidaknya pada suatu citra *input*.

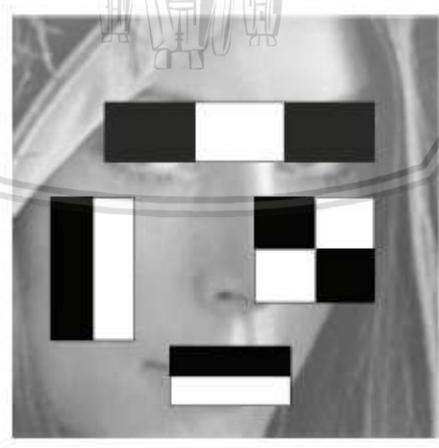




**Gambar 2.7 Metode AdaBoost**

Sumber: (Researchgate.net)

Gambar 2.7 Menjelaskan bagaimana metode AdaBoost akan menyeleksi fitur haar dari citra *input*. Citra *input* dideteksi dengan cara dibagi menjadi *subwindows* yang dibaca nilai fiturnya kemudian di cek pada setiap *stage*. Stage berisi kumpulan fitur *haar* yang telah dikelompokkan dari banyak kemungkinan fitur yang ada. Kumpulan fitur terbaik yang ada dalam *stage* ditulis dalam bentuk *.xml* pada library OpenCv. Viola dan Jones dalam papernya di tahun 2001 menyatakan ada sekitar 6000 fitur yang kemudian dibagi menjadi 38 *stage* yang digunakan pada algoritma *AdaBoost*. Ketika dilakukan pengecekan pada tahap *stage* dan ditemukan ketidakcocokan fitur sebelum pengecekan *stage* selesai maka, *subwindows* akan ditolak untuk melanjutkan ke *stage* berikutnya dan kembali pada tahap awal. Namun, jika *subwindows* berhasil melewati setiap *stage* yang ada maka akan mendapatkan hasil deteksi wajah. Proses Penyeleksian fitur yang dilakukan algoritma *AdaBoost* dapat dilihat pada Gambar 2.7.



**Gambar 2.8 Implementasi Fitur Haar pada citra input**

Sumber: (docs.opencv.org)

Tingkatan penyeleksian fitur haar diurutkan berdasarkan bobot nilai terberat hingga teringan sehingga waktu yang dibutuhkan untuk mengklasifikasikan bahwa tidak terdeteksi wajah dalam citra masukan sangat sedikit atau cepat, urutan proses penyeleksian fitur haar ini disebut sebagai *cascade Classifier*.

## 2.7 Skin Detection

*Skin Detection* merupakan tahap untuk mengenali warna kulit yang digunakan untuk mempermudah proses deteksi wajah dan bagiannya. Metode yang digunakan pada tahap ini adalah penggabungan warna *RGB*, *Hue* dan *Cb-Cr* yang disebut dengan *RGB-H-CbCr Skin Color*.

Aturan yang digunakan untuk mendeteksi warna kulit *RGB* menggunakan aturan dari Peer et al (2003).

Dimana, aturan standar pencahayaan siang hari:

$$\begin{aligned} &(R > 95) \text{ AND } (G > 40) \text{ AND } (B > 20) \text{ AND} \\ &(\max \{R, G, B\} - \min \{R, G, B\} > 15) \text{ AND} \\ &(|R - G| > 15) \text{ AND } (R > G) \text{ AND } (R > B) \end{aligned} \quad (2.5)$$

sedangkan warna kulit di bawah *flashlight* atau *dailylight lateral*:

$$\begin{aligned} &(R > 220) \text{ AND } (G > 210) \text{ AND } (B > 170) \text{ AND} \\ &(|R - G| \leq 15) \text{ AND } (R > B) \text{ AND } (G > B) \end{aligned} \quad (2.6)$$

Untuk mempertimbangkan kedua kondisi saat diperlukan, maka digunakan fungsi OR untuk menggabungkan kedua aturan (1) dan aturan (2). Sehingga untuk aturan RGB didapatkan

$$\text{Rule A: Persamaan (1) } \cup \text{ Persamaan (2)} \quad (2.7)$$

Berdasarkan penelitian yang dilakukan Abdul Rahman dkk pada tahun 2007 bahwa subruang *Cb-Cr* adalah diskriminan warna kulit yang kuat. Maka, aturan pembatas untuk warna kulit *Cb-Cr* dirumuskan sebagai berikut:

$$Cr \leq 1.5862 \times Cb + 20 \quad (2.8)$$

$$Cr \geq 0.3448 \times Cb + 76.2069 \quad (2.9)$$

$$Cr \geq -4.5652 \times Cb + 234.5652 \quad (2.10)$$

$$Cr \leq -1.15 \times Cb + 301.75 \quad (2.11)$$

$$Cr \leq -2.2857 \times Cb + 432.85 \quad (2.12)$$

Aturan (3) hingga (7) digabungkan menggunakan AND untuk mendapatkan aturan pembatasan CbCr.

$$\text{Rule B: Persamaan (3) } \cap \text{ Persamaan (4) } \cap \text{ Persamaan (5) } \cap \text{ Persamaan (6) } \cap \text{ Persamaan (7)} \quad (2.13)$$

Di ruang *HSV*, nilai *hue* paling menonjol terlihat pemisahan antara daerah kulit dan non-kulit. Maka sebagai batas *H*:

$$H < 25 \quad (2.14)$$

$$H > 230 \quad (2.15)$$

di mana kedua aturan digabungkan dengan fungsi OR, dilambangkan sebagai Rule C.

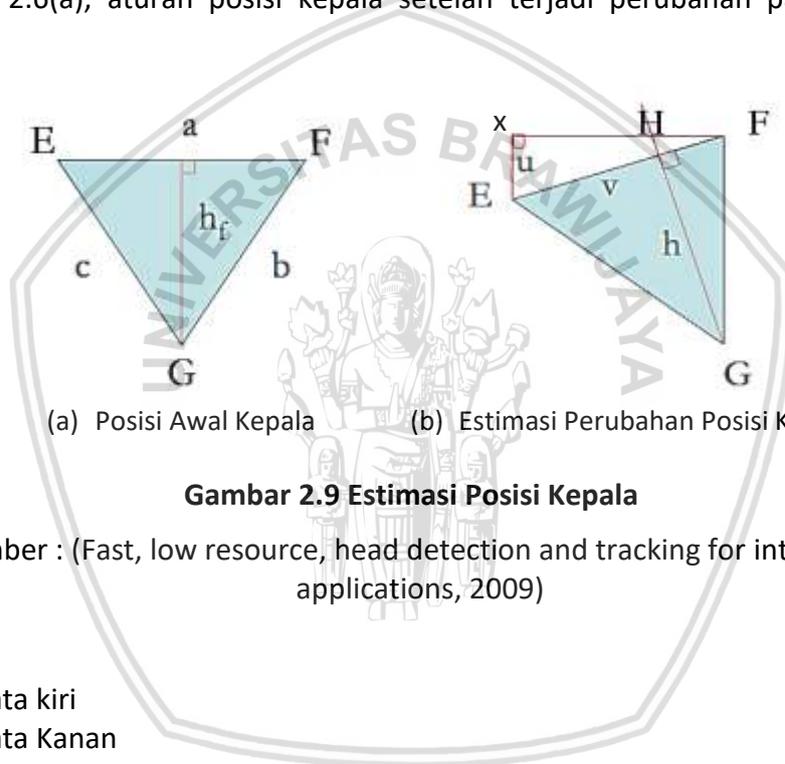
$$\text{Rule C: Persamaan (9) U Persamaan (10)} \quad (2.16)$$

Setelah itu, setiap piksel yang memenuhi *Rule A*, *Rule B* dan Aturan C diklasifikasikan sebagai piksel warna kulit,

$$\text{Rule A} \cap \text{Rule B} \cap \text{Rule C} \quad (2.17)$$

## 2.8 Triangle Geometry

*Triangle geometry* adalah metode yang digunakan untuk menganalisa arah pergerakan kepala. Citra *input* berupa video *real-time* dianggap sebagai posisi relative, mata dan mulut yang akan berubah ketika posisi kepala. Sehingga dapat menganalisis posisi kepala berdasarkan keadaan *real-time* serta menunjukkan nilai *pitch*, *yaw*, *roll*. Aturan posisi awal kepala sebelum terjadi perubahan pada Gambar 2.6(a), aturan posisi kepala setelah terjadi perubahan pada Gambar 2.6(b).



(a) Posisi Awal Kepala      (b) Estimasi Perubahan Posisi Kepala

**Gambar 2.9 Estimasi Posisi Kepala**

Sumber : (Fast, low resource, head detection and tracking for interactive applications, 2009)

dimana:

- $E$  = Mata kiri
- $F$  = Mata Kanan
- $G$  = Hidung
- $a$  = Jarak mata kiri dan kanan
- $b$  = Jarak mata kanan dengan hidung
- $c$  = Jarak mata kiri dengan hidung
- $hf$  = Jarak titik tengah kedua mata dengan hidung
- $u$  = Panjang antara titik  $x$  dengan  $E$
- $x$  = Perubahan posisi mata
- $v$  = Perubahan jarak posisi kedua mata
- $h$  = Perubahan jarak titik tengah mata dengan hidung

Nilai untuk kemiringan, putaran serta posisi *pitch*, *yaw* dan *roll* didapatkan dari hasil proses menggunakan metode *triangle geometry*. Ilustrasi persamaan Triangle geometry pada kepala dapat dilihat pada Gambar 2.8. Segitiga *EFG*

dianggap posisi mata kiri, mata kanan dan ujung hidung. Peneliti berhipotesis bahwa tampilan ini adalah segitiga sama kaki.

$$a = b = c \tag{2.18}$$

$$hf = \frac{a\sqrt{3}}{2} \tag{2.19}$$

Ketika kepala berubah posisi kita dapat menentukan nilai dengan rumus berikut:

$$yaw = \frac{v - \frac{a}{2}}{a} \quad \text{dimana, } v = \frac{\|\vec{EH}\|}{\|\vec{EF}\|} = \frac{\vec{EF} \cdot \vec{EG}}{\|\vec{EF}\|^2} \tag{2.20}$$

$$pitch = \frac{h}{hf} \tag{2.21}$$

$$roll = \frac{u}{a} \tag{2.22}$$



**Gambar 2.10 Ilustrasi *Triangle geometry* pada kepala**

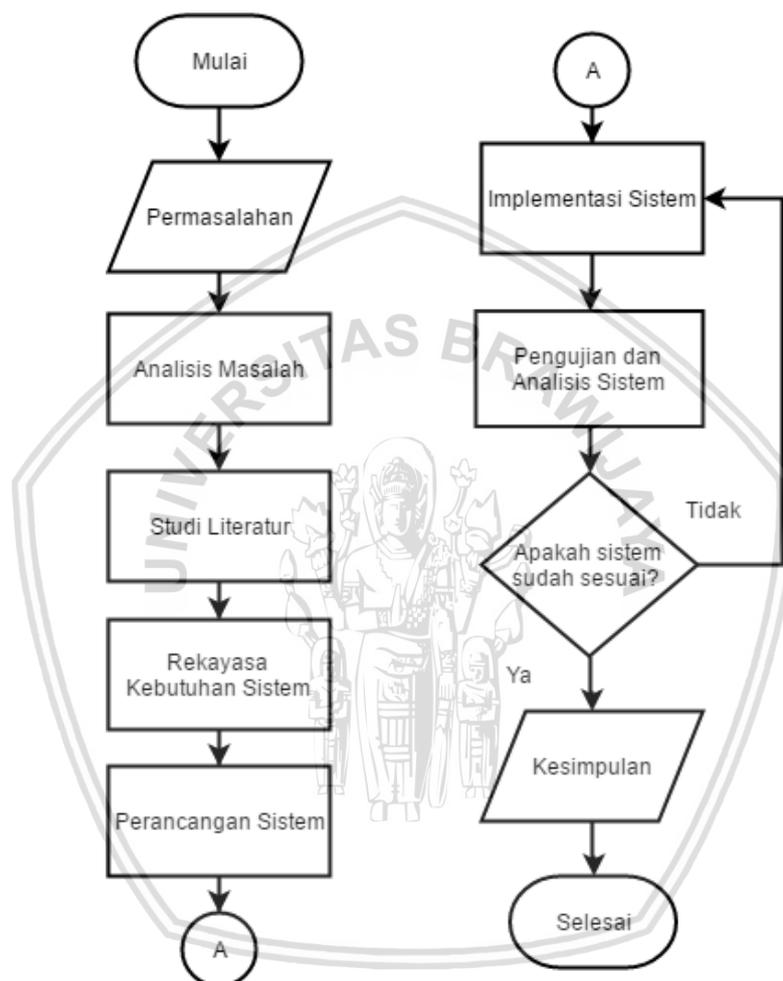
Ketika posisi kepala berubah, tidak lagi berada pada posisi awal maka rumus untuk deteksi arah pergerakan kepala dapat diilustrasikan seperti pada Gambar 2.9, posisi kepala berubah berdasarkan pergerakan arah *roll*.

**Gambar 2.11 Ilustrasi *Triangle Geometry* pada posisi *roll***



### BAB 3 METODOLOGI PENELITIAN

Metode penelitian membahas mengenai alur pelaksanaan penelitian terhadap sistem yang akan dibuat sehingga proses penelitian dapat terarah dengan baik sesuai tujuan penelitian. Pada bab ini akan dijelaskan langkah-langkah yang dilakukan dalam analisis sistem yang dirancang. Alur metode penelitian dapat dilihat pada diagram alir Gambar 3.1.



**Gambar 3.1 Alur Metode Penelitian**

Alur penelitian diawali dengan analisis masalah yang akan diangkat menjadi sebuah penelitian, kemudian mencari literatur sebagai pendukung sebagai bahan referensi untuk melakukan penelitian. Proses selanjutnya yaitu menganalisis kebutuhan untuk melakukan perancangan dan implementasi sistem. Setelah sistem selesai, dilakukan pengujian sistem untuk menguji apakah hasil yang didapat sesuai dengan yang diinginkan, jika tidak sesuai maka mengulangi proses implementasi sistem, dan jika hasil yang didapatkan sesuai maka didapat beberapa kesimpulan dari penelitian tersebut.



### 3.1 Analisis Masalah

Analisis masalah dilakukan untuk menarik kesimpulan terhadap suatu kondisi sehingga menghasilkan pokok permasalahan untuk diselesaikan pada penelitian ini. Analisis masalah dilakukan pada kondisi *real-time* posisi kepala dengan sistem. Dalam analisis tersebut ditarik kesimpulan untuk membuat rumusan masalah terhadap kondisi yang ada.

### 3.2 Studi Literatur

Pada tahap studi literatur dilakukan untuk pembelajaran literatur dari jurnal, paper, buku, pakar, informasi dari internet dan penelitian yang telah dilakukan sebelumnya yang berhubungan dengan deteksi arah pergerakan kepala. Studi literatur dilakukan untuk mendapatkan informasi tentang penerapan *triangle geometry* pada penelitian yang berjudul "Implementasi Metode Triangle Geometry Untuk Pengenalan Arah Pergerakan Kepala" ini. Berikut ini adalah bidang ilmu yang dipelajari pada penelitian, antara lain:

1. *Skin detection* menggunakan RGB-H-CbCr.
2. Metode *Haar Cascade*.
3. Metode *Triangle Geometry*.
4. Implementasi *Haar cascade* menggunakan *Library OpenCv 2.4.9*.

### 3.3 Rekayasa Kebutuhan Sistem

Rekayasa kebutuhan dalam perancangan sistem bertujuan untuk mendefinisikan kebutuhan-kebutuhan dalam penelitian seperti kebutuhan perangkat keras dan perangkat lunak, rekayasa kebutuhan dibutuhkan agar menjadi parameter kemampuan sebuah sistem saat diimplementasikan, seperti kebutuhan sistem berikut:

1. Kamera harus mampu mengambil gambar video secara *real-time*.
2. Citra *input* mendapatkan intensitas cahaya yang cukup agar sistem dapat berjalan.

#### 3.3.1 Kebutuhan Perangkat Keras

Perangkat keras yang digunakan dalam penelitian ini yaitu :

1. 1 buah kamera webcam logitech C920 HD1080p
2. 1 buah Laptop Dell Inspiron 5447

Processor : Intel(R) Core(TM) i3-4030U CPU @ 1.90GHz (4 CPUs), ~1.9GHz

Memori : 16GB

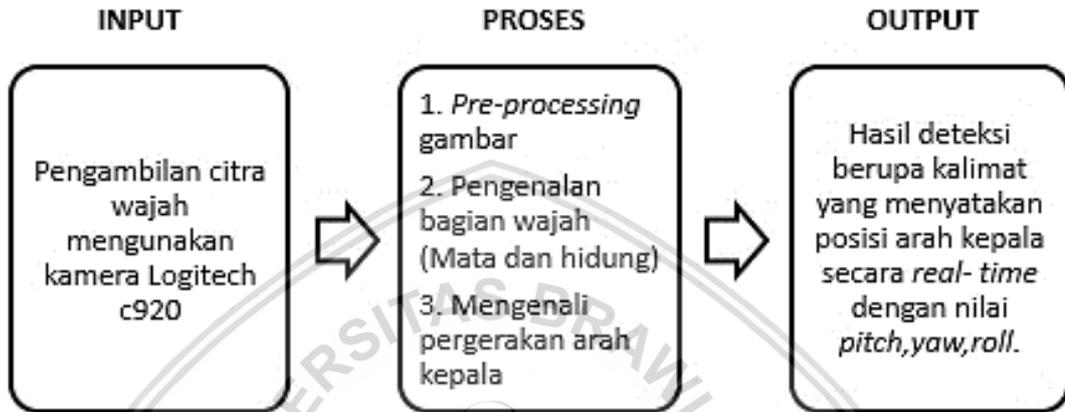
#### 3.3.2 kebutuhan Perangkat Lunak

Adapun kebutuhan perangkat lunak yang digunakan untuk mewujudkan sistem pada penelitian ini, antara lain:

1. IDE Eclipse Helios
2. Sistem Operasi Ubuntu 10.4

### 3.4 Perancangan Sistem

Perancangan sistem adalah tahapan yang dilakukan untuk merancang sistem. Menjabarkan mengenai kinerja sistem secara terstruktur, dimulai dari *input* hingga *output* yang dihasilkan. Berikut adalah diagram model perancangan penelitian ini, pada gambar 3.2.



