

INTERAKSI *HAND TRACKING* PADA OBJEK *AUGMENTED REALITY* MENGGUNAKAN *STEREO VISION*



OLEH:

ROBBYANSYAH ASMIR

09111001020

**JURUSAN SISTEM KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS SRIWIJAYA
2018**

**INTERAKSI *HAND TRACKING* PADA OBJEK *AUGMENTED REALITY*
MENGUNAKAN *STEREO VISION***

TUGAS AKHIR

**Diajukan Untuk Melengkapi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer**



OLEH :

**ROBBYANSYAH ASMIR
09111001020**

**JURUSAN SISTEM KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS SRIWIJAYA
2018**

LEMBAR PENGESAHAN

**INTERAKSI *HAND TRACKING* PADA OBJEK *AUGMENTED REALITY*
MENGUNAKAN *STEREO VISION***

TUGAS AKHIR

Diajukan Untuk Melengkapi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer

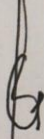
Oleh:

ROBBYANSYAH ASMIR

09111001020

Inderalaya, Mei 2018

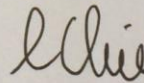
Pembimbing I,



Sutarno, M.T

NIP. 197811012010121003

Pembimbing II,

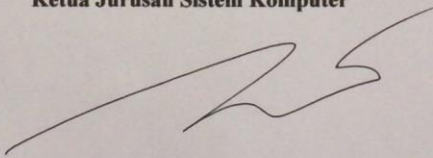


Sri Desy Siswanti, M.T

NIP. 197412072011082201

Mengetahui,

Ketua Jurusan Sistem Komputer



Rossi Passarella, M.Eng

NIP. 197806112010121004

HALAMAN PERSETUJUAN


Teah diuji dan lulus pada :

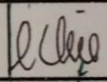
Hari : Sabtu

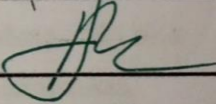
Tanggal : 24 Marct 2018

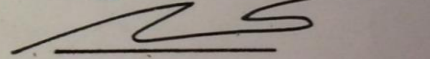
Tim Penguji :

1. Ketua : Sutarno, M.F.
2. Sekretaris : Sri Dedy Siwanti, M.T
3. Anggota I : Erwin, M.Si.
4. Anggota II : Rossi Passarella, M.Eng.

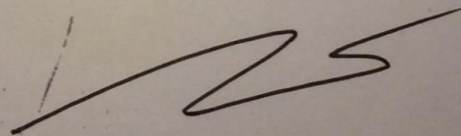








Mengetahui,
Ketua Jurusan Sistem Komputer



Rossi Passarella, M.Eng
NIP. 197806112010121004

HALAMAN PERNYATAAN KEASLIAN

Yang bertandatangan di bawah ini :

Nama : Robbyansyah Asmir

NIM : 09111001020

Menyatakan bahwa tugas akhir ini merupakan hasil karya saya sendiri dan bukan merupakan duplikasi maupun plagiasi (jiplakan) dari penelitian orang lain. Sepengetahuan saya, judul dari tugas akhir ini belum pernah ditulis oleh orang lain. Apabila tugas akhir ini terbukti merupakan hasil duplikasi atau plagiasi (jiplakan) dari hasil penelitian orang lain, maka saya bersedia menerima sanksi yang diberikan oleh Tim penguji dan jurusan Sistem Komputer Fakultas Ilmu Komputer Universitas Sriwijaya.

Demikian Surat Pernyataan ini saya buat dengan sebenar-benarnya.



Palembang, Mei 2018

Yang menyatakan,



Robbyansyah Asmir

NIM 09111001020

HALAMAN PERSEMBAHAN

الرَّحِيمِ الرَّحْمَنِ اللَّهُ بِسْمِ

**“sungguh atas kehendak Allah semua ini terwujud, tiada kekuatan kecuali
dengan pertolongan Allah”**

(QS. Al-Kahfi : 39)

Karya ini ku persembahkan untukmu, bapak dan umak

Terimakasih telah memberikan kasih sayang

yang tak terhingga untuk anakmu ini

Kesalahan, kenalakan dan kelalaian yang selama ini telah banyak kulakukan

Namun kalian senantiasa bersabar dan mendukungku dengan lantunan do'a

yang engkau panjatkan untukku

kurendahkan hati serta diri meminta beribu-ribu kata maaf tercurah

untuk menghapus semua khilaf ini

Hanya sebuah karya kecil dan untaian kata-kata ini yang dapat

kupersembahkan kepada kalian

Terimakasih bapak dan umak

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Allah SWT atas limpahan berkah, rahmat, taufik dan hidayah-Nya sehingga penulis dapat menyelesaikan tugas akhir dan menyusun laporan tugas akhir yang berjudul “**Interaksi *Hand Tracking* pada objek *Augmented Reality* menggunakan *Stereo Vision***“, dibuat dalam rangka memenuhi persyaratan untuk menyelesaikan pendidikan di jurusan Sistem Komputer Fakultas Ilmu Komputer Universitas Sriwijaya.

Dalam penulisan Tugas Akhir ini penulis menyadari bahwa penulis banyak sekali mendapat dukungan dari berbagai pihak. Oleh karena itu dalam kesempatan ini penulis ingin mengucapkan terima kasih kepada :

Oleh karena itu dalam kesempatan ini penulis ingin menyampaikan terima kasih kepada :

1. Allah Subhanahu Wa Ta'ala karna berkat dan rahmat-Nya penulis dapat menyelesaikan tugas akhir ini.
2. Nabi Muhammad Muhammad Shallallahu Alaihi Wasallam serta seluruh pengikutnya hingga akhir jaman
3. Terima kasih banyak penulis ucapkan Kedua orang tuaku Amiruddin dan Sri Mulyati, serta kakak dan adik-adikku Riyansyah Asmir, Rahma Cintya Anugrah, Afra Afifah yang selalu memberikan semangat,dukungan dan doa yang terbaik, serta pertolongan baik moril maupun materil.
4. Terima kasih banyak juga kepada mang Hendri dan tante Dina mardiana atas bantuannya selama ini yang telah menjadi motivator dan penyemangat saya sehingga dapat menyelesaikan tugas akhir ini
5. Terima kasih banyak untuk akas Asmawi sholeh dan dan embai Amsiah yang senantiasa memberikan dukungan ronahi dan jasmani kepada saya selama ini.
6. Terima kasih yang sebanyak-banyaknya kepada nenek tercinta Alm. H.Saliman yang telah menyekolahkan Cucumu ini sehingga bisa menjadi sarjana.

7. Terima kasih banyak kepada dosen pembimbingku bapak Sutarno dan ibu Sri Desi Siswanti yang telah membimbing mahasiswamu ini dari awal hingga dapat menyelesaikan skripsi.
8. Terima kasih banyak kepada dosen penguji bapak Erwin, M.Si. dan Rossi Passarella, M.Eng. selaku Dosen penguji sidang Tugas Akhir serta memberi banyak masukan untuk perbaikan tugas akhir ini
9. Terima kasih juga kepada teman-teman terdekatku Brillian Jhanatra, M. Akram Triparta, M. Hafizh YR, Yogi Duta Hartas, Dan Aprizal Ardi, lukie Herdi T, Titoyan, Rendika, Chandra, Bayu, Fitri, Herlina, Septi, Kemala, Lania, Wulan, Edho (Ridwan), Putri Baiduri, Pramudya, Theo, serta anak-anak SK 2011, 2012, 2013, 2014 dan semuanya
10. Terima kasih kepada anak-anak citin kost indralaya
11. Mbak Iis dan kak Reza selaku Admin Jurusan Sistem Komputer Fakultas Ilmu Komputer Universitas Sriwijaya.
12. Civitas akademika Jurusan Sistem Komputer Fakultas Ilmu Komputer Universitas Sriwijaya.

Penulis menyadari bahwa baik isi maupun penyajian laporan ini masih jauh dari kesempurnaan, oleh karena itu Penulis mengharapkan saran dan kritik yang sifatnya membangun untuk perbaikan laporan ini. Penulis berharap semoga Laporan Tugas Akhir ini dapat menambah pengetahuan serta dapat menunjang perkembangan ilmu pengetahuan khususnya bagi Penulis maupun pembaca khususnya mahasiswa / mahasiswi Fakultas Ilmu Komputer Universitas Sriwijaya..