Gambar 3.2 Blok Diagram Sistem

Pada gambar 3.2 dapat dilihat bahwa perancangan sistem terdiri dari tiga proses utama, yaitu:

1. *Input*  
*Input* adalah masukan wajah pengguna berupa gambar yang diambil oleh kamera Logitech C920 1080HD.
2. *Proses*  
Proses yang dilakukan pertama adalah *pre-processing* pada gambar dengan tahapan *flip gambar*, *skin detection* kemudian dilakukan deteksi bagian dari wajah dengan *Haar Cascade* menggunakan *library OpenCv* dan selanjutnya dilakukan deteksi pergerakan posisi kepala menggunakan metode *triangle geometry* yang dilakukan secara *real time*.
3. *Output*  
*Output* adalah keluaran dari sistem diharapkan berupa kalimat yang menyatakan posisi arah kepala secara *real time* dengan nilai *pitch,yaw,roll*.

### 3.5 Implementasi Sistem

Pada tahap ini dilakukan implementasi sistem yang sesuai dengan perancangan sistem yang telah dilakukan. Implementasi meliputi hal sebagai berikut:

1. Implementasi program dengan menggunakan bahasa pemrograman c++.
2. Implementasi metode RGB-H-CbCr untuk mengenali warna kulit.

3. Implementasi metode *haar cascade* dengan menggunakan *library OpenCv* 2.4.9 untuk mengenali wajah dan bagiannya (Mata dan hidung).
4. Implementasi metode *triangle geometry* untuk mengenali arah pergerakan kepala.

### 3.6 Pengujian Sistem

Pengujian dilakukan untuk mengetahui fakta-fakta dari sistem. Pada pengujian ini dilakukan pengujian untuk menjawab rumusan masalah apakah sudah terjawab atau belum. Pada tahap pengujian sistem ini menggunakan parameter yang disesuaikan pada perancangan sistem. Beberapa rancangan skenario pengujian sistem adalah sebagai berikut:

1. Pengujian kesesuaian deteksi warna kulit, mata dan hidung.
2. Pengujian batas atas dan batas bawah nilai *yaw*, *pitch* dan *roll*.
3. Pengujian akurasi masing - masing arah pergerakan kepala pada *output* sistem dengan keadaan nyata citra gambar.

$$Akurasi = \frac{\text{jumlah nilai keberhasilan}}{\text{Jumlah frame} \times (\text{Jumlah parameter uji})} \times 100\% \quad (3.1)$$

### 3.7 Analisis dan Pembahasan

Setelah semua tahap dilakukan, mulai dari implementasi dan pengujian, dilakukan tahap selanjutnya, yaitu analisis dan pembahasan. Pada tahap ini dilakukan analisis terhadap hasil semua hasil pengujian dan melakukan pembahasan, untuk menentukan nilai dari hasil pengujian sudah benar atau tidak, sehingga dapat dikatakan pengujian terhadap sistem berhasil.

### 3.8 Kesimpulan

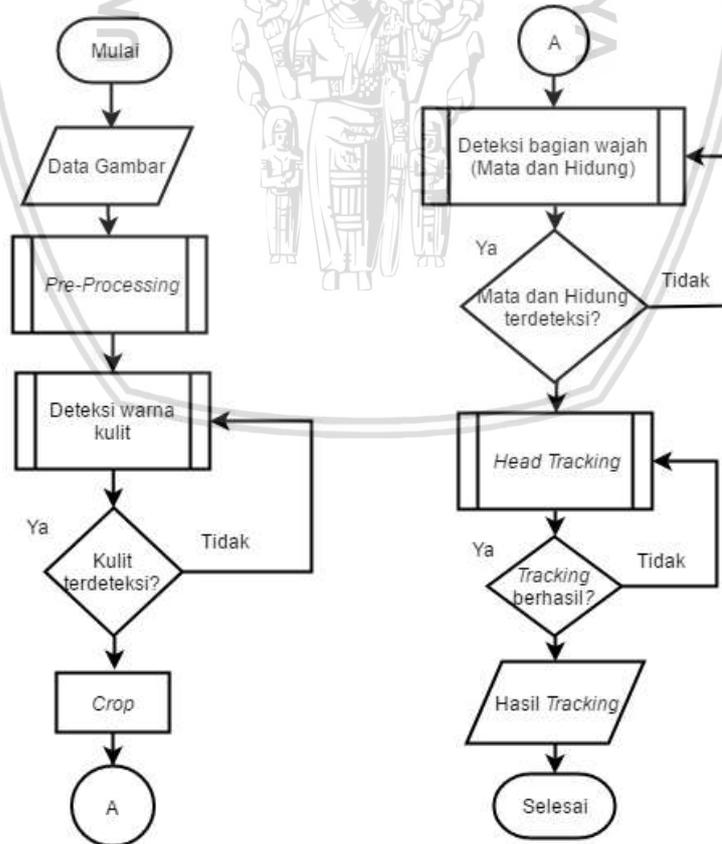
Setelah semua tahap dilakukan, mulai dari implementasi dan pengujian, dilakukan tahap selanjutnya, yaitu analisis dan pembahasan. Pada tahap ini dilakukan analisis terhadap hasil semua hasil pengujian dan melakukan pembahasan, untuk menentukan nilai dari hasil pengujian sudah benar atau tidak, sehingga dapat dikatakan pengujian terhadap sistem berhasil. Proses penarikan kesimpulan dilakukan setelah semua tahapan meliputi perancangan, implementasi, dan pengujian sistem telah selesai dilakukan. Tahap akhir dari penulisan adalah adanya saran yang dimaksudkan untuk memperbaiki kekurangan yang ada, serta untuk memberikan pertimbangan atas pengembangan metode selanjutnya.

## BAB 4 PERANCANGAN

Bab ini menjelaskan terkait dengan perancangan dan implementasi. Perancangan pada bab ini berisi perancangan proses dan perancangan pengujian.

### 4.1 Perancangan Proses

Pada sistem ini, digunakan kamera webcam logitech C920 HD1080p sebagai alat pengambilan video secara *realtime* untuk masukan yang kemudian akan diproses di dalam sistem. Masukan tersebut diolah dalam tahap *pre-processing* terlebih dahulu sebelum dilakukannya penerapan metode. Tahapan yang dilakukan dalam *pre-processing* antara lain adalah *flip*, *blur*, dan perbaikan nilai *contras* dan *brightness*. Tahap *pre-processing* berguna untuk memudahkan tahap selanjutnya, yaitu deteksi warna kulit dengan menggunakan metode *RGB-H-CbCr skin colour*. Setelah dilakukan deteksi warna kulit selanjutnya adalah *crop* untuk memfokuskan gambar pada daerah wajah agar mempermudah deteksi bagian wajah, dilakukan deteksi bagian wajah dengan memanfaatkan *library OpenCv (Haar cascade)*. Pada tahap ini akan terbagi menjadi deteksi mata dan hidung. Hasil dari proses ini dilakukan penerapan metode *triangle geometry* yang akan menghasilkan nilai *yaw*, *pitch*, *roll*. Nilai tersebut akan diklasifikasikan ke dalam *range* arah untuk mengenali posisi arah kepala secara *realtime*.



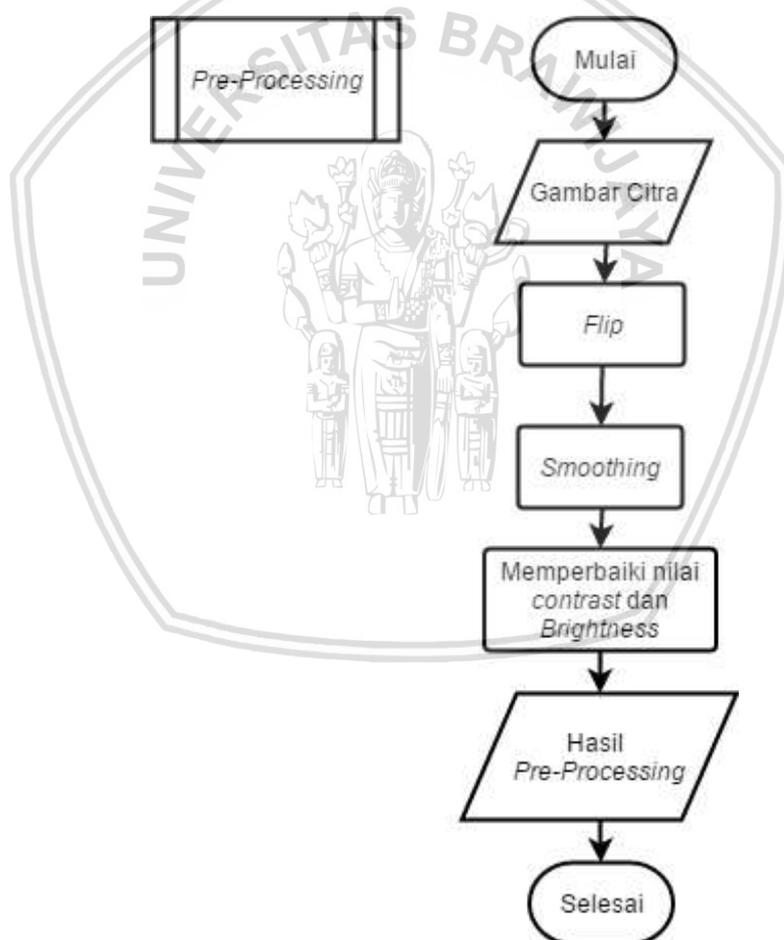
Gambar 4.1 Flowchart Utama Sistem



*Range* nilai yang didapatkan dari hasil data uji terhadap pengguna yang diambil setelah sistem ini selesai sampai tahap menghasilkan nilai *yaw*, *pitch*, *roll*. Sistem ini menghasilkan keluaran berupa kalimat yang menyatakan posisi arah pergerakan kepala pengguna secara *real time* dengan nilai *yaw*, *pitch*, *roll* yang telah didapatkan dari perhitungan metode. Detail dari perancangan proses sistem ini dapat dilihat pada Gambar 4.1.

#### 4.1.1 Pre-Processing

Pada tahap ini dilakukan beberapa proses antara lain *flip* gambar yang berfungsi mengubah sudut pandang gambar menjadi seperti kaca atau kondisi nyata. Kemudian dilakukan *blur* pada gambar untuk mengurangi *noise* pada gambar, mendapatkan hasil gambar yang lebih halus. Tahap berikutnya adalah pengkoreksian nilai *contrast* dan *brightness* untuk membantu ketika kondisi pencahayaan kurang dan memudahkan deteksi bagian wajah.

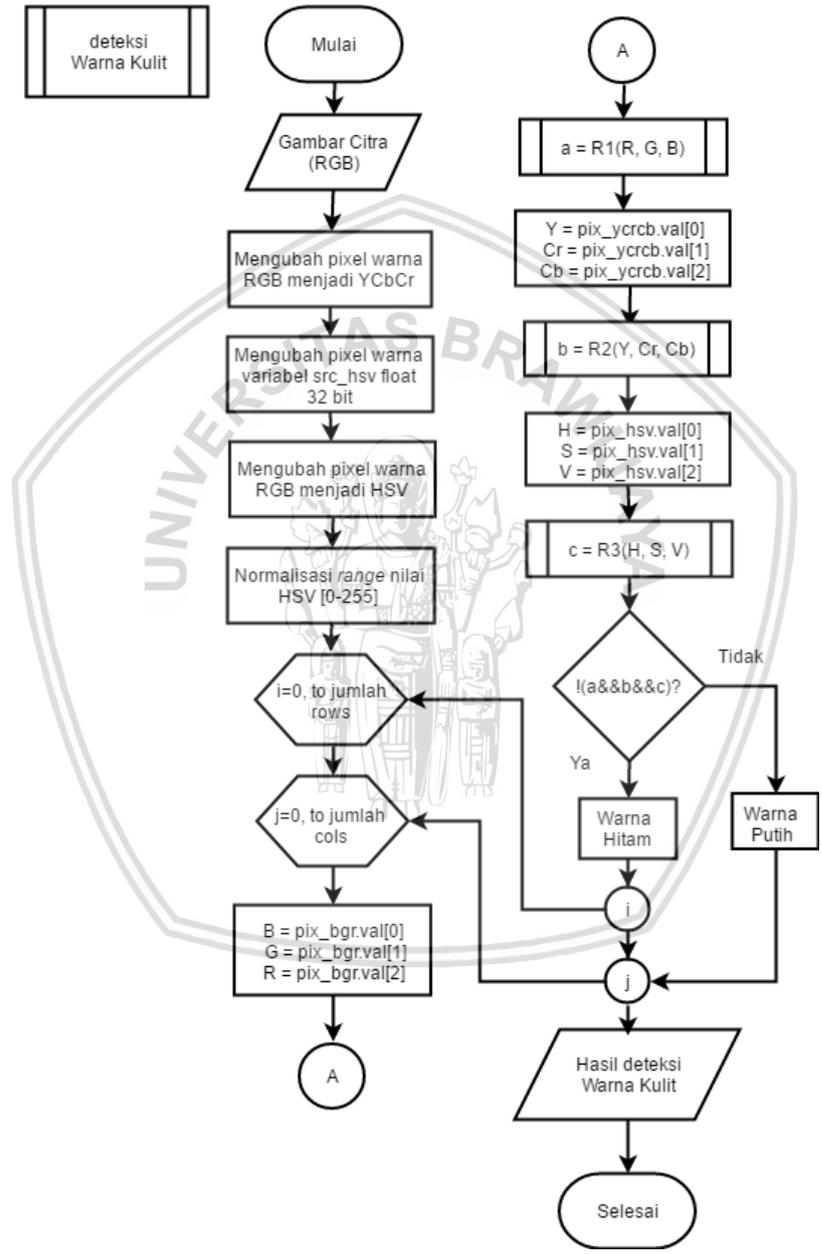


Gambar 4.2 FlowChart Pre-Processing

#### 4.1.2 Deteksi Warna Kulit (Skin Detection)

Deteksi warna kulit adalah proses deteksi pixel warna kulit yang terdapat pada gambar citra dengan cara mengimplementasikan aturan nilai metode *RGB-H-CbCr*.

Metode ini menggunakan Gambar citra yang semula *RGB* dirubah menjadi *HSV* dan *YCbCr* kemudian penyatuan pixel warna *RGB* (*Red, Green, Blue*), *Hue* dan *CbCr* sehingga mendapat penggabungan warna *RGB-H-CbCr*. Hasil dari proses ini adalah penandaan warna kulit dengan warna putih dan penandaan warna hitam untuk warna selain warna kulit. Proses deteksi warna kulit dapat dilihat pada Gambar 4.3.

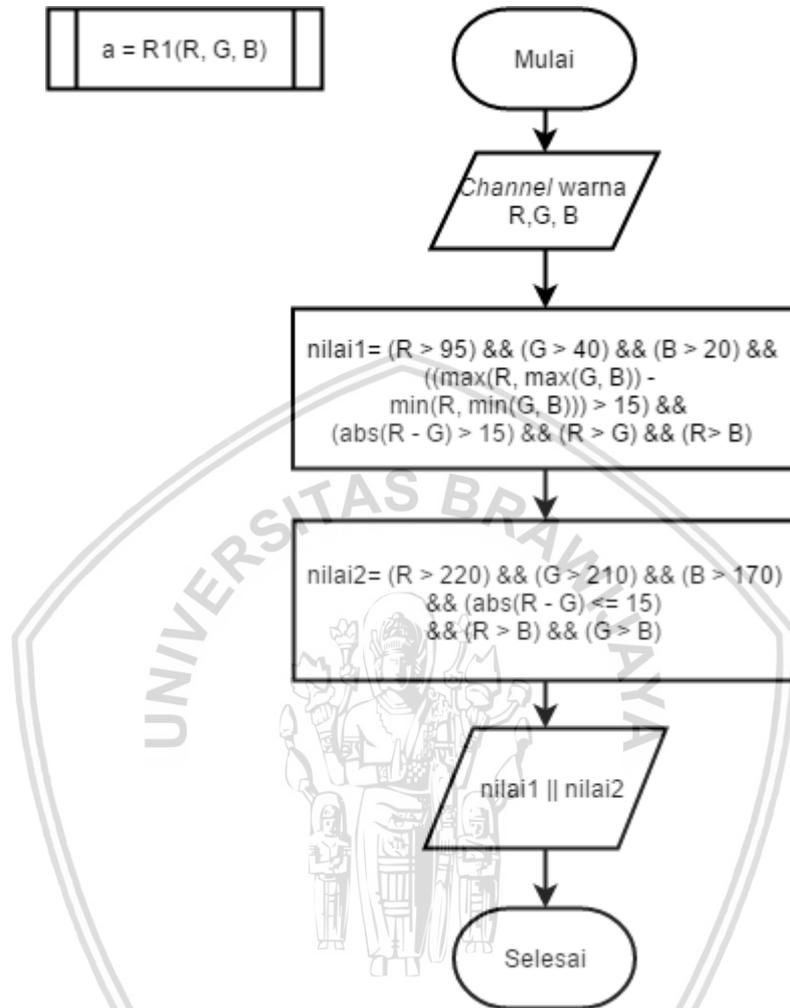


Gambar 4.3 Flowchart Deteksi Warna Kulit

4.1.2.1 Aturan Warna RGB

Aturan RGB menerapkan dua aturan nilai yaitu pada saat kondisi standar cahaya pencahayaan siang hari dan kondisi ketika warna kulit di bawah *flash light*

atau *daily light lateral*. Aturan nilai untuk dua kondisi tersebut dapat dilihat pada Gambar 4.4.

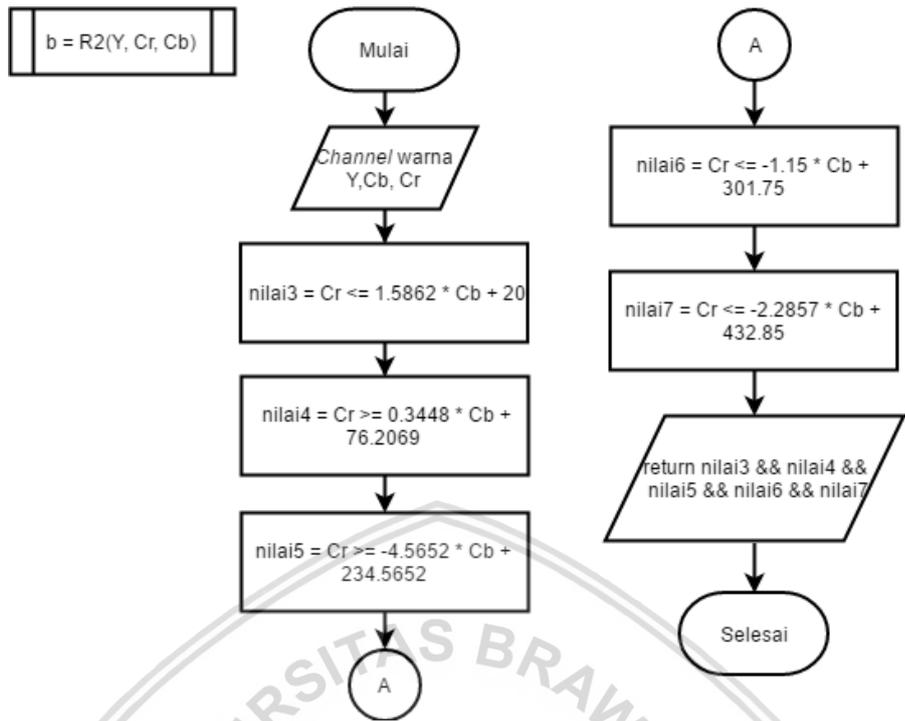


Gambar 4.4 Flowchart Aturan Warna RGB

#### 4.1.2.2 Aturan Warna YCbCr

Pada aturan warna *YCbCr* terdapat lima aturan warna untuk mendeteksi warna kulit. Lima aturan warna tersebut hanya melibatkan *channel* warna *Cb* dan *Cr*. Setelah masing – masing nilai sudah didapatkan maka akan di return untuk dilakukan penggabungan dengan *channel* warna yang lain dan dijalankan proses deteksi terhadap citra gambar. Flowchart aturan warna *YCbCr* dapat dilihat pada Gambar 4.5.

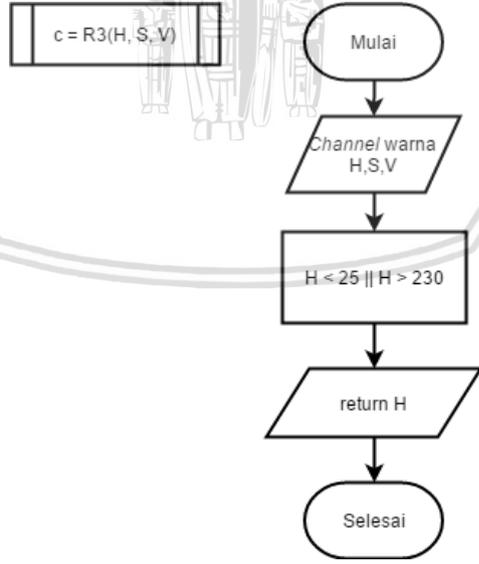




Gambar 4.5 Flowchart Aturan Warna YCbCr

4.1.2.3 Aturan Warna HSV

Pada aturan warna HSV yang berpengaruh terhadap deteksi warna kulit hanya channel warna hue. Channel warna ini memiliki aturan  $H < 25 \parallel H > 230$ . Flowchart aturan warna HSV dapat dilihat pada Gambar 4.6.



Gambar 4.6 Flowchart Aturan Warna HSV

4.1.3 Deteksi Bagian Wajah

Deteksi bagian wajah adalah proses deteksian mata dan hidung sebagai titik acuan untuk melakukan untuk melakukan proses head tracking. Tahap ini

memanfaatkan *library* OpenCv sebagai metode untuk deteksi bagian mata dan hidung dalam sistem. Awalnya, gambar citra yang sudah melalui tahap deteksi warna kulit diproses menjadi pixel *gray* yang kemudian dilakukan deteksi mata dan hidung menggunakan *library* OpenCV. Hasil yang di dapatkan berupa titik yang terbentuk di bagian tangan mata dan hidung. Proses deteksi wajah dapat dilihat pada Gambar 4.7.

#### 4.1.3.1 Deteksi Mata

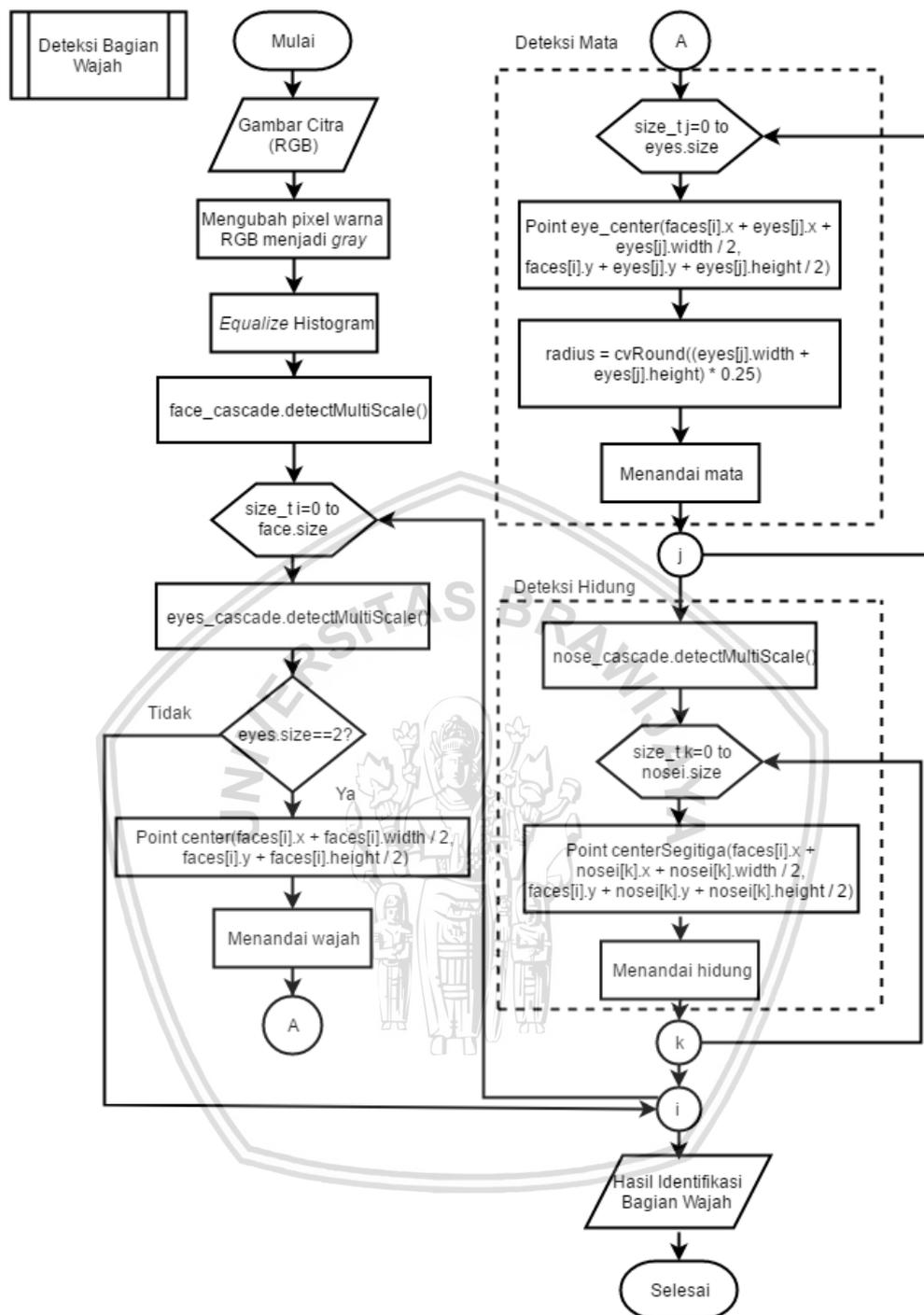
Pada tahap ini memanfaatkan *library openCv 2.4.9 (Haar Cascade)* untuk dapat mendeteksi mata. Langkah – langkah untuk mendeteksi mata adalah sebagai berikut.

1. Mulai.
2. Melakukan pembacaan pixel citra.
3. Melakukan perhitungan untuk menentukan titik pusat mata.
4. Menentukan radius mata.
5. Menandai bagian mata.
6. Selesai

#### 4.1.3.2 Deteksi Hidung

Pada tahap ini memanfaatkan *library openCv 2.4.9 (Haar Cascade)* untuk dapat mendeteksi hidung. Langkah – langkah untuk mendeteksi hidung adalah sebagai berikut.

1. Mulai
2. Memanggil method `nose_cascade.detectMultiScale`.
3. Melakukan pembacaan pixel citra.
4. Melakukan perhitungan untuk menentukan titik hidung.
5. Menandai hidung.
6. Selesai.

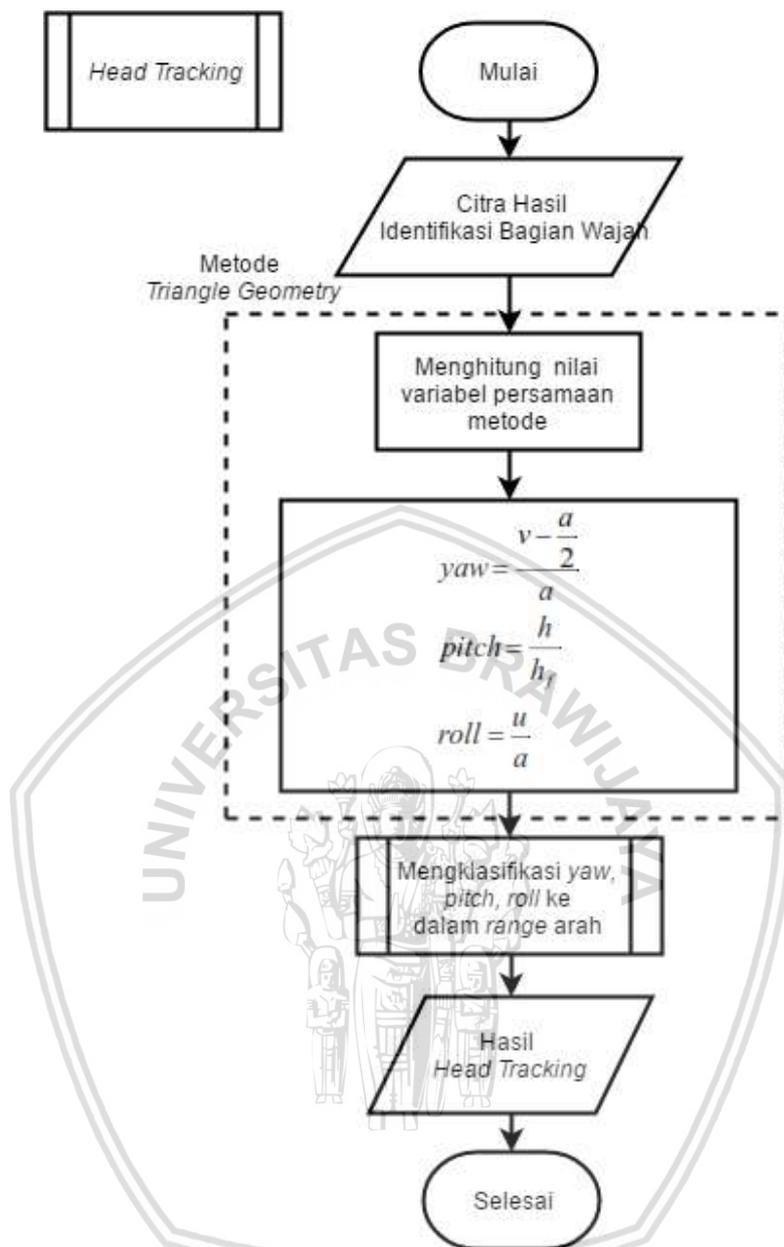


Gambar 4.7 Flowchart Deteksi Wajah

#### 4.1.4 Head Tracking

Head Tracking berfungsi sebagai deteksi gerakan arah kepala dari hasil perhitungan deteksi wajah berupa nilai *yaw*, *pitch*, *roll* dengan menggunakan metode *triangle geometry* yang kemudian di klasifikasikan ke dalam *range* nilai arah untuk menghasilkan keluaran berupa kalimat yang menyatakan arah pergerakan kepala seperti kanan, kiri, atas dan bawah. Proses pada tahap ini dapat dilihat pada Gambar 4.8.





Gambar 4.8 Flowchart Head Tracking

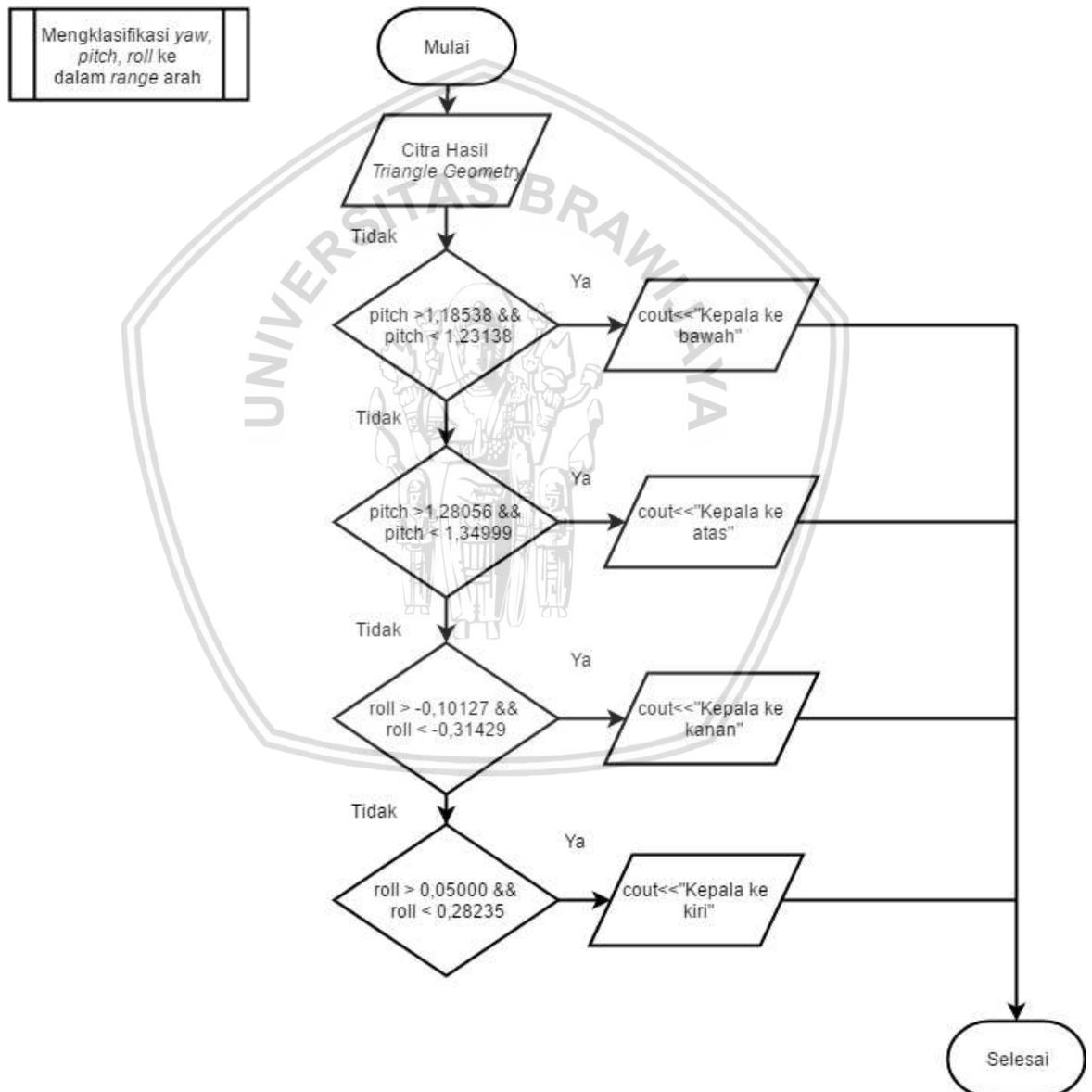
#### 4.1.4.1 Triangle Geometry

*Triangle Geometry* adalah metode yang digunakan untuk mengubah data citra gambar menjadi nilai yang dapat diolah dengan operasi matematika sesuai dengan rumus metode *triangle geometry*, menghasilkan persamaan nilai dari *yaw*, *pitch*, *roll*. Langkah-langkah *triangle geometry* adalah sebagai berikut.

1. Mulai.
2. Menghitung setiap nilai variabel persamaan metode.
3. Menghitung nilai *pitch*, *yaw*, *roll* sesuai dengan persamaan *triangle geometry*.
4. Selesai.

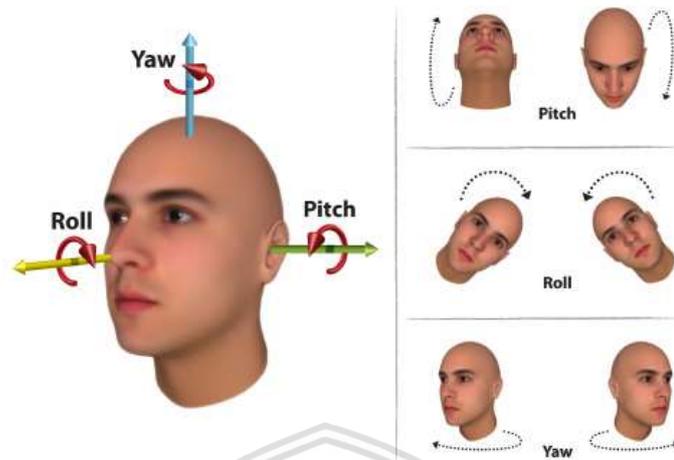
#### 4.1.4.2 Klasifikasi Range Arah

Klasifikasi *range* arah adalah proses dimana hasil dari *triangle geometry* di kelompokkan menurut arah posisi kepala antara lain kanan, kiri, atas dan bawah. *Range* arah diperoleh dari hasil pengambilan sample sistem dengan *output* hasil perhitungan metode *triangle geometry* sebelumnya yang dikumpulkan dan diolah. *Output* hasil metode berupa nilai *yaw*, *pitch*, *roll* (Gambar 4.10) yang kemudian di cari nilai terbesar dan terkecilnya untuk di jadikan batak atas dan batas bawah pada klasifikasi *range* arah. Flowchart klasifikasi *range* arah dapat dilihat pada Gambar 4.9.