Indralaya,

Penulis

INTERAKSI *HAND TRACKING* PADA OBJEK *AUGMENTED REALITY* MENGGUNAKAN *STEREO VISION*

Robbyansyah Asmir (09111001020)

Sistem Komputer, Fakultas Ilmu Komputer Universitas Sriwijaya

Email : Robbyasmir@gmail.com

Abstrak

Augmented Reality (AR) merupakan teknologi yang menggabungkan objek virtual 3D dengan dunia nyata menggunakan program tertentu. Interaksi langsung User dengan *Augmented Reality* jarang ditemukan pada aplikasi AR, jikapun ada maka harus menggunakan Alat bantu yang harganya tidak murah seperti *globe AR* dan *leap motion*. Penelitian ini bertujuan agar user dapat melakukan interaksi langsung dengan objek *Augmented Reality* secara *markless* menggunakan tangan. Penelitian ini terdiri dari beberapa tahapan diantaranya pendeteksian warna kulit menggunakan metode YCrCb, pengolahan citra isyarat jari tangan menggunakan metode *convexity hull defect*, *euclidean distance*, dan COG maka interaksi user dengan objek AR dapat dilakukan. Hasil dari penelitian ini didapatkan hasil jika objek Ar dapat berinteraksi langsung menggunakan tangan dengan lancar, tingkat akurasi sistem tergantung pada kondisi cahaya dan latar belakang pengujian.

Kata Kunci : *Augmented Reality*, *Stereo Vision*, YCrCb, pola isyarat jari tangan,

Realtime, *Virtual object*

INTERACTION HAND TRACKING ON AUGMENTED REALITY OBJECTS USING STEREO VISION

Robbyansyah Asmir - 09111001020

Computer Engineering, Computer Science Faculty of Sriwijaya University

Email : Robbyasmir@gmail.com

Abstract

Augmented Reality (AR) is a technology that combines 3D virtual objects with the real world using a specific program. Direct Interaction Users with Augmented Reality are rarely found in AR applications, even if they have to use tools that are not cheap like Globe AR and leap motion. This study aims to enable users to interact directly with markless Augmented Reality objects by hand. This research consists of several stages such as skin color detection using YCrCb method, finger gesture processing using convexity hull defect, euclidean distance, and COG method, user interaction with AR object can be done. The results of this study obtained results if the Ar object can interact directly using the hands smoothly, the level of system accuracy depends on light conditions and background.

Keywords: Augmented Reality, Stereo Vision, YCrCb, finger gesture ,

Realtime, Virtual object

DAFTAR ISI

	Halaman
HALAMAN JUDUL.....	<u>i</u>
LEMBAR PENGESAHAN.....	ii
HALAMAN PERSETUJUAN.....	iii
HALAMAN PERNYATAAN KEASLIAN.....	iv
HALAMAN PERSEMBAHAN.....	v
KATA PENGANTAR.....	vi
ABSTRACT.....	viii
ABSTRAK.....	ix
DAFTAR ISI.....	x
DAFTAR GAMBAR	xiii
DAFTAR TABEL.....	<u>xv</u>
DAFTAR LAMPIRAN.....	<u>xvi</u>
BAB I PENDAHULUAN.....	<u>1</u>
1.1. Latar Belakang.....	<u>1</u>
1.2. Rumusan dan dan Batasan masalah.....	<u>2</u>
1.3. Tujuan dan Manfaat.....	3
1.3.1. Tujuan.....	3
1.3.2. Manfaat.....	<u>3</u>
1.4. Metodologi Penelitian.....	3
1.7. Sistematika Penulisan	5
BAB II TINJAUAN PUSTAKA	<u>7</u>
2.1. Pendahuluan.....	<u>7</u>
2.2. <i>Stereo Vision</i>	7
2.3. <i>Hand Tracking</i> pada objek <i>Augmented Reality</i>	8
2.3.1. Segmentasi Warna Kulit.....	8
2.3.2. <i>Convexity Hull defect</i>	<u>9</u>
2.3.4. <i>Center of Gravity (COG)</i>	<u>11</u>
2.3.5. <i>Euclidean Distance</i>	<u>12</u>

2.4.	<i>Augmented Reality</i> (AR).....	13
2.5.	Cara Kerja <i>Augmented Reality</i>	13
2.6.	Tipe <i>Augmented Reality</i>	14
BAB III	METODOLOGI	<u>17</u>
3.1.	Pendahuluan.....	<u>17</u>
3.2.	Kerangka Kerja (<i>framework</i>).....	<u>17</u>
3.3.	Menentukan Objek penelitian dan Kamera	19
3.4.	Perancangan Perangkat Lunak.....	19
3.4.1.	Perancangan Sistem Perangkat Lunak.....	20
3.4.2.	Citra Input (Citra Asli)	21
3.4.3.	Image Processing	21
3.4.3.1.	Pengenalan Tangan.....	21
_____3.4.3.1.1.	Konversi Ruang Warna.....	21
_____3.4.3.1.2.	Deteksi Warna Kulit.....	21
_____3.4.3.1.3.	Penentuan Titik Fitur Ujung Jari	<u>22</u>
3.4.4.	Navigasi Objek <i>Augmented Reality</i>	<u>23</u>
3.4.5.	Perancangan <i>Interface</i>	<u>25</u>
BAB IV	PENGUJIAN DAN ANALISA	28
4.1.	Pendahuluan	28
4.2.	Pengujian Kamera Minoru 3D (<i>Stereo Vision</i>).....	28
4.3.	Pengujian Perangkat Lunak	28
4.3.1.	<i>Image Processing</i>	29
_____4.3.1.1.	Pengambilan Gambar	29
_____4.3.1.2.	Ruang Warna YCrCb	29
_____4.3.1.3.	Segmentasi Warna Kulit.....	30
4.3.2.	Pengujian Warna Kulit	32
4.3.2.1.	Pengujian Berdasarkan Kondisi Cahaya	34
4.2.4.2.	Pengujian Konversi Nilai RGB Ke YCrCb.....	37
4.3.4.	Pengujian Nilai Segmentasi	40
4.3.5.	Pola Isyarat Jari Tangan	41
4.3.5.1.	Pengujian Nilai Pola Isyarat Jari Tangan	43
4.3.6.	Pengujian Navigasi Untuk Objek <i>Augmented Reality</i>	47

BAB V KESIMPULAN DAN SARAN	60
5.1. Kesimpulan.....	66
5.2. Saran.....	67
DAFTAR PUSTAKA	68
LAMPIRAN.....	69

DAFTAR GAMBAR

Halaman

Gambar 1.1. <i>Flowchart</i> Metodologi Penelitian	4
Gambar 2.1. Kamera minoru 3D	8
Gambar 2.2. Ruang Warna YCrCb.....	9
Gambar 2.3. Arah dari 3 poin <i>Convexity hull deffect</i>	11
Gambar 2.4. Algoritma Graham Scan[5].....	11
Gambar 2.5. Diagram Sistem Cara Kerja <i>Augmented Reality</i>	14
Gambar 2.6. Contoh Marker	14
Gambar 2.7. Titik Koordinat <i>Virtual</i> pada <i>Marker</i>	15
Gambar 3.1. Kerangka Kerja (<i>Framework</i>).....	18
Gambar 3.2. Kamera Minoru 3D.....	19
Gambar 3.3. Bagan Umum Sistem	20
Gambar 3.4. <i>flowchart</i> Segmentasi warna kulit.	22
Gambar 3.5. <i>Flowchart hand tracking.</i>	23
Gambar 3.6. <i>Flowchart</i> navigasi objek AR.....	24
Gambar 3.7. <i>Interface</i> sistem 1	25
Gambar 3.8. <i>interface</i> sistem 2.....	25
Gambar 3.9. <i>interface</i> sistem 3	26
Gambar 4.1. Kamera Minoru 3D.....	28
Gambar 4.2. Pengambilan Citra Gambar Tangan.....	29
Gambar 4.3. Nilai Ruang Warna YCrCb.....	30
Gambar 4.4. Segmentasi warna kulit.....	31
Gambar 4.5. Rentang nilai YCrCb segmentasi warna kulit.	31
Gambar 4.6. Percobaan pertama kulit putih.	32
Gambar 4.7. Percobaan ketiga kulit sedang.	33
Gambar 4.8. Percobaan ketiga kulit sedang.	33
Gambar 4.9. Percobaan ketiga kulit gelap	35
Gambar 4.10. Percobaan pertama kondisi cahaya terang.....	36
Gambar 4.11. Percobaan pertama kondisi cahaya sedang.....	36
Gambar 4.12. Percobaan pertama kondisi minim cahaya	36