Gambar 4.9 Flowchart Klasifikasi Range Arah





**Gambar 4.10 Orientasi Gerakan Kepala**

dimana:

- Yaw* : Menoleh
- Pitch* : Mengangguk
- Roll* : Gerakan patah

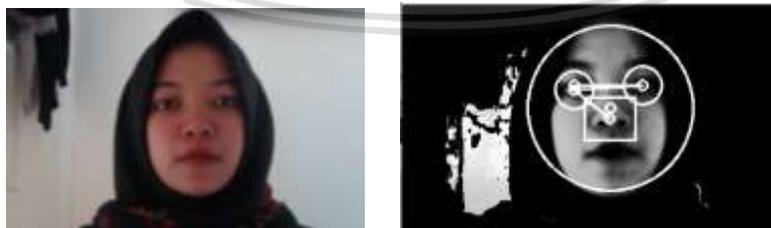
## 4.2 Perhitungan Manual

### 4.2.1 Nilai *yaw*, *pitch*, *roll*

Data yang diberikan adalah citra *RGB* yang diubah menjadi citra *gray*. Kondisi citra *gray* sudah berhasil mendeteksi mata dan hidung dengan arah pergerakan kepala atas, bawah, kanan dan kiri.

#### 1. Arah Atas

Data citra menunjukkan arah pergerakan kepala ke atas.



(b) Citra *RGB*

(a) Citra *Gray*

**Gambar 4.11 Citra Arah Atas**



Diketahui nilai citra:

```
wajah [219, 112]
eye_center 0 [252, 86]
eye_center 1 [181, 90]
nilai panjang a-71
nilai tengah a-35.5
koordinat titik tengah a[216, 86]
centerhidung[219, 121]
tinggiHf35
panjang EG 49.8523
koordinat x [252, 90]
panjang u [252, -4]
panjang u -4
panjang xF -71
panjang EF 71.1126
panjang v 49.8523
nilai o 21.2603
nilai h 45.0916
```

$$yaw = \frac{49,8523 - \frac{-71}{2}}{-71}$$

$$= -1,20214$$

$$pitch = \frac{45,0916}{35}$$

$$= 1,28833$$

$$Roll = \frac{-4}{-71}$$

$$= 0,05634$$

2. Arah Bawah

Data citra menunjukkan arah pergerakan kepala ke bawah.



(a) Citra RGB (b) Citra Gray

**Gambar 4.12 Citra Arah Bawah**

Diketahui nilai citra:

```
wajah [214, 155]
eye_center 0 [255, 134]
eye_center 1 [182, 140]
nilai panjang a-73
```

```

nilai tengah a-36.5
koordinat titik tengah a[218, 134]
centerihidung[222, 186]
tinggiHf52

panjang EG 63.5315
koordinat x [255, 140]
panjang u [255, -6]
panjang u -6
panjang xF -73
panjang EF 73.2462
panjang v 63.5315
nilai o 9.71467
nilai h 62.7844
    
```

$$yaw = \frac{63,5315 - \frac{-73}{2}}{-73}$$

$$= -1,37029$$

$$pitch = \frac{62,7844}{52}$$

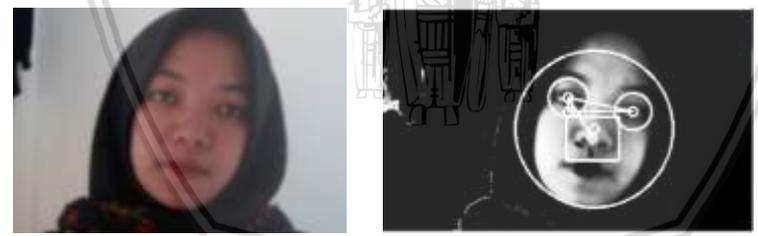
$$= 1,20739$$

$$Roll = \frac{-6}{-73}$$

$$= 0,08219$$

3. Arah Kanan

Data citra menunjukkan arah pergerakan kepala ke kanan.



(a) Citra RGB (b) Citra Gray

**Gambar 4.13 Citra Arah Kanan**

Diketahui nilai citra:

```

wajah [230, 128]
eye_center 0 [200, 96]
eye_center 1 [271, 108]
nilai panjang a71
nilai tengah a35.5
koordinat titik tengah a[235, 96]
centerihidung[228, 139]
tinggiHf43
panjang EG 55.7606
koordinat x [200, 108]
panjang u [200, -12]
panjang u -12
    
```

```
panjang xF 71
panjang EF 72.0069
panjang v 55.7606
nilai o 16.2463
nilai h 53.3414
```

$$yaw = \frac{55,7606 - \frac{71}{2}}{71}$$

$$= -0,28536$$

$$pitch = \frac{53,3414}{24}$$

$$= 1,24050$$

$$Roll = \frac{-12}{71}$$

$$= -0,16901$$

4. Arah Kiri

Data citra menunjukkan arah pergerakan kepala ke kiri.



(a) Citra RGB

(b) Citra Gray

**Gambar 4.14 Citra Arah Kiri**

Diketahui nilai citra:

```
wajah [189, 125]
eye_center 0 [155, 111]
eye_center 1 [225, 94]
nilai panjang a70
nilai tengah a35
koordinat titik tengah a[190, 111]
centerhidung[199, 140]
tinggiHf29
panjang EG 45.4533
koordinat x [155, 94]
panjang u [155, 17]
panjang u 17
panjang xF 70
panjang EF 72.0347
panjang v 45.4533
nilai o 26.5814
nilai h 36.8704
```



$$yaw = \frac{45,4533 - \frac{70}{2}}{70}$$

$$= -0,14933$$

$$pitch = \frac{36,8704}{29}$$

$$= 1,27139$$

$$Roll = \frac{17}{70}$$

$$= 0,24286$$

### 4.3 Perancangan Pengujian

Perancangan pengujian akan menjelaskan bagaimana sistem akan diuji setelah dibuat. Pengujian yang dilakukan adalah pengujian deteksi warna kulit, mata dan hidung, pengujian akurasi masing-masing arah pergerakan kepala yang menunjukkan posisi kepala sesuai dengan keadaan nyata. Contoh tabel pengujian dapat dilihat pada Tabel 4.1, Tabel 4.2 dan Tabel 4.3.

**Tabel 4.1 Tabel Pengujian Deteksi Warna Kulit, Mata dan Hidung**

No	Citra	Deteksi				Keberhasilan
		Kulit	Mata Kanan	Mata Kiri	Hidung	
1						
2						
3						
4						
5						

**Tabel 4.2 Tabel Pengujian Analisis Nilai Batas Atas dan Batas Bawah**

No	Citra	Output Nilai Yaw, Pitch dan Roll	Range yaw: -1,42353 - 0,46262	Range pitch: 1,18538 - 1,34999	Range roll: -0,31429 - 0,28235	Keberhasilan
1						
2						
3						
4						
5						



Tabel 4.3 Pengujian Arah Pergerakan Kepala

No	Citra	Output Sistem	Kondisi Arah Kepala	Range pitch: 1,28056 - 1,34999	Keberhasilan
1					
2					
3					
4					
5					



## BAB 5 IMPLEMENTASI

Bab ini menguraikan proses implementasi dari dasar teori yang telah dipelajari sesuai analisis dan perancangan sistem.

### 5.1 Implementasi Proses

#### 5.1.1 Implementasi *Pre-processing*

Pada tahap implementasi *pre-processing* akan ditunjukkan beberapa *source code* antara lain *flip*, *blur* dan pengkoreksian nilai *contras* dan *brightness*.

Pre-processing	
1	<code>namedWindow("slideBar", 1);</code>
2	<code>int slideValueB= 50;</code>
3	<code>createTrackbar("Brightness","slideBar", &amp;slideValueB, 150);</code>
4	<code>int slideValueC= 50;</code>
5	<code>createTrackbar("Contras","slideBar", &amp;slideValueC, 150);</code>
6	<code>while (1) {</code>
7	<code>Mat imageAwal, image1;</code>
8	<code>cap &gt;&gt; imageAwal;</code>
9	<code>Mat image;</code>
10	<code>flip(imageAwal, image1, 1);</code>
11	<code>blur(image1, image1, Size(5, 5));</code>
12	<code>int Brightness = slideValueB - 50;</code>
13	<code>double Contras = slideValueC / 50.0;</code>
14	<code>image1.convertTo(image, -1, Contras, Brightness);</code>

**Kode Program 5.1 Implementasi *Pre-Processing***

Pada baris 1-5 digunakan untuk membuat *trackbar* yang bertujuan untuk memudahkan pengguna melakukan pengaturan pada nilai *contrast* dan *brightness*. Kemudian pada baris 10 dilakukan *flip* untuk membuat posisi gambar citra sesuai dengan keadaan nyata, Pada baris 11 dilakukan proses *blur* untuk mengurangi noise pada citra gambar. Baris ke 12-14 adalah proses untuk merubah nilai *contras* dan *brightness* sesuai dengan nilai yang *diinput* pada slide bar.

#### 5.1.2 Implementasi Deteksi Warna Kulit

Pada tahap ini dilakukan beberapa tahap antara lain perubahan pixel warna, penerapan aturan warna *HSV*, *YCbCr* kemudian pemisahan *channels* warna *HSV*, *YCbCr* dan penggabungan *channels RGB-H-CbCr*.

Deteksi warna kulit	
1	<code>Mat MaskFace(const Mat &amp;src) {</code>
2	<code>Mat dst = src.clone();</code>
3	<code>Vec3b hitam =Vec3b::all(0);</code>



```

4 Mat src_ycrCb, src_hsv;
5 cvtColor(src, src_ycrCb, CV_BGR2YCrCb);
6 src.convertTo(src_hsv, CV_32FC3);
7 cvtColor(src_hsv, src_hsv, CV_BGR2HSV);
8 normalize(src_hsv, src_hsv, 0.0, 255.0, NORM_MINMAX, CV_32FC3);
9 for (int i = 0; i < src.rows; i++) {
10     for (int j = 0; j < src.cols; j++) {
11         Vec3b pix_bgr = src.ptr<Vec3b> (i)[j];
12         int B = pix_bgr.val[0];
13         int G = pix_bgr.val[1];
14         int R = pix_bgr.val[2];
15         bool a = R1(R, G, B);
16         Vec3b pix_ycrCb = src_ycrCb.ptr<Vec3b> (i)[j];
17         int Y = pix_ycrCb.val[0];
18         int Cr = pix_ycrCb.val[1];
19         int Cb = pix_ycrCb.val[2];
20         bool b = R2(Y, Cr, Cb);
21         Vec3f pix_hsv = src_hsv.ptr<Vec3f> (i)[j];
22         float H = pix_hsv.val[0];
23         float S = pix_hsv.val[1];
24         float V = pix_hsv.val[2];
25         bool c = R3(H, S, V);
26         if (!(a && b && c))
27             dst.ptr<Vec3b> (i)[j] = hitam;
28     }
29 }
30 return dst;
31 }

```

**Kode Program 5.2 Implementasi Deteksi Warna Kulit**

Pada Kode Program 5.2 merupakan implementasi dari proses pemisahan dan penggabungan *channels* warna. Baris 11-15 adalah proses pemisahan warna *RGB*, baris 16-20 adalah proses pemisahan warna *YCbCr*, baris 21-25 adalah proses pemisahan warna *HSV*. Untuk mendapatkan hasil warna *RGB-H-CbCr* maka di lakukan proses penggabungan warna yang di implementasikan pada baris 26-27. Baris 30 merupakan pengembalian hasil nilai penggabungan persamaan menggunakan variable *dst*.

Aturan nilai return R1, R2, R3

```

1     bool R1(int R, int G, int B) {
2         bool nilai1 = (R > 95) && (G > 40) && (B > 20) && ((max(R,
3         max(G, B)) - min(R, min(G, B))) > 15) && (abs(R - G) > 15) && (R >
4         G) && (R > B);

```

```

5      bool nilai2 = (R > 220) && (G > 210) && (B > 170) && (abs(R
6      - G) <= 15) && (R > B) && (G > B);
7
8      return (nilai1 || nilai2);
9
10     }
11     bool R2(float Y, float Cr, float Cb) {
12         bool nilai3 = Cr <= 1.5862 * Cb + 20;
13         bool nilai4 = Cr >= 0.3448 * Cb + 76.2069;
14         bool nilai5 = Cr >= -4.5652 * Cb + 234.5652;
15         bool nilai6 = Cr <= -1.15 * Cb + 301.75;
16         bool nilai7 = Cr <= -2.2857 * Cb + 432.85;
17         return nilai3 && nilai4 && nilai5 && nilai6 && nilai7;
18     }
19
20     bool R3(float H, float S, float V) {
21         return (H < 25) || (H > 230);
22     }

```

**Kode Program 5.3 Implementasi Penerapan aturan warna RGB, HSV dan YCbCr**

Pada baris 1-7 merupakan implementasi penerapan aturan warna untuk gambar citra RGB, baris 8-19 merupakan implementasi penerapan aturan warna untuk gambar citra *YCbCr*, kemudian pada baris 16-18 adalah implementasi penerapan aturan warna untuk gambar citra *HSV*.

**5.1.3 Implementasi Deteksi Bagian Wajah**

Pada tahap ini dilakukan implementasi untuk mengenali bagian dari wajah antara lain mata dan hidung. Deteksi bagian wajah yang menggunakan *library OpenCv (Haar Cascade)*.

Deteksi Bagian Wajah	
1	<code>void mata(Mat frame) {</code>
2	<code>    Mat frame_gray;</code>
3	<code>    cvtColor(frame, frame_gray, COLOR_BGR2GRAY);</code>
4	<code>    equalizeHist(frame_gray, frame_gray);</code>
5	<code>    face_cascade.detectMultiScale(frame_gray, faces, 1.1, 2,</code>
6	<code>0, Size(80, 80));</code>
7	<code>    for (size_t I = 0; I &lt; faces.size(); i++) {</code>
8	<code>        Mat faceROI = frame_gray(faces[i]);</code>
9	<code>        std::vector&lt;Rect&gt; eyes;</code>
10	<code>        eyes_cascade.detectMultiScale(faceROI, eyes, 1.1, 2,0</code>
11	<code> CASCADE_SCALE_IMAGE, Size(30, 30));</code>
12	<code>        if (eyes.size() == 2) {</code>
13	<code>            Point center(faces[i].x + faces[i].width / 2,</code>
14	<code>faces[i].y + faces[i].height / 2);</code>
15	<code>            ellipse(frame_gray, center,</code>



```

16         Size(faces[i].width / 2, faces[i].height / 2), 0,
17         0, 360,
18         Scalar(255, 0, 0), 2, 8, 0);
19         circle(frame_gray, center, 5, Scalar(255, 255,
20         255), 2, 8, 0);
21         //Deteksi Mata
22         for (size_t j = 0; j < eyes.size(); j++) {
23             Point eye_center(faces[i].x + eyes[j].x +
24             eyes[j].width / 2, faces[i].y + eyes[j].y + eyes[j].height / 2);
25             int radius = cvRound((eyes[j].width +
26             eyes[j].height) * 0.25);
27             circle(frame_gray, eye_center, radius,
28             Scalar(255, 0, 255), 2, 8, 0);
29             circle(frame_gray, eye_center, 5,
30             Scalar(255, 255, 255), 2, 8, 0);
31             tempArray[j] = eye_center;
32         }
33         //Deteksi Hidung
34         vector<Rect> nosei;
35         nose_cascade.detectMultiScale(faceROI, nosei, 1.1, 3, 0 |
36         CASCADE_SCALE_IMAGE, Size(30, 30));
37         for (size_t k = 0; k < nosei.size(); k++) {
38             Point centerSegitiga(faces[i].x + nosei[k].x
39             + nosei[k].width / 2,
40             faces[i].y + nosei[k].y + nosei[k].height /
41             2);
42             circle(frame_gray, centerSegitiga, 5,
43             Scalar(255, 255, 255), 2, 8, 0);
44             line(frame_gray, tempArray[0],
45             centerSegitiga,
46             Scalar(255, 255, 255), 2, 8, 0);
47         }
48     }

```

#### Kode Program 5.4 Implementasi Deteksi Bagian Wajah

Kode program 5.4 merupakan implementasi deteksi bagian mata. Baris 2 merupakan deklarasi variable. Baris 3 mengubah citra warna *RGB* menjadi *gray*. Baris 4 merupakan proses pemerataan nilai histogram menggunakan fungsi *equalizeHist*. Baris 5-20 merupakan proses untuk mendeteksi wajah dengan memanfaatkan fungsi *face\_cascade.detectMultiScale*. Sebelum mengenali wajah maka diproses deteksi mata dan hidung sebagai syarat deteksi wajah. Baris 21-32 merupakan bagian proses deteksi mata dengan memanfaatkan fungsi

eyes\_cascade.detectMultiScale. Baris 33-48 merupakan implementasi untuk mengenali hidung dengan memanfaatkan fungsi nose\_cascade.detectMultiScale.

#### 5.1.4 Implementasi *HeadTracking*

Pada tahap ini dilakukan implementasi penerapan metode *triangle geometry* sebagai tahap untuk mendeteksi arah pergerakan posisi kepala secara *realtime* dan mengklasifikasikan masing-masing arah pergerakan posisi kepala.

```
Perhitungan nilai persamaan triangle geometry
1      Point x(tempArray[0].x, tempArray[1].y); //titik x
2      (koordinat)
3      Point u(tempArray[0].x, tempArray[0].y - tempArray[1].y);
4      panjangU = u.y;
5      circle(frame_gray, x, 5, Scalar(255, 255, 255), 2, 8, 0);
6      line(frame_gray, tempArray[0], x, Scalar(255, 255, 255), 2,
7      8, 0);
8      line(frame_gray, tempArray[1], x, Scalar(255, 255, 255), 2,
9      8, 0);
10     cout << "\nkoordinat x " << x;
11     cout << "\npanjang u " << u;
12     cout << "\npanjang u " << u.y; //panjang u
13     double xF = tempArray[1].x - x.x;
14     cout << "\npanjang xF " << xF;
15     double EF = sqrt(pow(xF, 2) + pow(u.y, 2));
16     cout << "\npanjang EF " << EF;
17     double EH = (EF * EG) / EF;
18     cout << "\npanjang v " << EH;
19     double o = EF - EH;
20     cout << "\nnilai o " << o;
21     double h = sqrt(pow(EG, 2) - pow(o, 2));
22     cout << "\nnilai h " << h;
23     //Persamaan Metode Triangle Geometry
24     double yaw = (EH - (panjangA / 2)) / panjangA;
25     double pitch = h / tinggi;
26     double roll = panjangU / panjangA;
27     cout << "\n==perbandingan seitiga==";
28     cout << "\nxF/HG " << xF << "/" << tinggi;
29     cout << "\nxE/HF " << u.y << "/" << o;
30     cout << "\nEF/FG " << EF << "/" << h;
```

**Kode Program 5.5 Implementasi Perhitungan Nilai Variable Persamaan Metode**

Kode Program 5.5 adalah proses perhitungan matematika dasar untuk mendapatkan nilai variable yang digunakan dan menghitung nilai variabel sesuai

dengan persamaan *triangle geometry*. Baris 1-22 merupakan implementasi untuk perhitungan nilai setiap variabel. Baris 23-30 merupakan perhitungan variable sesuai dengan persamaan metode *triangle geometry*.

Klasifikasi range arah pergerakan kepala	
1	if (tempArray[0].y == tempArray[1].y) { //titik E.y = titik F.y
2	cout << "\nsejajar";
3	cout << "\n=====nilai=====";
4	cout << "\nYAW      : " << 0;
5	cout << "\nPITCHH   : " << 0;
6	cout << "\nROLL     : " << 0;
7	cout << "\n\n";
8	} else if ((pitch > 1.18538) && (pitch < 1.23138)) {
9	cout << "\nKepala ke bawah";
10	cout << "\n===== HASIL =====";
11	printf("\nYAW %.5f", yaw);
12	printf("\nPITCH %.5f", pitch);
13	printf("\nROLL %.5f", roll);
14	} else if ((pitch > 1.28056) && (pitch < 1.34999)) {
15	cout << "\nKepala ke atas";
16	cout << "\n===== HASIL =====";
17	printf("\nYAW %.5f", yaw);
18	printf("\nPITCH %.5f", pitch);
19	printf("\nROLL %.5f", roll);
20	} else if ((roll < -0.10127) && (roll > -0.31429)){
21	cout << "\nKepala ke kanan";
22	cout << "\n===== HASIL =====";
23	printf("\nYAW %.5f", yaw);
24	printf("\nPITCH %.5f", pitch);
25	printf("\nROLL %.5f", roll);
26	} else if ((roll > 0.05000) && (roll < 0.28235)) {
27	cout << "\nKepala ke kiri";
28	cout << "\n===== HASIL =====";
29	printf("\nYAW %.5f", yaw);
30	printf("\nPITCH %.5f", pitch);
31	printf("\nROLL %.5f", roll);
32	}

#### Kode Program 5.6 Implementasi Klasifikasi Range Arah Pergerakan Kepala

Baris 1-7 merupakan kondisi ketika kepala tidak mengalami perubahan posisi. Baris 8-13 merupakan kondisi ketika kepala ke bawah. Baris 14-19 merupakan kondisi ketika kepala ke atas. Baris 20-25 merupakan kondisi ketika kepala ke kanan. Baris 26-32 merupakan kondisi ketika kepala ke kiri.

### 5.1.5 Implementasi Klasifikasi Nilai Batas Atas dan Batas Bawah *Range* Arah

Pada tahap ini dilakukan pengelompokan *range* arah berdasarkan nilai *yaw*, *pitch*, *roll* dari pengambilan data uji sistem. Klasifikasi *range* arah berdasarkan arahnya dapat dilihat pada Tabel 5.1, 5.2, 5.3, dan 5.4 dengan keterangan bahwa (A) adalah Batas Atas dan (B) adalah Batas Bawah.

**Tabel 5.1 Klasifikasi Arah Atas**

Atas		
<i>Yaw</i>	<i>Pitch</i>	<i>Roll</i>
0,09473 (A)	1,34514	-0,66670
-1,17305	1,28056 (B)	-0,07692
-1,08077	1,34918	-0,07955 (A)
-1,11825	1,33386	-0,05682
0,11825	1,33610	-0,03409
-1,10093	1,34708	-0,02381
-1,12598 (B)	1,33008	-0,05195
-1,11202	1,34093	-0,01176 (B)
-1,1508	1,31295	-0,08333
-1,1653	1,30646	-0,07317
-1,1477	1,31734	-0,58823
-1,0966	1,34106	-0,08139
-1,0989	1,34643	-0,04395
-1,0944	1,34999 (A)	-0,35714
0,17376	1,30279	-0,06451

Dari hasil klasifikasi arah atas pada Tabel 5.1 didapatkan nilai batas atas dan batas bawah untuk masing-masing kondisi, antara lain:

Batas Bawah

*Yaw* : -1,12598

*Pitch* : 1,28056

*Roll* : -0,01176

Batas Atas

*Yaw* : 0,09473

*Pitch* : 1,34999

*Roll* : -0,07955

**Tabel 5.2 Klasifikasi Arah Bawah**

Bawah		
<i>Yaw</i>	<i>Pitch</i>	<i>Roll</i>
0,35627	1,21421	-0,02439
-1,36882	1,20875	-0,01316
0,34056	1,23138 (A)	0,01351
-1,38154	1,20255	0,08219
-1,36367	1,20992	0,09859 (A)
-1,42323 (B)	1,18538 (B)	0,02985
0,35146	1,20603	0,04054
-1,37896	1,21629	0,01205
0,35627	1,20430	0,03659
0,36129	1,21418	-0,01299
0,39577 (A)	1,21208	-0,02073 (B)
-1,36129	1,19698	-0,01299
-1,38070	1,21208	0,02500
0,36746	1,20935	0,01266
0,35834	1,21262	-0,08139

Dari hasil klasifikasi arah bawah pada Tabel 5.2 didapatkan nilai batas atas dan batas bawah untuk masing-masing kondisi, antara lain:

Batas Bawah	Batas Atas
<i>Yaw</i> : -1,42323	<i>Yaw</i> : 0,39577
<i>Pitch</i> : 1,18538	<i>Pitch</i> : 1, 23138
<i>Roll</i> : -0,02073	<i>Roll</i> : 0,09859

**Tabel 5.3 Klasifikasi Arah Kanan**

Kanan		
<i>Yaw</i>	<i>Pitch</i>	<i>Roll</i>
0,37348	1,20339	-0,17568



0,36217	1,20459	-0,23810
<b>0,32660 (B)</b>	<b>1,22616 (A)</b>	<b>-0,10127(A)</b>
0,38040	1,19889	-0,23188
0,33732	1,22078	-0,11940
0,38679	1,19600	-0,22535
0,35670	1,20706	-0,23188
0,39118	1,19664	-0,16393
0,43131	1,17593	<b>-0,31429 (B)</b>
0,37364	1,19934	-0,25373
0,41443	1,18247	-0,29688
0,43039	1,17725	-0,29231
0,35894	1,20774	-0,20635
0,40459	1,18692	-0,27695
<b>0,46262 (A)</b>	<b>1,16586(B)</b>	-0,30645

Dari hasil klasifikasi arah kanan pada Tabel 5.3 didapatkan nilai batas atas dan batas bawah untuk masing-masing kondisi, antara lain:

Batas Bawah

*Yaw* : 0,32660

*Pitch* : 1,16586

*Roll* : -0,31429

Batas Atas

*Yaw* : 0,46262

*Pitch* : 1,22616

*Roll* : -0,10127

**Tabel 5.4 Klasifikasi Arah Kiri**

Kiri		
<i>Yaw</i>	<i>Pitch</i>	<i>Roll</i>
-1,37618	1,20034	0,21951
-1,37190	1,20226	0,21429
-1,41132	1,18526	0,26190
<b>-1,42353 (B)</b>	<b>1,18004 (B)</b>	<b>0,28235 (A)</b>
-1,38168	1,20040	0,16667
-1,39862	1,19533	0,08000
<b>0,40139 (A)</b>	1,19345	0,12500

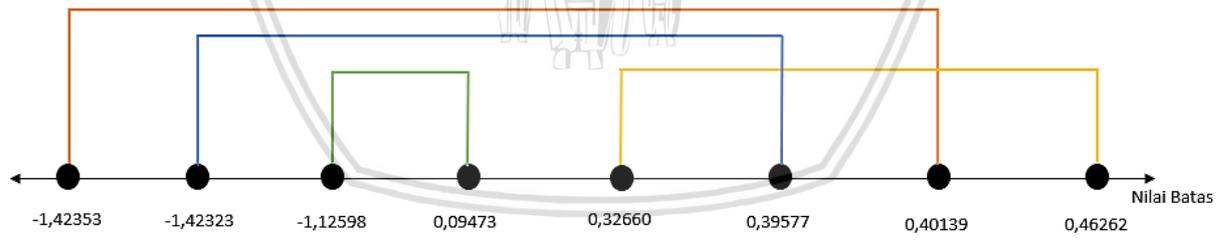


-1,36325	1,21037	0,08642
-1,40702	1,19193	0,06757
-1,34001	1,22133	0,05000 (B)
-1,38257	1,20220	0,07576
-1,32271	1,22857	0,08000
0,187201	1,25574	0,27142
0,191906	1,26235	0,24637
0,207107	1,26677 (A)	0,20833

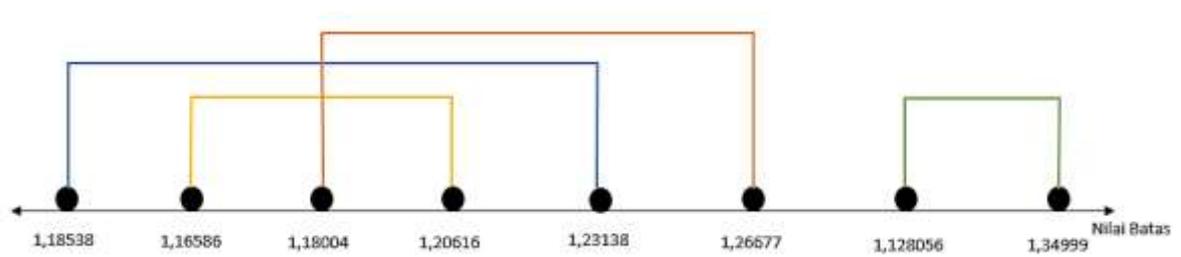
Dari hasil klasifikasi arah kiri pada Tabel 5.4 didapatkan nilai batas atas dan batas bawah untuk masing-masing kondisi, antara lain:

<b>Batas Bawah</b>	<b>Batas Atas</b>
<i>Yaw</i> : -1,42353	<i>Yaw</i> : 0,40139
<i>Pitch</i> : 1,18004	<i>Pitch</i> : 1,26677
<i>Roll</i> : 0,05000	<i>Roll</i> : 0,28235

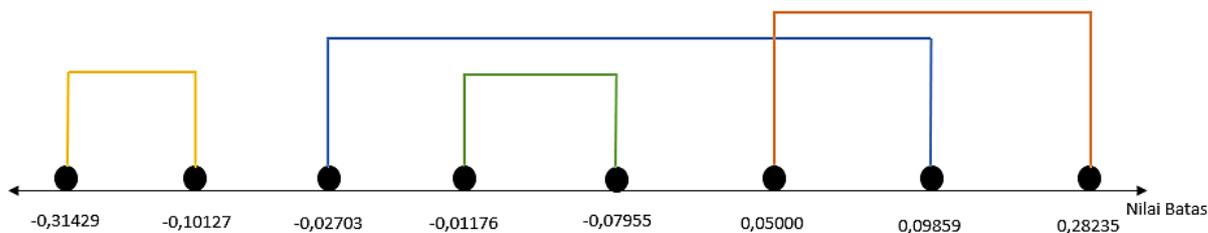
Masing-masing tabel klasifikasi arah diatas diolah menjadi *range* arah berdasarkan nilai *yaw*, *pitch* dan *roll*. Ilustrasi untuk klasifikasi *range* arah berdasarkan nilai *yaw*, *pitch* dan *roll* dapat dilihat pada Gambar 5.1, Gambar 5.2, Gambar 5.3 dan Gambar 5.4.



Gambar 5.1 Ilustrasi Klasifikasi *range* arah berdasarkan nilai *Yaw*



Gambar 5.2 Ilustrasi Klasifikasi *range* arah berdasarkan nilai *Pitch*



**Gambar 5.3 Ilustrasi Klasifikasi *range* arah berdasarkan nilai *Roll***

dimana:

- : Arah Kanan
- : Arah Atas
- : Arah Kiri
- : Arah Bawah

Berdasarkan hasil ilustrasi *range* arah diatas, maka dapat di tentukan nilai *yaw*, *pitch* dan *roll* yang paling mempengaruhi pada setiap arah gerakan kepala. Nilai yang mempengaruhi untuk setiap arah adalah sebgai berikut:

**Arah Atas dan Bawah: *Pitch***      **Arah Kiri dan Kanan: *Roll***

Sedangkan untuk nilai *yaw*, memiliki nilai yang berdekatan pada setiap gerakan. Ini terjadi dikarenakan gerakan arah *yaw* tidak sempurna. Ketidak sempurnaan gerakan *yaw* dikarenakan harus tetap mempertahankan objek kedua bola mata terlihat pada kamera untuk dapat mendeteksi wajah dan dilanjutkan untuk proses deteksi arah pergerakan kepala. Hal ini menyebabkan nilai *yaw* yang dihasilkan tidak dapat dijadikan acuan untuk pergerakan arah kanan dan kiri. Sehingga dalam kondisi ini, nilai *yaw* tidak memiliki peran yang signifikan pada masing - masing arah gerakan kepala.

## BAB 6 PENGUJIAN

Bab ini berisi pengujian yang dilakukan, antara lain pengujian deteksi warna kulit, mata dan hidung, pengujian analisis batas atas dan batas bawah nilai *yaw*, *pitch*, *roll* dan pengujian arah arah pergerakan kepala menggunakan metode *triangle geometry*. Data uji sebanyak 10 orang dengan warna kulit, kondisi mata dan tipe wajah yang berbeda, diambil pada kondisi tempat dan waktu yang berbeda. Pengujian ini menggunakan sistem eliminasi yang diukur dari tingkat keberhasilan citra. Pengambilan citra menggunakan video dengan durasi 27-50 detik menggunakan kamera logitech c920 HD 1080p dengan kondisi 20fps (20 *frame/ ms*). Kemudian dilakukan pengambilan 20 *frame* dari masing - masing data uji dengan cara membagi total *frame* dengan 20.

### 6.1 Pengujian Deteksi Warna Kulit, Mata dan Hidung

Pada tahap pengujian ini dilakukan deteksi warna kulit yang bertujuan untuk mempermudah deteksi mata dan hidung. Selanjutnya, dilakukan proses deteksi mata dan hidung yang digunakan sebagai titik penentu klasifikasi pergerakan arah kepala. Jika mata terdeteksi, maka gambar citra akan ditandai dengan lingkaran kecil dan apabila hidung terdeteksi, maka akan ditandai dengan kotak. Pada tabel pengujian jika bagian wajah terdeteksi akan diberi kata "Ya" dan "Tidak" ketika bagian wajah tidak terdeteksi. Angka keberhasilan tiap *frame* dinyatakan dalam *range* 0-4 yang mewakili keberhasilan deteksi kulit, mata kanan, mata kiri dan hidung. Keberhasilan sempurna dinyatakan dengan notasi angka 4.

**Tabel 6.1 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-1**

No	Citra	Deteksi				Keberhasilan
		Kulit	Mata Kanan	Mata Kiri	Hidung	
1		Ya	Ya	Ya	Tidak	3
2		Ya	Tidak	Tidak	Tidak	1
3		Ya	Ya	Ya	Ya	4
4		Ya	Tidak	Tidak	Ya	2

Tabel 6.1 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-1 (Lanjutan)

5		Ya	Tidak	Tidak	Ya	2
6		Ya	Tidak	Tidak	Ya	2
7		Ya	Tidak	Ya	Tidak	2
8		Ya	Ya	Ya	Tidak	3
9		Ya	Tidak	Ya	Tidak	2
10		Ya	Ya	Ya	Tidak	3
11		Ya	Tidak	Tidak	Ya	2
12		Ya	Ya	Ya	Ya	4
13		Ya	Tidak	Tidak	Ya	2
14		Ya	Ya	Ya	Ya	4
15		Ya	Ya	Ya	Ya	4

**Tabel 6.1 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-1 (Lanjutan)**

16		Ya	Tidak	Ya	Tidak	2
17		Ya	Ya	Ya	Ya	4
18		Ya	Tidak	Tidak	Tidak	1
19		Ya	Ya	Ya	Ya	4
20		Ya	Tidak	Tidak	Tidak	1

Pengambilan data uji Tabel 6.1 di tempat *indoor* (kamar) pada sore hari dengan kondisi intensitas cahaya 251 lx. Pada data uji ke-1 menggunakan koreksi nilai *brightness* 101 dan *contrast* 50. Pengujian data uji ke-1 mencapai tingkat keberhasilan 65%.