Gambar 4.13. Citra Tangan	38
Gambar 4.14. Input Citra Tangan	38
Gambar 4.13. tangan yang tersegmetasi	40
Gambar 4.14. Nilai Segmentasi	41
Gambar 4.15. (1) Citra isyarat jari tangan navigasi objek AR terdeteksi (2) Citra isyarat jari tangan navigasi objek AR tidak terdeteksi (3) Citra isyarat jari tangan navigasi objek AR <i>Zoom In</i> (4) Citra isyarat jari tangan navigasi objek AR <i>Zoom Out</i>	43
Gambar 4.16. Objek Augmented Reality	43
Gambar 4.17. setting kalibari YCrCb kondisi siang hari cahaya sedang.	47
Gambar 4.18. Navigasi Deteksi (D) pada objek Augmented Reality.....	48
Gambar 4.19. Navigasi Tidak Deteksi (D!) pada objek Augmented Reality. .	49
Gambar 4.20. Navigasi Mengecil (-) pada objek Augmented Reality.	50
Gambar 4.21. Navigasi Membesar (+) pada objek Augmented Reality.....	52
Gambar 4.22. setting kalibari YCrCb kondisi siang hari cahaya sedang.	53
Gambar 4.23. Navigasi Deteksi (D) pada objek Augmented Reality.....	54
Gambar 4.24. Navigasi Tidak Deteksi (D!) pada objek Augmented Reality. .	55
Gambar 4.25. Navigasi Mengecil (-) pada objek Augmented Reality.	56
Gambar 4.26. Navigasi Membesar (+) pada objek Augmented Reality.....	58
Gambar 4.27. setting kalibari YCrCb kondisi siang hari cahaya sedang.	59
Gambar 4.28. Navigasi Deteksi (D) pada objek Augmented Reality.....	60
Gambar 4.29. Navigasi Tidak Deteksi (D!) pada objek Augmented Reality. .	61
Gambar 4.30. Navigasi Mengecil (-) pada objek Augmented Reality.	62
Gambar 4.31. Navigasi Membesar (+) pada objek Augmented Reality.....	64

DAFTAR TABEL

Halaman

Tabel 1 Fungsi inputan citra pola isyarat menggunakan jari tangan.....	25
Tabel 2 Hasil Percobaan Pertama.....	32
Tabel 3 Hasil Percobaan Kedua	33
Tabel 4 Hasil Percobaan Ketiga	34
Tabel 5 Hasil Percobaan Pertama.....	35
Tabel 6 Hasil Percobaan Kedua	36
Tabel 7 Hasil Percobaan Ketiga	37
Tabel 8 Nilai RGB pada citra gambar tangan	38
Tabel 9 Data Ruang Warna YCrCb Hasil Perhitungan Sistem	39
Tabel 9 Data Ruang Warna YCrCb Hasil Perhitungan Sistem	39
Tabel 10 Data Ruang Warna YCrCb Hasil Perhitungan Manual.....	40
Tabel 11 Syarat inputan citra pola isyarat jari tangan	41
Tabel 12 Hasil Pengujian <i>euclidean distance</i> pada navigasi terdeteksi(D)	46
Tabel 13 Hasil Pengujian <i>euclidean distance</i> pada navigasi <i>Zoom Out(+)</i>	46
Tabel 14 Hasil Pengujian <i>euclidean distance</i> pada navigasi <i>Zoom In(-)</i>	46
Tabel 15 Data Pengujian Sistem Navigasi Deteksi (D).....	48
Tabel 16 Data Pengujian Sistem Navigasi Tidak Terdeteksi(D!).....	49
Tabel 17 Data Pengujian Sistem Navigasi <i>Zoom in (-)</i>	51
Tabel 18 Data Pengujian Sistem Navigasi <i>Zoom out (+)</i>	52
Tabel 19 Data Pengujian Sistem Navigasi Deteksi (D).....	54
Tabel 20 Data Pengujian Sistem Navigasi Tidak Terdeteksi(D!).....	55
Tabel 22 Data Pengujian Sistem Navigasi <i>Zoom in (-)</i>	57
Tabel 22 Data Pengujian Sistem Navigasi <i>Zoom out (+)</i>	58
Tabel 23 Data Pengujian Sistem Navigasi Deteksi (D).....	60
Tabel 24 Data Pengujian Sistem Navigasi Tidak Terdeteksi(D!).....	61
Tabel 25 Data Pengujian Sistem Navigasi <i>Zoom in (-)</i>	63
Tabel 26 Data Pengujian Sistem Navigasi <i>Zoom out (+)</i>	64

DAFTAR LAMPIRAN

- Nilai RGB
- Nilai YCrCb
- Nilai Segmentasi
- Data navigasi Objek *Augmented Reality* Deteksi (D), Tidak Terdeteksi (D!), *Zoom In* (-) dan *Zoom Out* (+)
- Berkas - Berkas Persyaratan

BAB I

PENDAHULUAN

1.1. Latar Belakang

Pesatnya perkembangan teknologi di dunia saat ini, tentu sudah tidak asing lagi dengan istilah *Augmented Reality*, pengertian dari *Augmented Reality* (AR) yaitu teknologi yang dibuat dengan cara menggabungkan secara *real-time* terhadap konten digital 2 dimensi atau 3 dimensi yang dibuat menggunakan laptop dan ditampilkan secara nyata.

Augmented reality (AR) telah menjadi teknik yang sangat bermanfaat bagi pengguna untuk mengalami perbedaan persepsi objek budaya yang diwakili dengan konten media yang dihasilkan komputer seperti model 3D, label, teks, gambar dan video pada lingkungan nyata [1]. Teknik interaksi *Augmented Reality* harus menjadi se-intuitif mungkin agar bisa diterima oleh *end-user* dan juga dapat disesuaikan untuk kebutuhan aplikasi lain, karena tangan merupakan anggota tubuh yang selalu digunakan manusia agar dapat berinteraksi pada lingkungan nyata, dan tangan dapat melakukan interaksi secara langsung pada konten virtual seperti *Augmented Reality* (AR).

Pengolahan citra dilakukan pada proses pendeteksian gerakan tangan *user*. Interaksi dari pengguna dengan objek virtual merupakan hal yang sangat penting dalam *Human Computer Interaction* (HCI). Pada penelitian yang dilakukan Y. Shiqiang, Q. Dan, and L. Peilei [2], menunjukkan bahwa penelitiannya yang mengenai pengenalan isyarat tangan menggunakan satu kamera dengan cara memisahkan warna kulit pada tangan dengan latar belakangnya. Setelah dilakukan segmentasi warna kulit maka pada penelitian ini akan dilakukan ditambahkan objek *Augmented Reality* pada penelitian yang dilakukan G. M. Re and M. Bordegoni [8], teknik interaksi *vision-based* pada *Augmented Reality* merupakan basis dari gambar 2D dari metode analisis dan pengenalan yang memiliki batasan untuk mendapatkan informasi citra 3D dari objek *virtual* dan pengguna. Untuk mengatasi masalah tersebut maka akan menggunakan kamera *stereo vision*. Sehingga bisa didapatkan informasi 3D dari interaksi antara tangan dengan objek *Augmented Reality* secara *realtime*.

Penelitian ini menggunakan dua kamera dengan 1 port USB atau lebih dikenal dengan Minoru 3D. Kamera minoru 3D bertugas untuk mendeteksi gerakan tangan dan menampilkan objek *Augmented Reality* (AR). Penelitian yang akan dilakukan dengan metode *tracking* dan interaksi terbaru pada objek *Augmented Reality* yang membuat pengguna dan objek *virtual* berinteraksi di dunia nyata. Pada penelitian ini akan dikembangkan sistem *Augmented Reality* menggunakan tangan kanan sebagai navigasi dari objek *augmented reality*.

Berdasarkan penjelasan diatas, masalah pada *Augmented reality* (AR) adalah bagaimana *user* dapat melakukan interaksi dengan objek AR secara natural. Teknik *Augmented reality* memiliki keterbatasan untuk mendapatkan informasi 3D pada objek virtual dan pengguna yang terlibat dalam interaksi. Berdasarkan penjelasan sebelumnya maka diusulkan judul “**Interaksi Hand Tracking pada Objek Augmented Reality Menggunakan Stereo Vision**” pada tugas akhir ini.

1.2. Rumusan dan Batasan Masalah

Rumusan masalah yang dihadapi pada penelitian ini ialah bagaimana melakukan proses *Hand Tracking* dapat terdeteksi oleh kamera Minoru 3D dan ketika tangan telah terdeteksi maka objek AR akan muncul, kemudian objek AR akan bergerak sesuai perintah navigasi tangan.

Adapun batasan masalah untuk menyelesaikan perumusan masalah tersebut, adalah:

1. Kamera yang digunakan adalah kamera Minoru 3D yaitu dua kamera dalam 1 *port* USB yang di koneksikan pada PC.
2. Tugas akhir ini tidak membahas pemodelan tangan secara 3D
3. Metode yang digunakan dalam melakukan *Hand Tracking* adalah segmentasi warna kulit dan penentuan fitur ujung jari.
4. Perancangan di fokuskan pada proses pendeteksian *Hand Tracking* dan interaksinya pada objek *Augmented Reality*.
5. Lingkungan pengujian didalam ruangan dengan kondisi cahaya terang, sedang dan minim cahaya

6. Perhitungan jarak antara ujung jari dengan titik pusat menggunakan *Euclidean Distance*
7. Hanya menggunakan tangan kanan untuk melakukan navigasi objek *Augmented Reality*

1.3. Tujuan dan Manfaat

1.3.1. Tujuan

Tujuan yang hendak dicapai oleh penulis pada penelitian ini adalah:

1. Membuat sistem segmentasi warna kulit dan deteksi ujung jari untuk dapat melakukan *tracking*.
2. Melihat tingkat keberhasilan objek *Augmented Reality* (AR) mengikuti navigasi pergerakan tangan.

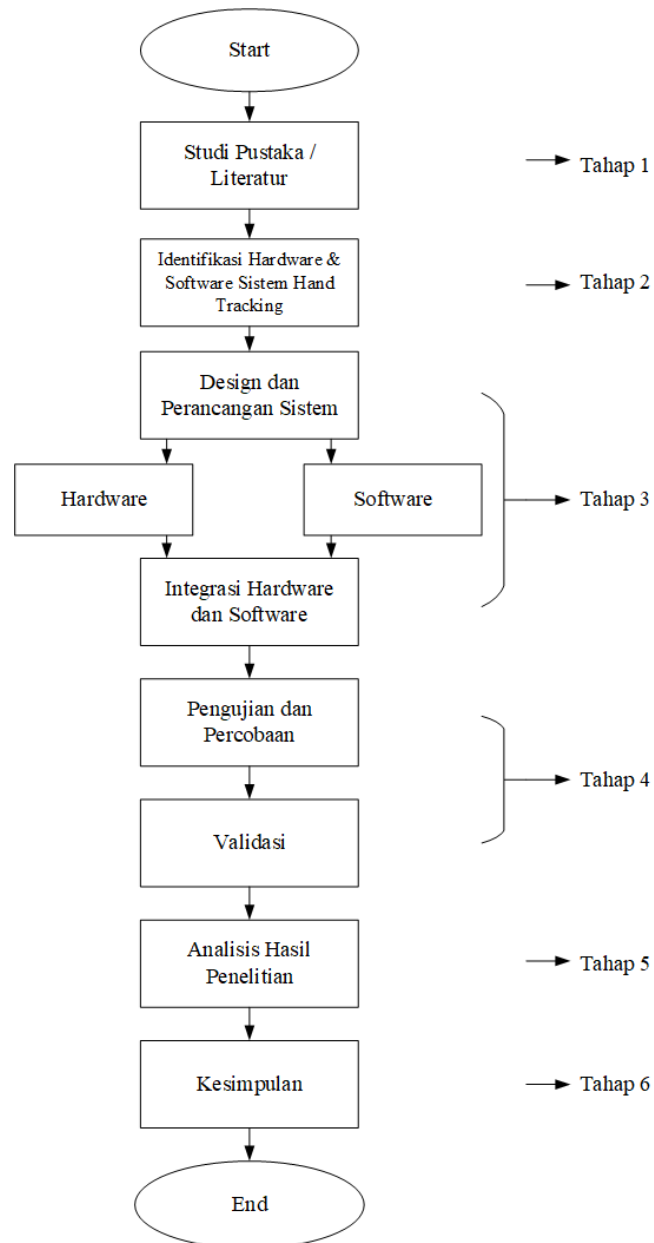
1.3.2. Manfaat

Manfaat dari penelitian ini adalah:

1. Sistem yang dibuat dapat membantu *user* berinteraksi langsung menggunakan tangan pada objek AR tanpa alat bantu.
2. Sebagai pelengkap sensor *leap motion*.

1.4. Metodologi Penelitian

Metodologi yang akan digunakan pada tugas akhir ini terdiri dari tahapan-tahapan yang dilakukan, Berikut ini adalah tahapan-tahapan metodologi dan flowchart yang ditunjukkan pada Gambar 1.1.



Gambar 1.1 *Flowchart* Metodologi Penelitian

1. Tahap Pertama (Studi Pustaka / Literatur)

Tahap ini dilakukan dengan cara mencari dan membaca literatur serta referensi tentang “*Stereo Vision, Hand Recognition dan Augmented Reality*” sehingga menunjang laporan tugas akhir.;

2. Tahap kedua (Perancangan Sistem)

Pada tahap ini dilakukan perancangan sistem yaitu pertama pembuatan *interface* yang akan menampilkan hasil tangkapan kamera ,kemudian melalui beberapa proses yaitu segmentasi warna kulit dan penentuan fitur ujung jari. Sehingga terjadi interaksi dengan objek Augmented Reality.

3. Tahap Ketiga (pengujian)

Tahap ini meliputi pengujian perancangan sistem perangkat lunak dan kamera yang dapat mengikuti gerakan tangan kemudian memastikan objek AR mengikuti navigasi yang di lakukan oleh tangan.

4. Tahap keempat (Analisis)

Tahap ini merupakan hasil dari pengujian yang dilakukan sebelumnya dan dianalisis dengan tujuan mengetahui kekurangan pada hasil perancangan dan penyebabnya selanjutnya bisa dipakai pada penelitian selanjutnya.

5. Tahap Kelima (Penulisan)

Tahap ini merupakan hasil penulisan yang didapatkan selama penelitian yang telah dilakukan.

6. Tahap Keenam (Kesimpulan dan Saran)

Pada tahapan terakhir ini penulis melakukan penarikan kesimpulan dari hasil analisis dan saran mengenai interaksi *Hand Tracking* pada objek *Augmented Reality*.

1.5. Sistematika Penulisan

Untuk lebih memudahkan dalam penyusunan tugas akhir dan memperjelas isi dari setiap bab pada laporan , maka dibuatlah sistematika penulisan sebagai berikut :

BAB I PENDAHULUAN

Bab ini berisi tentang penjabaran secara sistematis topik yang diambil yaitu latar belakang, perumusan dan batasan masalah,tujuan dan manfaat, metodologi penelitian.

BAB II TINJAUAN PUSTAKA

Bab ini berisi tentang dasar teori *Augmented Reality*, *Stereo-Vision*, Segmentasi Warna kulit, *Hand Tracking* pada objek *Augmented Reality*, *EmguCV*.

BAB III METODOLOGI

Bab ini memaparkan secara bertahap dan terperinci tentang langkah-langkah (metodelogi) yang dipakai pada pembuatan kerangka kerja (*framework*) dalam menyelesaikan tugas akhir serta perancangan *software* dan hardware selama penelitian

BAB IV PENGUJIAN DAN ANALISA

Bab ini menjelaskan tentang hasil pengujian yang telah dilakukan pada *hand tracking* menggunakan kamera Minoru 3D dan pengujian *software*, serta analisa dari data yang didapatkan.

BAB V KESIMPULAN

Bab ini berisi tentang kesimpulan yang didapatkan oleh penulis serta merupakan jawaban dari tujuan yang ingin dicapai pada bab 1 (pendahuluan)

BAB II

TINJAUAN PUSTAKA

2.1. Pendahuluan

Pada tugas akhir ini akan dirancang sebuah sistem *hand tracking* pada objek AR menggunakan kamera *stereo vision* dengan menerapkan sistem navigasi menggunakan tangan untuk berinteraksi dengan objek *Augmented Reality*. Dengan menggunakan kamera *stereo vision* maka akan dilakukan pengolahan citra digital sehingga sistem dapat mengenali pola dari gerakan tangan dan menampilkan objek *Augmented Reality* yang dapat mengikuti pola navigasi yang telah ditentukan pada sistem.

Prinsip kerja dari sistem *hand tracking* pada objek *Augmented Reality* menggunakan kamera *stereo vision* yaitu proses dilakukan dengan penangkapan gambar tangan secara *realtime* kemudian kemudian tangan tangan disegmentasi setelah itu akan ditentukan pola isyarat jari tangan untuk interaksi objek *Augmented Reality*. Setelah menentukan pola isyarat jari tangan maka kita akan mendapatkan nilai dari pola isyarat jari tangan.

2.2. Stereo-Vision

Stereo-Vision (penglihatan Ganda) yang sering juga disebut dengan *Binocular Vision* atau *Stereo* merupakan suatu metode yang menggunakan dua buah kamera untuk mengambil gambar atau video kemudian menangkap informasi dari suatu objek dalam pandangan tiga dimensi (3D) berdasarkan disparitas antara sejumlah gambar. Hal ini memungkinkan dilakukan rekonstruksi profil sebuah objek kemudian menentukan lokasinya pada suatu ruang 3D .

Pada penelitian ini akan digunakan perangkat keras yaitu kamera minoru 3D yang dapat di lihat pada gambar 2.1. Fungsi dari dari kamera *stereo vision* pada penelitian ini yaitu terbagi menjadi dua dimana salah satu kamera memiliki fungsi untuk mendeteksi pergerakan tangan dan kamera yang lainnya berfungsi untuk menampilkan objek *Augmented Reality* kamera tersebut merupakan jenis kamera webcam *stereoscopic*, kamera minoru 3D memiliki dua buah kamera

dengan resolusi 800x600 dengan sensor *CMOS*, keduanya merupakan lensa *wide-angle* yang terhubung dengan satu kabel *port USB*.