**Tabel 6.2 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-2**

No	Citra	Deteksi				Keberhasilan
		Kulit	Mata Kanan	Mata Kiri	Hidung	
1		Ya	Tidak	Tidak	Ya	2
2		Ya	Tidak	Tidak	Tidak	1
3		Ya	Ya	Ya	Ya	4

**Tabel 6.3 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-2  
(Lanjutan)**

4		Ya	Ya	Ya	Ya	4
5		Ya	Tidak	Tidak	Ya	2
6		Ya	Ya	Ya	Ya	4
7		Ya	Ya	Ya	Ya	4
8		Ya	Ya	Ya	Ya	4
9		Ya	Tidak	Tidak	Ya	2
10		Ya	Ya	Ya	Ya	4
11		Ya	Ya	Ya	Ya	4
12		Ya	Ya	Ya	Ya	4
13		Ya	Tidak	Tidak	Ya	2
14		Ya	Ya	Ya	Ya	4

**Tabel 6.2 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-2 (Lanjutan)**

15		Ya	Ya	Ya	Ya	4
16		Ya	Tidak	Ya	Tidak	2
17		Ya	Ya	Ya	Tidak	3
18		Ya	Ya	Ya	Ya	4
19		Ya	Tidak	Tidak	Tidak	1
20		Ya	Ya	Ya	Ya	4

Pengambilan data uji Tabel 6.2 di tempat *outdoor* (gazebo kampus) pada siang hari dengan kondisi intensitas cahaya 417 lx -584 lx. Pada data uji ke-2 menggunakan koreksi nilai *brightness* 133 dan *contrast* 53. Pengujian data uji ke-2 mencapai tingkat keberhasilan 76,25%.

**Tabel 6.4 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-3**

No	Citra	Deteksi				Keberhasilan
		Kulit	Mata Kanan	Mata Kiri	Hidung	
1		Ya	Ya	Ya	Ya	4
2		Ya	Tidak	Tidak	Tidak	1
3		Ya	Ya	Ya	Ya	4

**Tabel 6.3 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-3 (Lanjutan)**

4		Ya	Tidak	Tidak	Tidak	1
5		Ya	Tidak	Tidak	Ya	2
6		Ya	Ya	Ya	Tidak	3
7		Ya	Tidak	Ya	Tidak	2
8		Ya	Tidak	Tidak	Tidak	1
9		Ya	Tidak	Tidak	Ya	2
10		Ya	Ya	Ya	Ya	4
11		Ya	Tidak	Tidak	Ya	2
12		Ya	Tidak	Tidak	Tidak	1
13		Ya	Tidak	Tidak	Ya	2
14		Ya	Ya	Ya	Ya	4

**Tabel 6.3 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-3 (Lanjutan)**

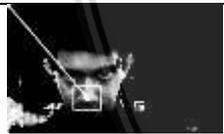
15		Ya	Ya	Ya	Ya	4
16		Ya	Tidak	Tidak	Tidak	1
17		Ya	Tidak	Tidak	Tidak	1
18		Ya	Ya	Ya	Ya	4
19		Ya	Tidak	Tidak	Tidak	1
20		Ya	Ya	Ya	Ya	4

Pengambilan data uji Tabel 6.3 di tempat *outdoor* (gazebo kampus) pada siang hari dengan kondisi intensitas cahaya 417 lx – 584 lx. Pada data uji ke-3 menggunakan koreksi nilai *brightness* 130 dan *contrast* 73. Pengujian data uji ke-3 mencapai tingkat keberhasilan 60%.

**Tabel 6.5 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-4**

No	Citra	Deteksi				Keberhasilan
		Kulit	Mata Kanan	Mata Kiri	Hidung	
1		Ya	Ya	Ya	Ya	4
2		Ya	Tidak	Tidak	Tidak	1
3		Ya	Ya	Ya	Ya	4

**Tabel 6.6 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-4 (Lanjutan)**

4		Ya	Ya	Ya	Ya	4
5		Ya	Ya	Tidak	Tidak	2
6		Ya	Ya	Ya	Ya	4
7		Ya	Ya	Ya	Tidak	3
8		Ya	Tidak	Tidak	Ya	2
9		Ya	Ya	Ya	Ya	4
10		Ya	Tidak	Tidak	Ya	2
11		Ya	Tidak	Tidak	Ya	2
12		Ya	Ya	Ya	Ya	4
13		Ya	Ya	Ya	Tidak	3
14		Ya	Ya	Ya	Ya	4

**Tabel 6.7 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-4 (Lanjutan)**

15		Ya	Tidak	Tidak	Tidak	1
16		Ya	Ya	Ya	Ya	4
17		Ya	Tidak	Tidak	Tidak	1
18		Ya	Ya	Ya	Ya	4
19		Ya	Tidak	Tidak	Tidak	1
20		Ya	Ya	Ya	Ya	4

Pengambilan data uji Tabel 6.4 di tempat *outdoor* (gazebo kampus) pada siang hari dengan kondisi intensitas cahaya 417 lx – 584 lx. Pada data uji ke-4 menggunakan koreksi nilai *brightness* 100 dan *contrast* 62. Pengujian data uji ke-4 mencapai tingkat keberhasilan 72,5%.

**Tabel 6.8 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-5**

No	Citra	Deteksi				Keberhasilan
		Kulit	Mata Kanan	Mata Kiri	Hidung	
1		Ya	Tidak	Tidak	Ya	2
2		Ya	Tidak	Tidak	Ya	2

**Tabel 6.9 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-5  
(Lanjutan)**

3		Ya	Ya	Ya	Ya	4
4		Ya	Tidak	Tidak	Tidak	1
5		Ya	Ya	Ya	Ya	4
6		Ya	Tidak	Tidak	Tidak	1
7		Ya	Tidak	Tidak	Tidak	1
8		Ya	Ya	Ya	Ya	4
9		Ya	Tidak	Tidak	Tidak	1
10		Ya	Ya	Ya	Ya	4
11		Ya	Tidak	Tidak	Tidak	1
12		Ya	Ya	Ya	Tidak	3
13		Ya	Ya	Ya	Ya	4

**Tabel 6.10 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-5 (Lanjutan)**

14		Ya	Ya	Ya	Ya	4
15		Ya	Ya	Ya	Ya	4
16		Ya	Tidak	Tidak	Tidak	1
17		Ya	Ya	Ya	Ya	4
18		Ya	Ya	Ya	Ya	4
19		Ya	Ya	Ya	Ya	4
20		Ya	Ya	Ya	Tidak	3

Pengambilan data uji Tabel 6.5 di tempat *outdoor* (gazebo kampus) pada siang hari dengan kondisi intensitas cahaya 417-584. Pada data uji ke-5 menggunakan koreksi nilai *brightness* 148 dan *contrast* 59. Pengujian data uji ke-5 mencapai tingkat keberhasilan 70%.

**Tabel 6.11 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-6**

No	Citra	Deteksi				Keberhasilan
		Kulit	Mata Kanan	Mata Kiri	Hidung	
1		Ya	Ya	Ya	Ya	4
2		Ya	Tidak	Tidak	Tidak	1

**Tabel 6.12 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-6  
(Lanjutan)**

3		Ya	Ya	Ya	Ya	4
4		Ya	Ya	Ya	Ya	4
5		Ya	Tidak	Tidak	Ya	2
6		Ya	Tidak	Tidak	Ya	2
7		Ya	Tidak	Tidak	Ya	2
8		Ya	Ya	Ya	Ya	4
9		Ya	Tidak	Tidak	Ya	2
10		Ya	Ya	Ya	Ya	4
11		Ya	Tidak	Tidak	Ya	2
12		Ya	Ya	Ya	Ya	4
13		Ya	Ya	Ya	Ya	4

**Tabel 6.13 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-6 (Lanjutan)**

14		Ya	Ya	Ya	Ya	4
15		Ya	Tidak	Tidak	Tidak	1
16		Ya	Ya	Ya	Ya	4
17		Ya	Tidak	Tidak	Tidak	1
18		Ya	Tidak	Tidak	Tidak	1
19		Ya	Tidak	Tidak	Tidak	1
20		Ya	Ya	Ya	Ya	4

Pengambilan data uji Tabel 6.6 di tempat *outdoor* (gazebo kampus) pada siang hari dengan kondisi intensitas cahaya 417 lx – 584 lx. Pada data uji ke-6 menggunakan koreksi nilai *brightness* 130 dan *contrast* 77. Pengujian data uji ke-6 mencapai tingkat keberhasilan 68,75%.

**Tabel 6.14 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-7**

No	Citra	Deteksi				Keberhasilan
		Kulit	Mata Kanan	Mata Kiri	Hidung	
1		Ya	Tidak	Tidak	Tidak	1
2		Ya	Ya	Ya	Ya	4

**Tabel 6.15 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-7 (Lanjutan)**

3		Ya	Ya	Tidak	Ya	3
4		Ya	Tidak	Tidak	Tidak	1
5		Ya	Ya	Ya	Ya	4
6		Ya	Tidak	Tidak	Ya	2
7		Ya	Ya	Tidak	Ya	3
8		Ya	Ya	Ya	Ya	4
9		Ya	Tidak	Tidak	Ya	2
10		Ya	Tidak	Ya	Tidak	2
11		Ya	Ya	Ya	Ya	4
12		Ya	Tidak	Tidak	Tidak	1
13		Ya	Ya	Ya	Ya	4

**Tabel 6.16 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-7 (Lanjutan)**

14		Ya	Tidak	Tidak	Tidak	1
15		Ya	Tidak	Tidak	Ya	2
16		Ya	Tidak	Tidak	Tidak	1
17		Ya	Ya	Ya	Ya	4
18		Ya	Ya	Ya	Ya	4
19		Ya	Tidak	Tidak	Tidak	1
20		Ya	Ya	Ya	Ya	4

Pengambilan data uji Tabel 6.7 di tempat *outdoor* (gazebo kampus) pada siang hari dengan kondisi intensitas cahaya 417 lx – 584 lx. Pada data uji ke-7 menggunakan koreksi nilai *brightness* 142 dan *contrast* 128. Pengujian data uji ke-7 mencapai tingkat keberhasilan 58,75%.

**Tabel 6.17 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-8**

No	Citra	Deteksi				Keberhasilan
		Kulit	Mata Kanan	Mata Kiri	Hidung	
1		Ya	Tidak	Tidak	Ya	2

**Tabel 6.188 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-8  
(Lanjutan)**

2		Ya	Tidak	Tidak	Ya	2
3		Ya	Ya	Ya	Ya	4
4		Ya	Ya	Ya	Ya	4
5		Ya	Tidak	Tidak	Ya	2
6		Ya	Ya	Ya	Ya	4
7		Ya	Tidak	Tidak	Ya	2
8		Ya	Ya	Ya	Ya	4
9		Ya	Tidak	Tidak	Tidak	1
10		Ya	Ya	Ya	Ya	4
11		Ya	Tidak	Tidak	Ya	2
12		Ya	Tidak	Tidak	Ya	2

13		Ya	Ya	Ya	Ya	4
14		Ya	Ya	Ya	Ya	4
15		Ya	Ya	Ya	Ya	4
16		Ya	Ya	Ya	Ya	4
17		Ya	Tidak	Tidak	Tidak	1
18		Ya	Ya	Ya	Ya	4
19		Ya	Tidak	Tidak	Tidak	1
20		Ya	Tidak	Tidak	Ya	2

Pengambilan data uji Tabel 6.8 di tempat *outdoor* (cafe) pada malam hari dengan kondisi intensitas cahaya 907 lx -1200 lx. Pada data uji ke-8 menggunakan koreksi nilai *brightness* 50 dan *contrast* 38. Pengujian data uji ke-8 mencapai tingkat keberhasilan 71,25%.

**Tabel 6.19 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-9**

No	Citra	Deteksi				Keberhasilan
		Kulit	Mata Kanan	Mata Kiri	Hidung	
1		Ya	Tidak	Tidak	Tidak	1

**Tabel 6.20 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-9 (Lanjutan)**

2		Ya	Tidak	Tidak	Ya	2
3		Ya	Ya	Ya	Ya	4
4		Ya	Tidak	Tidak	Tidak	1
5		Ya	Tidak	Tidak	Ya	2
6		Ya	Tidak	Tidak	Tidak	1
7		Ya	Ya	Ya	Tidak	3
8		Ya	Ya	Ya	Ya	4
9		Ya	Tidak	Tidak	Tidak	1
10		Ya	Ya	Ya	Ya	4
11		Ya	Tidak	Tidak	Tidak	1
12		Ya	Tidak	Tidak	Tidak	1

13		Ya	Tidak	Tidak	Ya	2
14		Ya	Tidak	Tidak	Tidak	1
15		Ya	Ya	Ya	Ya	4
16		Ya	Ya	Ya	Ya	4
17		Ya	Tidak	Tidak	Tidak	1
18		Ya	Ya	Ya	Ya	4
19		Ya	Ya	Ya	Ya	4
20		Ya	Ya	Ya	Ya	4

Pengambilan data uji Tabel 6.9 di tempat *outdoor* (cafe) pada malam hari dengan kondisi intensitas cahaya 907 lx – 1200 lx. Pada data uji ke-9 menggunakan koreksi nilai *brightness* 50 dan *contrast* 38. Pengujian data uji ke-9 mencapai tingkat keberhasilan 61,25%.

Tabel 6.21 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-10

No	Citra	Deteksi				Keberhasilan
		Kulit	Mata Kanan	Mata Kiri	Hidung	
1		Ya	Ya	Ya	Ya	4
2		Ya	Tidak	Tidak	Ya	2
3		Ya	Tidak	Tidak	Ya	2
4		Ya	Ya	Ya	Ya	4
5		Ya	Ya	Ya	Ya	4
6		Ya	Ya	Ya	Ya	4
7		Ya	Ya	Ya	Ya	4
8		Ya	Tidak	Tidak	Ya	2
9		Ya	Tidak	Tidak	Ya	2
10		Ya	Ya	Ya	Ya	4
11		Ya	Ya	Ya	Ya	4



**Tabel 6.22 Pengujian Deteksi Warna Kulit, Mata dan Hidung Data Uji Ke-10 (Lanjutan)**

12		Ya	Tidak	Tidak	Ya	2
13		Ya	Ya	Ya	Ya	4
14		Ya	Tidak	Tidak	Ya	2
15		Ya	Ya	Ya	Ya	4
16		Ya	Ya	Ya	Ya	4
17		Ya	Ya	Ya	Ya	4
18		Ya	Tidak	Tidak	Tidak	1
19		Ya	Ya	Ya	Ya	4
20		Ya	Tidak	Tidak	Ya	2

Pengambilan data uji Tabel 6.10 di tempat *indoor* (kamar) pada malam hari dengan kondisi intensitas cahaya 35 lx – 60 lx. Pada data uji ke-10 menggunakan koreksi nilai *brightness* 125 dan *contrast* 71. Pengujian data uji ke-10 mencapai tingkat keberhasilan 78,75%.

## 6.2 Pengujian Analisis Nilai Batas Atas dan Batas Bawah

Pada tahap ini adalah pengujian data citra dari pengujian sebelumnya yang mendapatkan angka keberhasilan 4. Pada pengujian ini, *output* citra tersebut akan diuji kesesuaiannya dengan batas atas dan batas bawah nilai *yaw*, *pitch* dan *roll*.

Jika *output* citra termasuk kedalam *range* batas atas dan bawah nilai *yaw*, *pitch* dan *roll*, maka akan diberi keterangan “sesuai” pada tabel pengujian dan “tidak sesuai” untuk *output* yang tidak termasuk dalam *range* batas atas dan batas bawah. Tingkat keberhasilan pada pengujian ini dinotasikan dengan angka 0-3 yang mewakili keberhasilan analisis nilai batas atas dan batas bawah dari *yaw*, *pitch* dan *roll*. Angka 3 merupakan angka sempurna keberhasilan. Notasi (-) merupakan notasi ketika *output* citra tidak mengeluarkan hasil perhitungan dari nilai *yaw*, *pitch* atau *roll*. *Output* berupa nan atau -nan merupakan arti dari “*not a number*”, dianggap *output* tidak mengeluarkan hasil perhitungan dari nilai *yaw*, *pitch* atau *roll*.

**Tabel 6.23 Pengujian Analisis Nilai Batas Atas dan Batas Bawah Data Uji Ke-1**

No	Citra	Output Nilai Yaw, Pitch dan Roll	Range yaw: -1,42353 - 0,46262	Range pitch: 1,18538 - 1,34999	Range roll: -0,31429 - 0,28235	Keberhasilan
1		Yaw -1,31899 Pitch 1,23023 Roll 0,08108	Sesuai	Sesuai	Sesuai	3
2		Yaw 0,21699 Pitch 1,27903 Roll 0,08333	Sesuai	Sesuai	Sesuai	3
3		Yaw 0,29776 Pitch 1,24080 Roll 0,05405	Sesuai	Sesuai	Sesuai	3
4		Yaw -1,19736 Pitch 1,29204 Roll 0,02778	Sesuai	Sesuai	Sesuai	3
5		Yaw 0,22701 Pitch 1,27192 Roll	Sesuai	Sesuai	Sesuai	3



		-0,11111				
6		Yaw 0,21197 Pitch 1,28387 Roll 0,04110	Sesuai	Sesuai	Sesuai	3

Hasil pengujian data uji ke-1 pada Tabel 6.11 menunjukkan bahwa *output* nilai *yaw*, *pitch* dan *roll* termasuk dalam *range* batas atas dan batas bawah sehingga mencapai tingkat keberhasilan 100%.