Gambar 2.1. Kamera Minoru 3D

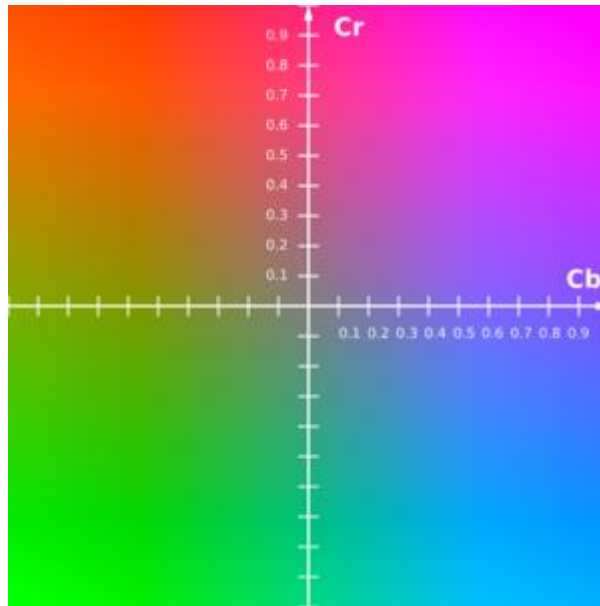
2.3.Hand Tracking pada objek Augmented Reality

Hand Tracking pada objek *Augmented Reality* terdiri dari beberapa tahapan yaitu :

1. Segmentasi warna kulit
2. *Convexity hull defect*
3. Interaksi tangan pada objek *Augmented Reality*

2.3.1.Segmentasi Warna Kulit

Pada segmentasi warna kulit dapat dideteksi menggunakan informasi dari *tone* warna, dimana frame yang di tangkap kamera kemudian mengkonversi nilai RGB ke ruang warna YCrCb. Ruang warna YCrCb memiliki tiga jenis warna yang sering digunakan pada perangkat video. Fungsi dari YCrCb itu sendiri adalah memisahkan komponen *luminance* (pencahayaan) dan komponen warna. Ruang YCrCb akan memisahkan citra kedalam suatu komponen *luminance* dan komponen warna. Informasi *luminance* diinformasikan oleh komponen Y, sedangkan informasi untuk warna dipresentasikan oleh komponen Cb dan Cr [3]. Seperti pada gambar 2.2.



Gambar 2.2. Ruang Warna YCrCb

Berikut ini adalah rumus YCrCb yang digunakan oleh [2]:

$$\left. \begin{aligned}
 Y &= (0,29900 \times R) + (0,58700 \times G) + (0,11400 \times B) \\
 Cr &= (0,50000 \times R) - (0,41869 \times G) - (0,08131 \times B) + 128 \\
 Cb &= (-0,16874 \times R) - (0,33126 \times G) + (0,5000 \times B) + 128
 \end{aligned} \right\} \dots\dots(2.1)$$

2.3.2. Convexity Hull Defect

Convexity adalah suatu ilmu dibidang geometri. Dalam beberapa metode yang dipakai untuk mencari *convexity hull defect* dalam suatu deret poin, salah satu metode yang di gunakan adalah *Graham scan*. Berikut ini terdiri dari proses yang dilakukan dengan menggunakan *Graham scan* [4].

1. Mencari titik dengan posisi paling kiri dan terendah pada deret poin. Pada tahapan ini, deret poin yang akan dicari memiliki posisi paling kiri dan terendah. Poin ini merupakan point awal atau *Set point*. Fungsi berikut digunakan untuk mendapatkan poin paling kiri dan terendah.

$$\begin{aligned}
 Left &= Min (point1.x, point2.x, point3.x, \dots, pointn.x) \\
 Botton &= Max (point1.y, point2.y, point3.y, \dots, pointn.y)
 \end{aligned} \dots\dots(2.2)$$

2. Mengurutkan deret poin searah jarum jam atau lawan arah jarum jam

Point paling kiri dan terendah merupakan *set point*. pengurutan akan dilakukan dengan berdasarkan arah jarum jam (CW) atau berlawanan arah jarum jam (CCW) dengan *set point* menjadi titik pusatnya. Pengurutan putaran dilakukan sebesar 180 derajat. Dimulai dari *pixel* yang terdekat dengan *set point* hingga *pixel* paling jauh untuk tiap besaran sudut. Agar dapat mencari nilai x berdasarkan lawan arah jarum jam (CCW) dengan *set point* paling kiri dapat menggunakan persamaan berikut.

$$Jarak = \begin{cases} j = 0 \text{ sampai } j \leq image.Height - setpoint.y, \text{ sudut} < 90 \\ j = 0 \text{ sampai } j < setpoint.y, \text{ sudut} > 90 \end{cases} \dots\dots (2.3)$$

$$x = (Tan(PI * sudut / 180) * jarak$$

Dengan menggunakan persaaam di atas maka akan didapatkan posisi *pixel* yang di urutkan, dimana y didapatkan dari pengurangan nilai y pada *set point* dengan j pada perulangan.

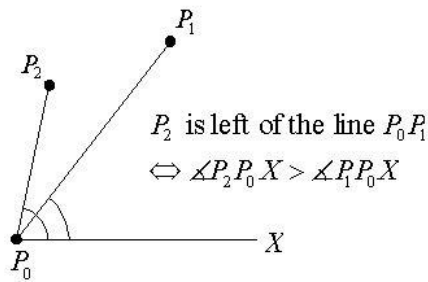
$$y = setpoint.y - j \dots\dots(2.4)$$

3. Menentukan *hull* dan *defect*

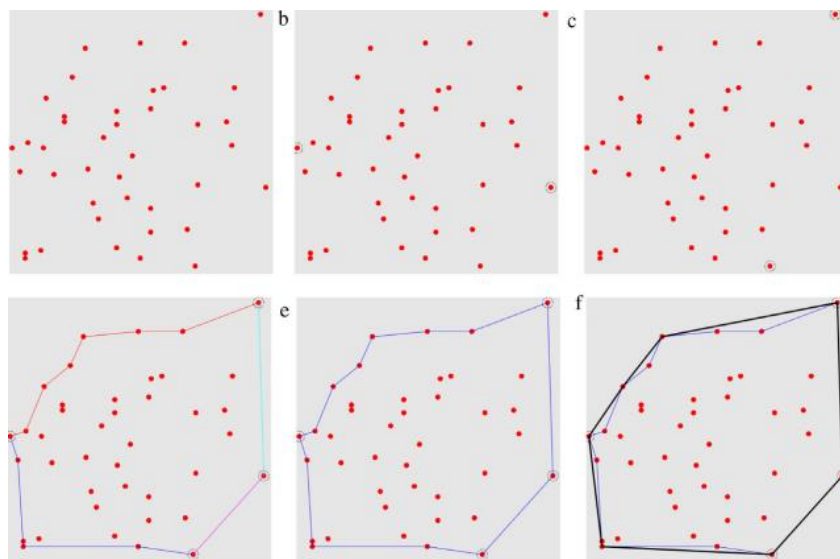
Setelah mendapatkan deret poin yang telah diurutkan maka selanjutnya dapat menentukan *hull* dan *defect* dengan cara mengecek arah dari 3 point pertama. Jika arah 3 poin tersebut berbelok ke kanan (CW) maka point kedua adalah defect berlaku sebaliknya. Untuk dapat menentukan poin tersebut merupakan *hull* atau *defect* dapat menggunakan persamaan berikut ini.

$$CCW = (p2.x - p1.x) * (p3.y - p1.y) - (p2.y - p1.y) * (p3.x - p1.x) \dots\dots(2.5)$$

Apabila nilai CCW > 0 maka point kedua merupakan *hull* dan jika nilai CCW < 0 maka point kedua merupakan *defect*. Pada gambar 2.3. menunjukan arah dari 3 poin untuk mendapatkan *convexity hull defect*.



Gambar 2.3. Arah 3 Poin *Convexity Hull Defect*.



Gambar 2.4. Algoritma Graham Scan[5].

2.3.3. Center of Gravity (COG)

Setelah mendapatkan *convexity hull defect*, maka proses yang akan dilakukan selanjutnya adalah menentukan titik tengah atau *center of Gravity* (COG), COG merupakan suatu metode yang digunakan untuk mendapatkan posisi pusat dari telapak tangan agar dapat mengetahui arah sudut pada jari. Koordinat dari sumbu pusat gravitasi dari X dan Y di jabarkan secara sederhana dengan urutan *moment* $m_{1,0}$ dan $m_{0,1}$ dan dibagi dengan *moment* urutan $m_{0,0}$ seperti persamaan 2.6 [6].

$$\begin{aligned}
x_c &= \frac{m_{1,0}}{A} = \frac{m_{1,0}}{m_{0,0}} \\
y_c &= \frac{m_{0,1}}{A} = \frac{m_{0,1}}{m_{0,0}}
\end{aligned}
\tag{2.6}$$

Keterangan :

- x_c = Nilai titik X dari *Center Of Gravity* (COG)
- y_c = Nilai titik Y dari *Center Of Gravity* (COG)
- $m_{1,0}$ = *Center Of Gravity* (COG) dengan nilai $p=1$ dan $q=0$
- $m_{0,1}$ = *Center Of Gravity* (COG) dengan nilai $p=0$ dan $q=1$
- $m_{0,0}$ = Area *Center Of Gravity* (COG)

dimana x_c dan y_c untuk dapat mencari titik x dan y dari *moment* pusat, $m_{1,0}$ *moment* $p=1$ dan $q=0$, setelah mendapatkan hasil persamaan diatas maka akan didapatkan nilai *center of gravity* dan akan dijadikan nilai acuan untuk menghitung jarak *euclidean distance* pada sistem.

2.3.4. *Euclidean Distance*

Setelah mendapatkan nilai COG maka dapat menghitung jarak COG dengan titik ujung jari yang disebut *euclidean distance*. Jarak *euclidean* atau *euclidean metrix* merupakan jarak antara dua buah nilai, misalkan P dan Q adalah kumpulan dari panjang garis, dimana $P = (p_1, p_2, p_3, \dots, p_n)$ dan $Q = (q_1, q_2, q_3, \dots, q_n)$, maka persamaan *euclidean* dapat ditunjukkan oleh persamaan 2.7 berikut ini [2].

$$d(\mathbf{q}, \mathbf{p}) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \dots + (q_n - p_n)^2} = \sqrt{\sum_{i=1}^n (q_i - p_i)^2}$$

.....(2.7)

Dimana q adalah titik ujung jari (*finger tips*) sedangkan p merupakan titik acuan dari *center of Gravity* (COG). *Euclidean distance* digunakan agar dapat mengidentifikasi jari baik ketika membuka atau menutup.

2.4. *Augmented Reality (AR)*

Augmented Reality adalah teknologi yang dibuat dengan cara penggabungan secara real-time terhadap konten digital dua dimensi atau tiga dim

ensi yang di buat pada komputer yang diproyeksikan pada lingkungan nyata. Bukan secara *Virtual* yang menggantikan lingkungan nyata, namun *Augmented Reality* akan menggabungkan keduanya [8].

Benda-benda *virtual* menampilkan informasi yang tidak dapat diproses secara langsung oleh indera manusia. Oleh karena itu, *Augmented Reality* dapat digunakan untuk membantu pemahaman interaksi antar pengguna dunia nyata. Informasi yang ditampilkan oleh benda maya membantu pengguna melaksanakan kegiatan-kegiatan secara *realtime*. Selain menampilkan objek *virtual* pada lingkungan nyata, *Augmented Reality* juga dapat menghilangkan benda-benda nyata. Dengan menambahkan satu lapisan objek *virtual* agar bisa menyamarkan lingkungan dunia nyata dari penglihatan *user* [8].

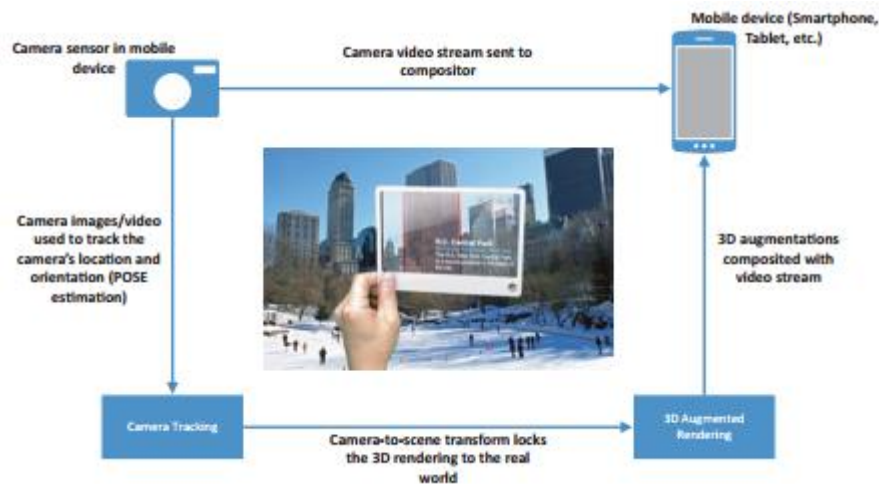
Augmented Reality bisa digunakan pada indera manusia, diantaranya pendengaran, sentuhan, dan hidung. Bukan hanya bisa dipakai pada ilmu kesehatan, militer, industri, *Augmented Reality* dapat diaplikasikan pada perangkat - perangkat yang dipakai oleh orang banyak, seperti pada telpon genggam [6].

2.5. Cara Kerja *Augmented Reality*

Seperti yang ditunjukkan oleh gambar 2.5, cara kerja *Augmented Reality* dalam menampilkan objek virtual pada lingkungan nyata adalah sebagai berikut [9] :

1. kamera merekam video kemudian dikirimkan ke prosesor.
2. Perangkat lunak pada prosesor mengolah video dan mencari suatu pola
3. Perangkat lunak menghitung posisi pola untuk mengetahui lokasi objek *AR* akan diletakan.
4. Perangkat lunak mengenali pola dan akan dicocokkan dengan informasi yang dimiliki *Software*.

5. Objek *AR* akan ditambahkan sesuai dengan hasil pencocokkan informasi dan diletakan pada posisi yang telah dihitung sebelumnya.
6. Objek *virtual* akan melalui perangkat tampilan.



Gambar 2.5. Diagram Sistem Cara Kerja *Augmented Reality* [9]

2.6. Tipe *Augmented Reality*

Augmented Reality terbagi menjadi dua tipe, yaitu :

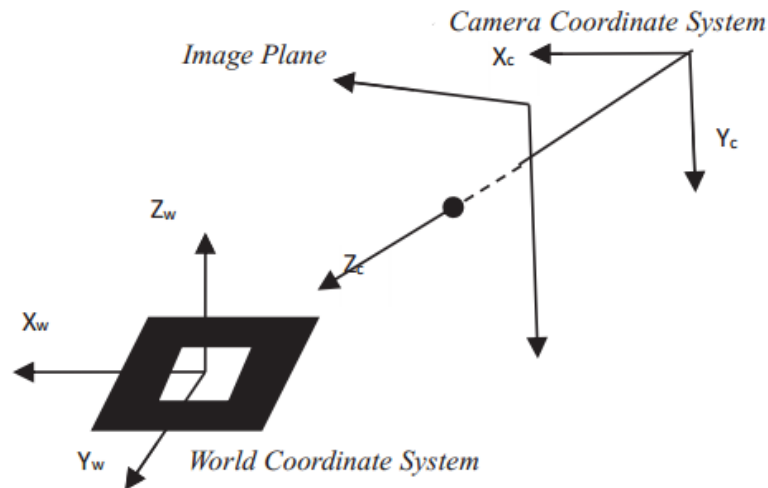
1. *Marker-based Augmented Reality*

Marker-based Augmented Reality, disebut juga pelacakan memakai *marker*, merupakan tipe *AR* yang mengenali pola dari *marker* dan ketika pola tersebut sudah diidentifikasi, maka perangkat lunak akan menampilkan objek virtual diatas permukaan marker. *Marker* adalah putih dan hitam dimana terdapat pola hitam dibagian tengah dan latar belakangnya putih berguna untuk membantu dalam mengidentifikasi *marker* dari latar belakang yang kompleks[7].Berikut ini merupakan contoh marker pada *AR*.



Gambar 2.6. Contoh *Marker* [7].

Pada *marker* terdapat titik koordinat yang berfungsi untuk mengatur posisi dari objek *virtual* yang akan di tambahkan. Posisi objek *virtual* berada tegak lurus dengan *marker*. Dimana objek *virtual* berdiri segaris dengan sumbu Z (lurus keatas) serta tegak lurus dengan sumbu X (sisi kiri dan kanan) dan sumbu Y (sisi depan dan belakang) dari koordinat *marker*. pada gambar 2.9 merupakan ilustrasi dari titik koordinat *virtual marker*.



Gambar 2.7. Titik Koordinat *Virtual* pada *Marker*.

2. *Markerless Augmented Reality*

Markerless Augmented Reality merupakan tipe *Augmented Reality* yang tidak memakai *marker* untuk menambahkan suatu objek *AR* pada lingkungan nyata. *Markerless* di bagi menjadi dua teknik [6], yaitu:

a. *Pose Tracking*

Pose Tracking merupakan suatu teknik yang bekerja dengan cara mengamati lingkungan tidak bergerak (*static*) dengan perangkat keras pada *AR* yang bergerak, teknik *Pose Tracking* diterapkan pada *Global Positioning System* (GPS), keunggulan dari teknik *Pose Tracking* adalah perangkat keras *AR* tidak perlu melakukan adaptasi dengan suatu pola atau *marker*, namun perangkat keras harus memiliki sensor yang mempunyai sensitivitas yang baik hingga bisa menambahkan objek *AR* ke dalam lingkungan nyata.

b. *Pattern Matching*

Pattern Matching memiliki kemiripan dengan tipe *Marker based AR*, namun *marker* diganti dengan gambar biasa. Berbeda dengan teknik *Pose Tracking*,

teknik *Pattern Matching* bekerja dengan caramengamati lingkungan nyata melalui pendeteksian pola dan orientasi gambar dengan perangkat keras *AR* yang statis. Teknik ini dapat mengenali pola apa saja selain *marker*, seperti patung, lukisan, wajah manusia jendela bus dan lain sebagainya.