**Tabel 6.24 Pengujian Analisis Nilai Batas Atas dan Batas Bawah Data Uji Ke-2**

No	Citra	Output Nilai Yaw, Pitch dan Roll	Range yaw: -1,42353 - 0,46262	Range pitch: 1,18538 - 1,34999	Range roll: -0,31429 - 0,28235	Keberhasilan
1		Yaw 0,12859 Pitch 1,32903 Roll -0,04762	Sesuai	Sesuai	Sesuai	3
2		Yaw 0,34196 Pitch 1,21988 Roll -0,08065	Sesuai	Sesuai	Sesuai	3
3		Yaw 0,34609 Pitch 1,21462 Roll -0,17460	Sesuai	Sesuai	Sesuai	3
4		Yaw 0,34240 Pitch 1,19495 Roll -0,38983	Sesuai	Sesuai	Sesuai	3
5		Yaw 0,23570 Pitch 1,26776 Roll	Sesuai	Sesuai	Sesuai	3



		-0,11111				
6		Yaw - 1,20152 Pitch 1,28422 Roll 0,11111	Sesuai	Sesuai	Sesuai	3
7		Yaw 0,10392 Pitch 1,34567 Roll 0,01613	Sesuai	Sesuai	Sesuai	3
8		Yaw 0,11033 Pitch 1,33915 Roll -0,05000	Sesuai	Sesuai	Sesuai	3
9		Yaw 0,35398 Pitch 1,21534 Roll -0,01538	Sesuai	Sesuai	Sesuai	3
10		Yaw 0,28886 Pitch 1,24244 Roll -0,11864	Sesuai	Sesuai	Sesuai	3
11		Yaw -Nan Pitch -Nan Roll 0,05085	Tidak Sesuai	Tidak Sesuai	Sesuai	1
12		Yaw 0,20152 Pitch 1,29018 Roll 0,00000	Sesuai	Sesuai	Sesuai	3

Hasil pengujian data uji ke-2 pada Tabel 6.12 menunjukkan bahwa *output* nilai *yaw*, *pitch* dan *roll* mencapai tingkat keberhasilan 94,4%.

Tabel 6.25 Pengujian Analisis Nilai Batas Atas dan Batas Bawah Data Uji Ke-3

No	Citra	Output Nilai Yaw, Pitch dan Roll	Range yaw: -1,42353 - 0,46262	Range pitch: 1,18538 - 1,34999	Range roll: -0,31429 - 0,28235	Keberhasilan
1		Yaw 0,10677 Pitch 1,34155 Roll -0,04688	Sesuai	Sesuai	Sesuai	3
2		Yaw 0,16260 Pitch 1,30930 Roll -0,05797	Sesuai	Sesuai	Sesuai	3
3		Yaw 0,16681 Pitch 1,30909 Roll 0,01471	Sesuai	Sesuai	Sesuai	3
4		Yaw 0,32958 Pitch 1,22568 Roll - 0,07042	Sesuai	Sesuai	Sesuai	3
5		Yaw - Pitch - Roll -	Tidak Sesuai	Tidak Sesuai	Tidak Sesuai	0
6		Yaw 0,16681 Pitch 1,30909 Roll 0,01471	Sesuai	Sesuai	Sesuai	3
7		Yaw 0,26927 Pitch 1,24024	Sesuai	Sesuai	Sesuai	3

		Roll -0,23077				
--	--	------------------	--	--	--	--

Hasil pengujian data uji ke-3 pada Tabel 6.13 menunjukkan bahwa *output* nilai *yaw*, *pitch* dan *roll* mencapai tingkat keberhasilan 85,7%.

**Tabel 6.26 Pengujian Analisis Nilai Batas Atas dan Batas Bawah Data Uji Ke-4**

No	Citra	Output Nilai Yaw, Pitch dan Roll	Range yaw: -1,42353 - 0,46262	Range pitch: 1,18538 - 1,34999	Range roll: -0,31429 - 0,28235	Keberhasilan
1		Yaw - Pitch - Roll -	Tidak Sesuai	Tidak Sesuai	Tidak Sesuai	0
2		Yaw 0,29703 Pitch 1,24120 Roll 0,05172	Sesuai	Sesuai	Sesuai	3
3		Yaw - 1,35733 Pitch 1,21132 Roll 0,14286	Sesuai	Sesuai	Sesuai	3
4		Yaw - 1,32417 Pitch 1,22773 Roll 0,08621	Sesuai	Sesuai	Sesuai	3
5		Yaw -1,30691 Pitch 1,23570 Roll 0,08333	Sesuai	Sesuai	Sesuai	3
6		Yaw 0,24330 Pitch 1,25675 Roll	Sesuai	Sesuai	Sesuai	3



		-0,18333				
7		Yaw -1,28368 Pitch 1,24234 Roll 0,15517	Sesuai	Sesuai	Sesuai	3
8		Yaw 0,19502 Pitch 1,29353 Roll 0,01724	Sesuai	Sesuai	Sesuai	3
9		Yaw -1,27048 Pitch 1,25268 Roll 0,08621	Sesuai	Sesuai	Sesuai	3

Hasil pengujian data uji ke-4 pada Tabel 6.14 menunjukkan bahwa *output* nilai yaw, pitch dan roll mencapai tingkat keberhasilan 88,8%.

**Tabel 6.27 Pengujian Analisis Nilai Batas Atas dan Batas Bawah Data Uji Ke-5**

No	Citra	Output Nilai Yaw, Pitch dan Roll	Range yaw: -1,42353 - 0,46262	Range pitch: 1,18538 - 1,34999	Range roll: -0,31429 - 0,28235	Keberhasilan
1		Yaw 0,38384 Pitch 1,20216 Roll -0,01695	Sesuai	Sesuai	Sesuai	3
2		Yaw 0,24878 Pitch 1,25844 Roll -0,14754	Sesuai	Sesuai	Sesuai	3
3		Yaw 0,14650 Pitch 1,32077 Roll 0,00000	Sesuai	Sesuai	Sesuai	3



4		<p><i>Yaw</i> 0,12259 <i>Pitch</i> 1,33450 <i>Roll</i> 0,01613</p>	Sesuai	Sesuai	Sesuai	3
5		<p><i>Yaw -</i> 1,27349 <i>Pitch</i> 1,24278 <i>Roll</i> -0,19672</p>	Sesuai	Sesuai	Sesuai	3
6		<p><i>Yaw</i> 0,42162 <i>Pitch</i> 1,18538 <i>Roll</i> -0,11290</p>	Sesuai	Sesuai	Sesuai	3
7		<p><i>Yaw</i> 0,23666 <i>Pitch</i> 1,26701 <i>Roll</i> -0,11475</p>	Sesuai	Sesuai	Sesuai	3
8		<p><i>Yaw</i> 0,32858 <i>Pitch</i> 1,22688 <i>Roll</i> -0,01786</p>	Sesuai	Sesuai	Sesuai	3
9		<p><i>Yaw</i> 0,47515 <i>Pitch</i> 1,16431 <i>Roll</i> 0,06977</p>	Sesuai	Sesuai	Sesuai	3
10		<p><i>Yaw</i> 0,34910 <i>Pitch</i> 1,21711 <i>Roll</i> 0,05882</p>	Sesuai	Sesuai	Sesuai	3



Hasil pengujian data uji ke-5 pada Tabel 6.15 menunjukkan bahwa *output* nilai *yaw*, *pitch* dan *roll* termasuk dalam *range* batas atas dan batas bawah sehingga mendapatkan tingkat keberhasilan 100%.

**Tabel 6.28 Pengujian Analisis Nilai Batas Atas dan Batas Bawah Data Uji Ke-6**

No	Citra	Output Nilai Yaw, Pitch dan Roll	Range yaw: -1,42353 - 0,46262	Range pitch: 1,18538 - 1,34999	Range roll: -0,31429 - 0,28235	Keberhasilan
1		Yaw 0,20711 Pitch 1,28706 Roll 0,01667	Sesuai	Sesuai	Sesuai	3
2		Yaw Nan Pitch Nan Roll Nan	Sesuai	Sesuai	Sesuai	3
3		Yaw 0,36813 Pitch 1,20882 Roll 0,04839	Sesuai	Sesuai	Sesuai	3
4		Yaw 0,34822 Pitch 1,21778 Roll 0,03704	Sesuai	Sesuai	Sesuai	3
5		Yaw - Pitch - Roll -	Tidak Sesuai	Tidak Sesuai	Tidak Sesuai	0
6		Yaw - Pitch - Roll	Tidak Sesuai	Tidak Sesuai	Tidak Sesuai	0

		-				
7		Yaw 0,36650 Pitch 1,20968 Roll 0,03077	Sesuai	Sesuai	Sesuai	3
8		Yaw - Pitch - Roll -	Tidak Sesuai	Tidak Sesuai	Tidak Sesuai	0
9		Yaw -1,20711 Pitch 1,27677 Roll -0,15000	Sesuai	Sesuai	Sesuai	3
10		Yaw -1,49658 Pitch 1,15598 Roll 0,10345	Sesuai	Sesuai	Sesuai	3

Hasil pengujian data uji ke-6 pada Tabel 6.16 menunjukkan bahwa *output* nilai *yaw*, *pitch* dan *roll* mencapai tingkat keberhasilan 70%.

**Tabel 6.29 Pengujian Analisis Nilai Batas Atas dan Batas Bawah Data Uji Ke-7**

No	Citra	Output Nilai Yaw, Pitch dan Roll	Range yaw: -1,42353 - 0,46262	Range pitch: 1,18538 - 1,34999	Range roll: -0,31429 - 0,28235	Keberhasilan
1		Yaw - Pitch - Roll -	Tidak Sesuai	Tidak Sesuai	Tidak Sesuai	0
2		Yaw 0,32904 Pitch 1,22466	Sesuai	Sesuai	Sesuai	3

		<i>Roll</i> 0,11290				
3		<i>Yaw</i> -1,24211 <i>Pitch</i> 1,26167 <i>Roll</i> 0,14516	Sesuai	Sesuai	Sesuai	3
4		<i>Yaw</i> -1,16898 <i>Pitch</i> 1,30603 <i>Roll</i> 0,05556	Sesuai	Sesuai	Sesuai	3
5		<i>Yaw</i> -1,21210 <i>Pitch</i> 1,28236 <i>Roll</i> 0,07042	Sesuai	Sesuai	Sesuai	3
6		<i>Yaw</i> 0,13450 <i>Pitch</i> 1,26856 <i>Roll</i> 0,25000	Sesuai	Sesuai	Sesuai	3
7		<i>Yaw</i> 0,22954 <i>Pitch</i> 1,27085 <i>Roll</i> -0,10938	Sesuai	Sesuai	Sesuai	3
8		<i>Yaw</i> -1,18875 <i>Pitch</i> 1,28571 <i>Roll</i> 0,14474	Sesuai	Sesuai	Sesuai	3

Hasil pengujian data uji ke-7 pada Tabel 6.17 menunjukkan bahwa *output* nilai *yaw*, *pitch* dan *roll* mencapai tingkat keberhasilan 87,5%.

Tabel 6.30 Pengujian Analisis Nilai Batas Atas dan Batas Bawah Data Uji Ke-8

No	Citra	Output Nilai Yaw, Pitch dan Roll	Range yaw: -1,42353 - 0,46262	Range pitch: 1,18538 - 1,34999	Range roll: -0,31429 - 0,28235	Keberhasilan
1		Yaw -1,30318 Pitch 1,23817 Roll 0,05714	Sesuai	Sesuai	Sesuai	3
2		Yaw 0,33333 Pitch 1,22423 Roll 0,05797	Sesuai	Sesuai	Sesuai	3
3		Yaw - Pitch - Roll -	Tidak Sesuai	Tidak Sesuai	Tidak Sesuai	0
4		Yaw 0,15420 Pitch 1,30469 Roll 0,12500	Sesuai	Sesuai	Sesuai	3
5		Yaw -1,20200 Pitch 1,28982 Roll 0,01449	Sesuai	Sesuai	Sesuai	3
6		Yaw 0,24099 Pitch 1,26740 Roll 0,07812	Sesuai	Sesuai	Sesuai	3

7		<i>Yaw</i> -1,37707 <i>Pitch</i> 1,20495 <i>Roll</i> 0,04412	Sesuai	Sesuai	Sesuai	3
8		<i>Yaw</i> 0,23481 <i>Pitch</i> 1,27053 <i>Roll</i> 0,07692	Sesuai	Sesuai	Sesuai	3
9		<i>Yaw</i> -0,50000 <i>Pitch</i> -Nan <i>Roll</i> -0,27869	Sesuai	Tidak Sesuai	Sesuai	2
10		<i>Yaw</i> 0,11110 <i>Pitch</i> 1,32406 <i>Roll</i> 0,12162	Sesuai	Sesuai	Sesuai	3

Hasil pengujian data uji ke-8 pada Tabel 6.18 menunjukkan bahwa *output* nilai *yaw*, *pitch* dan *roll* mencapai tingkat keberhasilan 86,6%.

**Tabel 6.31 Pengujian Analisis Nilai Batas Atas dan Batas Bawah Data Uji Ke-9**

No	Citra	Output Nilai Yaw, Pitch dan Roll	Range yaw: -1,42353 - 0,46262	Range pitch: 1,18538 - 1,34999	Range roll: -0,31429 - 0,28235	Keberhasilan
1		<i>Yaw</i> - 1,34028 <i>Pitch</i> 1,21797 <i>Roll</i> 0,15584	Sesuai	Sesuai	Sesuai	3
2		<i>Yaw</i> -1,29205 <i>Pitch</i> 1,24192 <i>Roll</i> 0,10000	Sesuai	Sesuai	Sesuai	3



3		<i>Yaw</i> -1,14603 <i>Pitch</i> 1,31845 <i>Roll</i> 0,05682	Sesuai	Sesuai	Sesuai	3
4		<i>Yaw</i> -1,15016 <i>Pitch</i> 1,31228 <i>Roll</i> 0,09091	Sesuai	Sesuai	Sesuai	3
5		<i>Yaw</i> - <i>Pitch</i> - <i>Roll</i> -	Tidak Sesuai	Tidak Sesuai	Tidak Sesuai	0
6		<i>Yaw</i> 0,14873 <i>Pitch</i> 1,31894 <i>Roll</i> 0,02667	Sesuai	Sesuai	Sesuai	3
7		<i>Yaw</i> 0,24384 <i>Pitch</i> 1,24407 <i>Roll</i> -0,26087	Sesuai	Sesuai	Sesuai	3
8		<i>Yaw</i> -1,21673 <i>Pitch</i> 1,26431 <i>Roll</i> 0,20270	Sesuai	Sesuai	Sesuai	3

Hasil pengujian data uji ke-9 pada Tabel 6.19 menunjukkan bahwa *output* nilai *yaw*, *pitch* dan *roll* mendapatkan tingkat keberhasilan 87,5%.

Tabel 6.32 Pengujian Analisis Nilai Batas Atas dan Batas Bawah Data Uji Ke-10

No	Citra	Output Nilai Yaw, Pitch dan Roll	Range yaw: -1,42353 - 0,46262	Range pitch: 1,18538 - 1,34999	Range roll: -0,31429 - 0,28235	Keberhasilan
1		Yaw - Pitch - Roll -	Tidak Sesuai	Tidak Sesuai	Tidak Sesuai	0
2		Yaw - Pitch - Roll -	Tidak Sesuai	Tidak Sesuai	Tidak Sesuai	0
3		Yaw -1,32973 Pitch 1,22640 Roll -0,00000	Sesuai	Sesuai	Sesuai	3
4		Yaw -1,35912 Pitch 1,21305 Roll -0,01370	Sesuai	Sesuai	Sesuai	3
5		Yaw 0,31839 Pitch 1,23135 Roll -0,04225	Sesuai	Sesuai	Sesuai	3
6		Yaw 0,32942 Pitch 1,22600 Roll -0,05882	Sesuai	Sesuai	Sesuai	3



7		Yaw - Pitch - Roll -	Tidak Sesuai	Tidak Sesuai	Tidak Sesuai	0
8		Yaw -1,21197 Pitch 1,28387 Roll -0,04110	Sesuai	Sesuai	Sesuai	3
9		Yaw -1,31955 Pitch 1,23066 Roll -0,05195	Sesuai	Sesuai	Sesuai	3
10		Yaw 0,43314 Pitch 1,17649 Roll -0,28788	Sesuai	Sesuai	Sesuai	3
11		Yaw -1,29205 Pitch 1,24043 Roll 0,12857	Sesuai	Sesuai	Sesuai	3

Hasil pengujian data uji ke-10 pada Tabel 6.20 menunjukkan bahwa *output* nilai *yaw*, *pitch* dan *roll* mencapai tingkat keberhasilan 72,7%.

Total keseluruhan data citra dari pengujian data uji ke-1 sampai dengan data uji ke-10 adalah 91 data citra. Data Citra yang berhasil berjumlah 78 sedangkan yang tidak berhasil berjumlah 13 data citra sehingga akurasi dari pengujian batas atas dan batas bawah dari nilai *yaw*, *pitch* dan *roll* adalah 85.7%.

### 6.3 Pengujian Arah Pergerakan Kepala

Pengujian ini diambil dari pengujian citra sebelumnya yang mendapatkan nilai keberhasilan dengan notasi angka 2-3 dengan parameter arah pergerakan kepala. Parameter yang digunakan adalah atas, bawah, kanan dan kiri. Jika *output* sesuai dengan kondisi arah kepala pada citra maka akan dinyatakan dengan kata “sesuai” pada tabel pengujian, jika tidak maka dinyatakan dengan kata “tidak sesuai”. Nilai *pitch* digunakan untuk menguji arah atas dan bawah sedangkan nilai *roll* untuk

menguji arah kanan dan kiri. Keberhasilan dalam pengujian ini di notasikan dengan angka 0-2. Angka 2 merupakan notasi angka sempurna.

**Tabel 6.33 Pengujian Data Uji Arah Atas**

No	Citra	Output Sistem	Kondisi Arah Kepala	Range pitch: 1,28056 - 1,34999	Keberhasilan
1		Kepala ke atas <i>Pitch 1,29204</i>	Sesuai	Sesuai	2
2		Kepala ke atas <i>Pitch 1,28387</i>	Tidak Sesuai	Sesuai	1
3		Kepala ke atas <i>Pitch 1,32903</i>	Sesuai	Sesuai	2
4		Kepala ke atas <i>Pitch 1,28422</i>	Sesuai	Sesuai	2
5		Kepala ke atas <i>Pitch 1,34567</i>	Sesuai	Sesuai	2
6		Kepala ke atas <i>Pitch 1,33915</i>	Sesuai	Sesuai	2



Tabel 6.34 Pengujian Data Uji Arah Atas (Lanjutan)

7		Kepala ke atas <i>Pitch 1,29018</i>	Sesuai	Sesuai	2
8		Kepala ke atas <i>Pitch 1,34155</i>	Sesuai	Sesuai	2
9		Kepala ke atas <i>Pitch 1,30930</i>	Tidak Sesuai	Sesuai	1
10		Kepala ke atas <i>Pitch 1,30909</i>	Tidak Sesuai	Sesuai	1
11		Kepala ke atas <i>Pitch 1,30909</i>	Tidak Sesuai	Sesuai	1
12		Kepala ke atas <i>Pitch 1,29353</i>	Tidak Sesuai	Sesuai	1
13		Kepala ke atas <i>Pitch 1,32077</i>	Sesuai	Sesuai	2
14		Kepala ke atas <i>Pitch 1,33450</i>	Sesuai	Sesuai	2

Tabel 6.35 Pengujian Data Uji Arah Atas (Lanjutan)

15		Kepala ke atas <i>Pitch 1,28706</i>	Sesuai	Sesuai	2
16		Kepala ke atas <i>Pitch 1,30603</i>	Sesuai	Sesuai	2
17		Kepala ke atas <i>Pitch 1,28236</i>	Tidak Sesuai	Sesuai	1
18		Kepala ke atas <i>Pitch 1,28571</i>	Sesuai	Sesuai	2
19		Kepala ke atas <i>Pitch 1,30469</i>	Sesuai	Sesuai	2
20		Kepala ke atas <i>Pitch 1,28982</i>	Sesuai	Sesuai	2
21		Kepala ke atas <i>Pitch 1,32406</i>	Sesuai	Sesuai	2
22		Kepala ke atas <i>Pitch 1,31845</i>	Sesuai	Sesuai	2

**Tabel 6.36 Pengujian Data Uji Arah Atas (Lanjutan)**

23		Kepala ke atas <i>Pitch 1,31228</i>	Sesuai	Sesuai	2
24		Kepala ke atas <i>Pitch 1,31894</i>	Sesuai	Sesuai	2
25		Kepala ke atas <i>Pitch 1,28387</i>	Sesuai	Sesuai	2

Dari data nilai keberhasilan pada Tabel 6.21 dapat diketahui bahwa nilai akurasi pengujian arah atas adalah 88%.

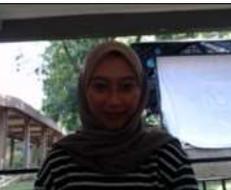
**Tabel 6.37 Pengujian Data Uji Arah Bawah**

No	Citra	Output Sistem	Kondisi Arah Kepala	Range pitch: 1,18538 - 1,23138	Keberhasilan
1		Kepala ke bawah <i>Pitch 1,23023</i>	Sesuai	Sesuai	2
2		Kepala ke bawah <i>Pitch 1,21988</i>	Sesuai	Sesuai	2
3		Kepala ke bawah <i>Pitch 1,19495</i>	Tidak Sesuai	Sesuai	1

Tabel 6.38 Pengujian Data Uji Arah Bawah (Lanjutan)

4		Kepala ke bawah <i>Pitch 1,21534</i>	Sesuai	Sesuai	2
5		Kepala ke bawah <i>Pitch 1,22568</i>	Tidak Sesuai	Sesuai	1
6		Kepala ke bawah <i>Pitch 1,21132</i>	Tidak Sesuai	Sesuai	1
7		Kepala ke bawah <i>Pitch 1,22773</i>	Tidak Sesuai	Sesuai	1
8		Kepala ke bawah <i>Pitch 1,20216</i>	Sesuai	Sesuai	2
9		Kepala ke bawah <i>Pitch 1,18538</i>	Sesuai	Sesuai	2
10		Kepala ke bawah <i>Pitch 1,22688</i>	Tidak Sesuai	Sesuai	1
11		Kepala ke bawah <i>Pitch 1,21711</i>	Tidak Sesuai	Sesuai	1

Tabel 6.39 Pengujian Data Uji Arah Bawah (Lanjutan)

12		Kepala ke bawah <i>Pitch 1,20882</i>	Sesuai	Sesuai	2
13		Kepala ke bawah <i>Pitch 1,21778</i>	Sesuai	Sesuai	2
14		Kepala ke bawah <i>Pitch 1,20968</i>	Sesuai	Sesuai	2
15		Kepala ke bawah <i>Pitch 1,22466</i>	Sesuai	Sesuai	2
16		Kepala ke bawah <i>Pitch 1,22423</i>	Sesuai	Sesuai	2
17		Kepala ke bawah <i>Pitch 1,20495</i>	Sesuai	Sesuai	2
18		Kepala ke bawah <i>Pitch 1,21797</i>	Sesuai	Sesuai	2
19		Kepala ke bawah <i>Pitch 1,22640</i>	Sesuai	Sesuai	2

**Tabel 6.40 Pengujian Data Uji Arah Bawah (Lanjutan)**

20		Kepala ke bawah <i>Pitch 1,21305</i>	Sesuai	Sesuai	2
21		Kepala ke bawah <i>Pitch 1,23135</i>	Tidak sesuai	Sesuai	1
22		Kepala ke bawah <i>Pitch 1,22600</i>	Tidak Sesuai	Sesuai	1
23		Kepala ke bawah <i>Pitch 1,23066</i>	Sesuai	Sesuai	2

Dari data nilai keberhasilan pada Tabel 6.22 dapat diketahui bahwa nilai akurasi pengujian arah bawah adalah 82,6%.

**Tabel 6.41 Pengujian Data Uji Arah Kanan**

No	Citra	Output Sistem	Kondisi Arah Kepala	Range roll: -0,10127 - -0,31429	Keberhasilan
1		Kepala ke kanan <i>Roll -0,11111</i>	Tidak Sesuai	Sesuai	1
2		Kepala ke kanan <i>Roll -0,17460</i>	Sesuai	Sesuai	2

Tabel 6.42 Pengujian Data Uji Arah Kanan (Lanjutan)

3		Kepala ke kanan <i>Roll -0,11111</i>	Sesuai	Sesuai	2
4		Kepala ke kanan <i>Roll -0,11864</i>	Sesuai	Sesuai	2
5		Kepala ke kanan <i>Roll -0,23077</i>	Sesuai	Sesuai	2
6		Kepala ke kanan <i>Roll -0,18333</i>	Sesuai	Sesuai	2
7		Kepala ke kanan <i>Roll -0,14754</i>	Tidak Sesuai	Sesuai	1
8		Kepala ke kanan <i>Roll -0,11475</i>	Tidak Sesuai	Sesuai	1
9		Kepala ke kanan <i>Roll -0,10345</i>	Sesuai	Sesuai	2
10		Kepala ke kanan <i>Roll -0,10938</i>	Sesuai	Sesuai	2

**Tabel 6.43 Pengujian Data Uji Arah Kanan (Lanjutan)**

11		Kepala ke kanan <i>Roll -0,27869</i>	Sesuai	Sesuai	2
12		Kepala ke kanan <i>Roll -0,26087</i>	Sesuai	Sesuai	2
13		Kepala ke kanan <i>Roll -0,28788</i>	Tidak Sesuai	Sesuai	1

Dari data nilai keberhasilan pada Tabel 6.23 dapat diketahui bahwa nilai akurasi pengujian arah kanan adalah 84,6%.

**Tabel 6.44 Pengujian Data Uji Arah Kiri**

No	Citra	Output Sistem	Kondisi Arah Kepala	Range roll: 0,05000 - 0,28235	Keberhasilan
1		Kepala ke kiri <i>Roll 0,08333</i>	Tidak Sesuai	Sesuai	1
2		Kepala ke kiri <i>Roll 0,05405</i>	Tidak Sesuai	Sesuai	1
3		Kepala ke kiri <i>Roll 0,05085</i>	Sesuai	Sesuai	2

Tabel 6.45 Pengujian Data Uji Arah Kiri (Lanjutan)

4		Kepala ke kiri <i>Roll</i> 0,05172	Tidak Sesuai	Sesuai	1
5		Kepala ke kiri <i>Roll</i> 0,08333	Tidak Sesuai	Sesuai	1
6		Kepala ke kiri <i>Roll</i> 0,15517	Sesuai	Sesuai	2
7		Kepala ke kiri <i>Roll</i> 0,08621	Tidak Sesuai	Sesuai	1
8		Kepala ke kiri <i>Roll</i> 0,06977	Sesuai	Sesuai	2
9		Kepala ke kiri <i>Roll</i> 0,10345	Sesuai	Sesuai	2
10		Kepala ke kiri <i>Roll</i> 0,07042	Tidak Sesuai	Sesuai	1
11		Kepala ke kiri <i>Roll</i> 0,25000	Sesuai	Sesuai	2

**Tabel 6.46 Pengujian Data Uji Arah Kiri (Lanjutan)**

12		Kepala ke kiri <i>Roll 0,05714</i>	Tidak Sesuai	Sesuai	1
13		Kepala ke kiri <i>Roll 0,07812</i>	Tidak Sesuai	Sesuai	1
14		Kepala ke kiri <i>Roll 0,07692</i>	Tidak Sesuai	Sesuai	1
15		Kepala ke kiri <i>Roll 0,10000</i>	Sesuai	Sesuai	2
16		Kepala ke kiri <i>Roll 0,20270</i>	Sesuai	Sesuai	2
17		Kepala ke kiri <i>Roll 0,12857</i>	Sesuai	Sesuai	2

Dari data nilai keberhasilan pada Tabel 6.24 dapat diketahui bahwa nilai akurasi pengujian arah kiri adalah 73,5%.

**Tabel 6.47 Akurasi Masing-Masing Arah Pergerakan Kepala**

Arah Atas	Arah Bawah	Arah Kanan	Arah Kiri
88%	82,6%	84,6%	73,5%

Tabel 6.25 menunjukkan nilai akurasi masing – masing arah pergerakan kepala. Hasil pengujian menunjukkan bahwa kondisi arah kiri memiliki nilai yang rendah dibandingkan dengan arah yang lain, mendapatkan nilai akurasi 73.5%. Hal



tersebut dikarenakan hasil *output* yang terseleksi dari tahapan awal banyak menyisakan *frame* dengan arah pergerakan kepala yang tidak sesuai dengan keadaan kepala secara nyata pada tahap pengujian arah pergerakan kepala. Faktor lain penyebab hasil akurasi adalah kondisi intensitas cahaya. Menganalisis pengujian yang telah dilakukan, intensitas cahaya dipengaruhi oleh beberapa faktor antara lain kondisi tempat dan waktu.



## BAB 7 PENUTUP

### 7.1 Simpulan

Setelah dilakukan penelitian mengenai *head detection* dan *tracking* menggunakan metode *Triangle Geometry*, terdapat beberapa simpulan yang diambil antara lain.

1. *Triangle Geometry* dapat dilakukan sebagai metode untuk mengenali posisi arah kepala, dengan melalui beberapa tahap utama antara lain melakukan koreksi nilai bobot *brightness* dan *contrast* pada tahap *pre-processing*. Tahap lainnya adalah tahap deteksi warna kulit yang menggunakan metode *RGB-H-CbCr*. Tahap ini akan mengeliminasi warna selain kulit sesuai dengan nilai bobot dan merubah citra menjadi angka decimal yang dapat dilakukan perhitungan sesuai dengan persamaan metode sehingga warna kulit dapat dideteksi dan memudahkan untuk tahap selanjutnya yakni deteksi mata dan hidung. Tahap deteksi mata dan hidung memanfaatkan *library* *openCv*. Deteksi mata dan hidung digunakan untuk mendapatkan nilai *variable* sesuai dengan persamaan metode yang selanjutnya dilakukan perhitungan nilai *yaw*, *pitch*, *roll* dengan metode *Triangle Geometry*. Setelah nilai *yaw*, *pitch*, *roll* didapatkan maka akan dilakukan pengklasifikasian nilai tersebut ke dalam *range* arah untuk mengenali arah masing-masing pergerakan kepala sesuai dengan keadaan *realtime*.
2. Analisis nilai batas atas dan batas bawah *yaw*, *pitch*, *roll* pada arah pergerakan kepala (kanan, kiri, atas, bawah) adalah nilai *yaw* tidak dapat dijadikan acuan untuk menentukan arah pergerakan kepala dikarenakan harus tetap mempertahankan objek kedua bola mata terlihat pada kamera untuk dapat mendeteksi wajah dan dilanjutkan untuk proses deteksi arah pergerakan kepala. Akurasi pada pengujian ini menunjukkan angka sebesar 85,7%.
3. Nilai akurasi masing – masing arah pergerakan kepala menunjukkan angka 88% untuk arah atas, 82,6% untuk arah bawah, 84,6% untuk arah kanan dan 73,5% untuk arah kiri.

### 7.2 Saran

Setelah melakukan penelitian ini, terdapat beberapa saran yang ditujukan untuk penelitian lebih lanjut pada masa mendatang antara lain.

1. Pada Penelitian selanjutnya diharapkan dapat memperbaiki nilai intensitas cahaya agar sistem dapat lebih leluasa digunakan tanpa terhalang nilai intensitas cahaya.
2. Pada penelitian selanjutnya diharapkan menambahkan training pada fitur *haar cascade* dengan data training mata sipit sehingga data uji mendapatkan hasil lebih baik.
3. Pada penelitian berikutnya diharapkan melakukan pengujian *range* arah lebih banyak dilakukan sehingga di dapatkan *range* arah yang lebih akurat.



4. Pada penelitian selanjutnya dapat menerapkan metode klasifikasi untuk membantu mengklasifikasikan *range* arah, seperti penggunaan metode klasifikasi KNN.



## DAFTAR PUSTAKA

- Abdul Rahman, Nusirwan Anwar bin., Wei, Kit Chong., See, John., 2007. RGB-H-CbCr Skin Colour Model for Human Face Detection. Faculty of Information Technology, Multimedia University.
- Barr, P., Noble, J., & Biddle, R., 2007. Video *game* values: Human-computer interaction and *games*. *Interacting with Computers*, 19(2), 180-195.
- Churiwala, Karanraj., Lopes, Raoul., Shah, Aditya., Shah, Neepa., 2012. Drowsiness Detection b (Peer, et al., 2003)ased on Eye Movement, Yawn Detection and Head Rotation. *International Journal of Applied Information Systems (IJ AIS) – ISSN: 2249-0868, Foundation of Computer Science FCS, New York, USA Volume 2– No.6, May 2012.*
- Hajraoui, Abdellatif., Sabri, Mohamed., 2014. Face Detection Algorithm based on Skin Detection, Watershed Method and Gabor Filters. *International Journal of Computer Applications (0975 – 8887), Volume 94 – No 6.*
- Hidayat, Nurul., Rahman, Muh. Arif., 2015., Cara Cepat Untuk Mendeteksi Keberadaan Wajah Pada Citra Yang Mempunyai Background Kompleks Menggunakan Model Warna YCbCr Dan HSV. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*, Vol. 2, No. 2, hlm. 138-142.
- Kementerian Kesehatan RI. (2014, Desember). Buletin Jendela Data dan Informasi Kesehatan. Situasi Penyandang Disabilitas.
- Lumaris, Riandika., Setyati, Endang., 2015. Deteksi Dan Representasi Fitur Mata Pada Sebuah Citra Wajah Menggunakan Haar Cascade Dan Chain Code. Seminar Nasional “Inovasi dalam Desain dan Teknologi” - IDeaTech 2015, ISSN: 2089-1121.
- Manjare, Snehal., Chougule, S.R., 2013., *International Journal of Engineering Sciences & Research Technology*. *International Journal of Engineering Sciences & Research Technology [1883-1887], ISSN: 2277-9655.*
- Mulligan, Jeffrey B., Gabayan, Kevin N., 2010. Robust Optical Eye Detection During Head Movement. In *Proceedings of the 2010 Symposium on Eye Tracking Research and Applications (ETRA)*, ACM, New York, pp. 129-132.
- Murphy-Chutorian, Erik., Student Member, IEEE., Xu., Manubhai T., Mohan & Fellow., 2009. Head Pose Estimation in Computer Vision: ASurvey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 31, No. 4, April 2009.
- Papazov, Chavdar., Marks, Tim K., Jones, Michael Jeffrey., 2015. Real-time 3D Head Pose and Facial Landmark Estimation from Depth Images Using Triangular Surface Patch Features. *ResearchGate*.

- Peer, P., Kova, J. & Solina, F., 2003. Human Skin Colour Clustering for Face Detection. IEEE Region 8 EUROCON 2003. Computer as a Tool, Volume 2, pp. 144-148.
- Perreira Da Silvia, Matthieu., Courboulay, Vincent., Prigent, Armelle., Estrailier, Pascal., 2009. Fast, low resource, head detection and tracking for nteractive applications. PsychNology Journal, 7(3), 243 – 264.
- Rizvandi, Nikzad Babaii., Pižurica, Aleksandra., Philips, ilfried., 2008. VISAPP – International Conference on Computer Vision Theory and Applications.
- Saikia, Pallabi., Janam, Gollo., Kathing, Margaret., 2012. Face Detection using Skin Colour Model and distance between Eyes. International Journal of Computing, Communications and Networking, Volume 1, No.3, SSN 2319-2720.
- Singh, Pragati., Lodhi, Rashmi Singh., 2014. Hand Tracking and Head Movement Detection: A State-of-Art Survey. International Journal of Computer and Advanced Engineering Research (IJCAER), Volume 01– Issue 02.
- Sutoyo, T. et al., 2009. *Teori Pengolahan Citra Digital*. 1 penyunt. Yogyakarta;Semarang: ANDI Yogyakarta; UDINUS Semarang.
- Yu,Wang., Gang,Liu., 2011. Head Pose Estimation Based on Head Tracking and the Kalman Filter. International Conference on Physics Science and Technology (ICPST 2011), Physics Procedia 22 (2011) 420 – 427.
- Wibowo, Jati Sasongko., 2011. Deteksi dan Klasifikasi Citra Berdasarkan Warna Kulit Menggunakan HSV. Jurnal Teknologi Informasi DINAMIK Volume 16, No.2, Juli 2011: 118-123.
- Zhang, Wan-zhi., Wang, Zeng-cai., Xu, Jun-kai. & Cong, Xiao-yan., 2013. A Method of Gaze Direction Estimation Considering Head Posture. International Journal of Signal Processing, Image Processing and Pattern Recognition, Vol. 6, No. 2, April, 2013.