BAB III

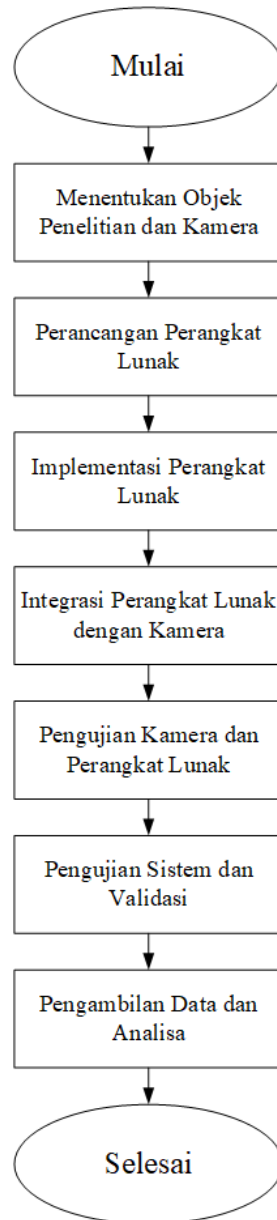
METODELOGI PENELITIAN

3.1. Pendahuluan

Pada bab ini akan dibahas metode yang dipakai untuk membuat kamera dapat mendeteksi gerak objek berupa tangan kemudian menampilkan objek *Augmented Reality*. Untuk membangun sistem software ini hal pertama yang dapat dikerjakan adalah mengkoneksikan kamera minoru 3D pada laptop kemudian menggunakan metode segmentasi warna kulit dimana frame yang ditangkap oleh kamera dikonversi dari nilai RGB ke ruang warna YCbCr. Kemudian masuk ke tahap metode penentuan fitur ujung jari dimana penentuan fitur menggunakan *Contour* tersegmentasi dari hasil segmentasi warna kulit untuk mengidentifikasi tangan, kemudian melakukan interaksi tangan pada objek AR dimana interaksi terjadi di dasarkan pada koordinat ujung jari yang di dapatkan dari proses sebelumnya. Koordinat tersebut kemudian disesuaikan pada reaksi yang dipilih oleh pengguna.

3.2. Kerangka Kerja

Perancangan pada interaksi *hand tracking* pada objek *Augmented Reality* dengan menggunakan kamera *stereo vision* pada tugas akhir kali ini melalui beberapa proses tahapan. Tahapan ini dilakukan dengan mengikuti kerangka kerja (*frame work*) yang telah dibuat sebelumnya agar dalam pembuatan tugas akhir ini akan terstruktur dan mengikuti alur. Gambar 3.1 merupakan bentuk dari kerangka kerja yang akan dilakukan.



Gambar 3.1 Kerangka Kerja (*Framework*)

Perangkat lunak interaksi hand tracking pada objek augmented reality dirancang menggunakan Laptop dengan spesifikasi sebagai berikut:

Operating System : *Windows 7 Ultimate*

Processor : *Intel Core i3*

Memory : 4 GB

Software : *Visual Studio 2010*

3.3. Menentukan Objek Penelitian dan Kamera

Pada penelitian ini menentukan objek yang akan diteliti yaitu *Hand Tracking* pada objek *Augmented Reality*, objek AR berupa bola sederhana dimana ketika kamera Minoru 3D berhasil mendeteksi tangan, maka objek AR otomatis akan muncul dan mengikuti arah gerakan tangan sehingga terjadilah sebuah interaksi.

Perangkat keras yang dipakai pada penelitian kali ini adalah kamera Minoru 3D(kamera *stereo vision*), kamera Minoru 3D merupakan dua kamera yang terbungung dengan satu port USB, masing-masing dari kamera tersebut memiliki fungsi yang berbeda, dimana salah satu kamera memiliki fungsi untuk mendeteksi pergerakan tangan dan kamera yang lainnya berfungsi untuk menampilkan *objek Augmented Reality*



Gambar 3.2 Kamera Minoru 3D

Webcam Type : *Stereoscopic*

Resolution : 800x600

Sensor : *CMOS*

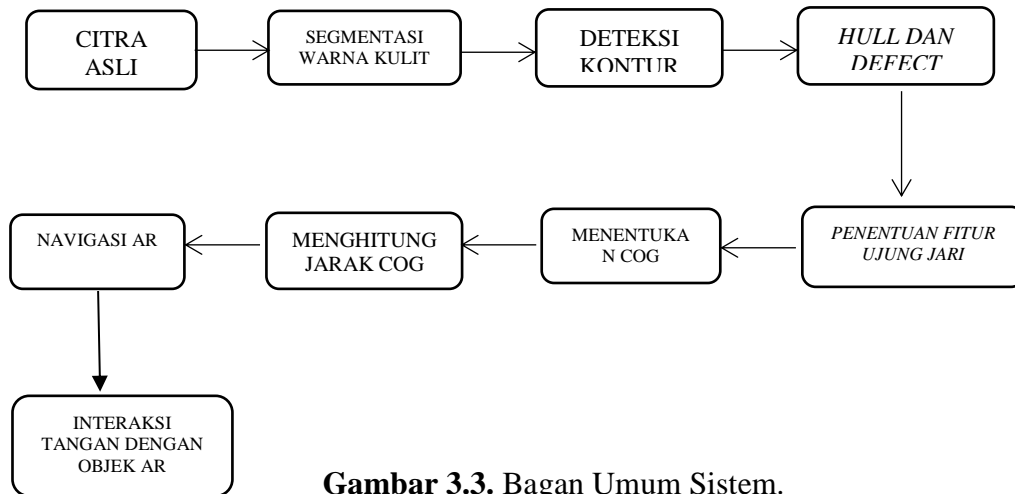
Interface : Serial USB

3.4. Perancangan Perangkat Lunak

Perancangan perangkat lunak pada sistem interaksi *Hand Tracking* pada objek *Augmented Reality* terdiri dari beberapa tahapan dan proses yang dilakukan agar sistem yang di inginkan dapat tercapai.

3.4.1. Perancangan Sistem Perangkat Lunak

Dalam perancangan sistem perangkat lunak pada sistem interaksi *Hand Tracking* pada *Objek Augmented Reality* terdiri dari beberapa tahapan dan proses yang harus dilakukan. Perancangan sistem pada penelitian dapat dilihat pada bagan umum sistem pada gambar 3.3.



Gambar 3.3. Bagan Umum Sistem.

Tahapan pertama yang harus dilakukan adalah melakukan inisialisasi agar kamera dapat terintegrasi dengan PC. Agar kamera dapat terintegrasi dengan PC yaitu dengan cara menginstall aplikasi *stereo3dinstaller* yang berfungsi agar PC dapat menampilkan gambar yang di tangkap oleh kamera minoru 3D. Setelah kamera Minoru 3D dapat terhubung ke PC dengan *output* berupa gambar yang akan di tampilkan pada *interface* pada *software* yang sudah dirancang. Kemudian gambar tersebut akan di proses melalui *image processing* menggunakan metode segmentasi warna kulit, penentuan titik fitur ujung jari dan *Augmented Reality*. Cara kerja dari metode segmentasi warna kulit adalah dengan mengkonversikan nilai RGB (*red, green, blue*) ke ruang warna YCbCr dari *frame* yang ditangkap oleh kamera, pada saat diimplementasikan metode ini dilakukan dengan memisahkan objek-objek yang tidak ingin dideteksi dengan objek yang ingin dideteksi, objek yang tidak ingin dideteksi akan disegmentasi atau digelapkan, sebaliknya objek yang akan di deteksi akan berwarna putih. Setelah dilakukan segmentasi warna kulit, kemudian kita masuk ketahap penentuan titik fitur ujung jari, pada saat penentuan fitur menggunakan *contour* yang telah tersegmentasi dari hasil segmentasi warna kulit untuk mengidentifikasi tangan. Dengan menentukan kontur terbesar dalam *output* gambar. Kemudian pada tahap akhir yaitu interaksi

tangan dengan objek *AR* dimana setelah koordinat tertinggi dari penentuan titik fitur ujung jari telah dilakukan, maka objek *augmented reality* akan melakukan interaksi dimana ketika tangan berhasil dideteksi maka objek *AR* akan muncul dan mengikuti navigasi dari tangan user.

3.4.2. Citra Input (Citra Asli)

Pada penelitian ini citra inputan berupa tangan yang ditangkap secara *realtime* oleh kamera Minoru 3D dengan resolusi 800x600 *pixel*. Kemudian citra inputan akan diproses oleh *software*.

3.4.3. Image Processing

Dalam pembuatan sistem interaksi *hand tracking* pada objek *AR*. *Image Processing* sangat diperlukan agar citra asli dapat ditransformasikan menjadi sistem yang dapat mendeteksi objek tertentu dan beberapa tahapan yang harus dilakukan agar sistem ini dapat terwujud adalah sebagai berikut.

3.4.3.1. Pengenalan Tangan

Pada tahap pengenalan tangan dilakukan proses segmentasi warna kulit yang terbagi menjadi tiga bagian yaitu konversi citra RGB ke ruang warna YCbCr, deteksi warna kulit dan pemilihan kontur terbesar dari warna kulit yang sudah dideteksi.

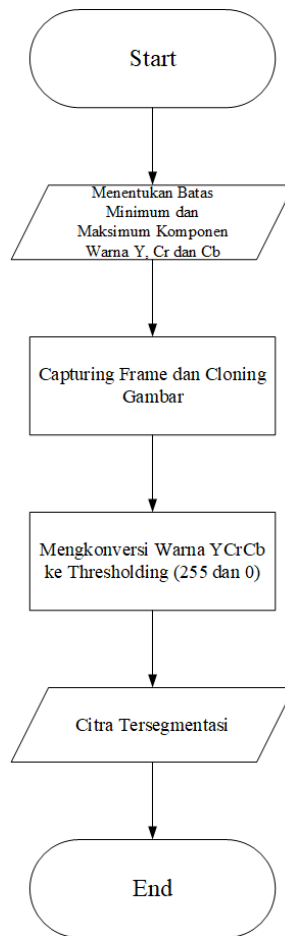
3.4.3.1.1. Konversi Ruang Warna

RGB merupakan warna-warna dasar yang terdapat pada semua warna yang ada dalam citra yaitu merah, hijau, biru. Dari warna RGB-lah tercipta warna-warna yang sering kita temui. Untuk dapat menghitung rata - rata warna kulit ketiga warna dasar tersebut akan diubah menjadi ruang warna berbeda yaitu YCbCr.

3.4.3.1.2. Deteksi Warna Kulit

Untuk pendeteksian warna kulit akan dilakukan setelah citra yang telah dikonversi ke ruang warna YCbCr dengan *output* berupa citra biner dimana 0 merupakan warna yang tidak diinginkan dan tidak termasuk dalam rentang warna

kulit sedangkan 1 merupakan warna rentang kulit. Adapun flowchart deteksi warna kulit ditunjukkan pada Gambar 3.4.



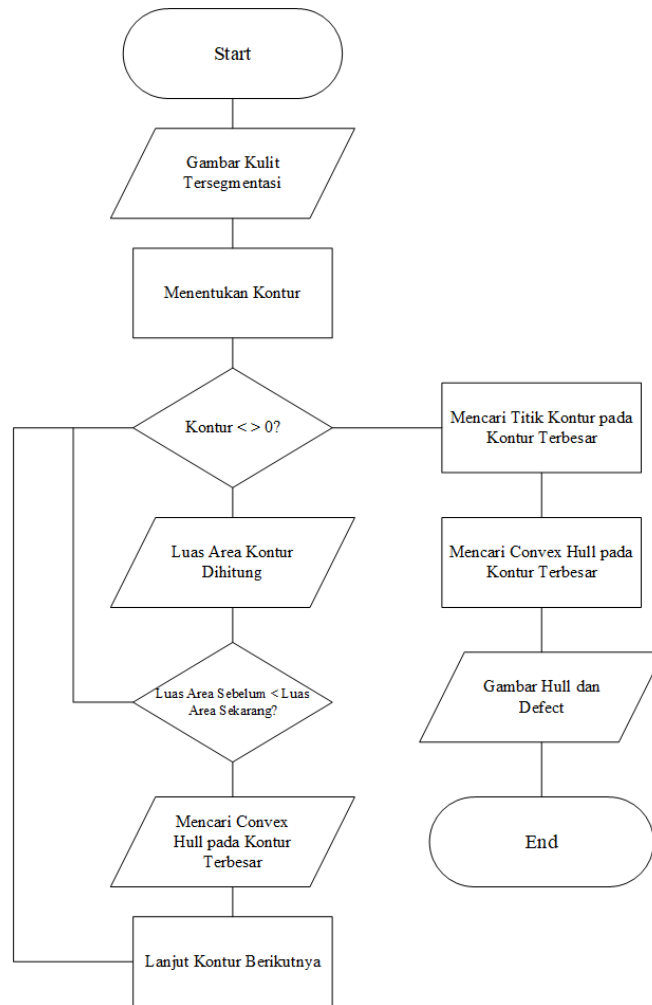
Gambar 3.4. *Flowchart* Segmentasi Warna Kulit.

3.4.3.1.3. Penentuan Titik Fitur Ujung Jari

Untuk proses penentuan titik fitur ujung jari secara umum adalah sebagai berikut:

1. Penentuan *Convex hull*
2. Menemukan *convexity defect* pada *hull*
3. *Filtering* fitur ujung jari pada *convexity hull*
4. Pencarian fitur ujung jari.

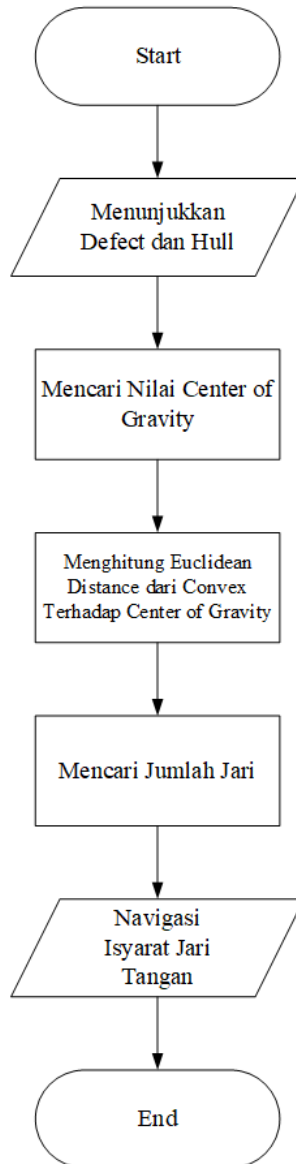
Berikut ini adalah *flowchart* dari *hand tracking* setelah melalui proses segmentasi warna kulit ditunjukkan pada Gambar 3.5.



Gambar 3.5 *Flowchart Hand Tracking*

3.4.4. Navigasi Objek *Augmented Reality*

Tahap berikut ini adalah pembuatan program agar objek *augmented reality* dapat mengikuti perintah dari gerakan tangan user dapat di lihat pada gambar 3.6.



Gambar 3.6. *Flowchart* Navigasi Objek AR

Agar dapat berkerja dengan baik maka citra pola jari tangan sebagai inputan agar objek AR dapat bekerja harus dapat memenuhi pola isyarat menggunakan jari tangan yang dibuat pada program. Pada Tabel 1 merupakan syarat yang akan di gunakan sebagai inputan citra pola jari tangan dari *user* dan perintah agar untuk navigasi objek AR.

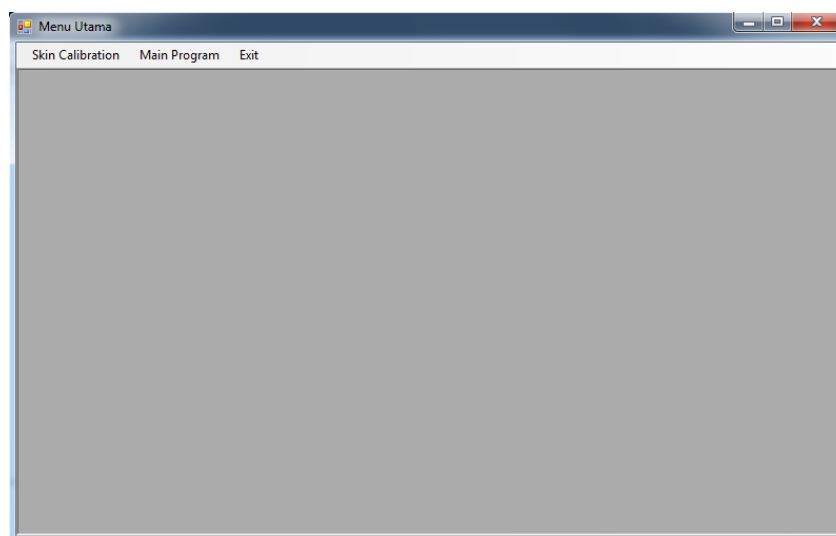
TABEL 1

Fungsi Inputan Citra Pola Isyarat Menggunakan Jari Tangan

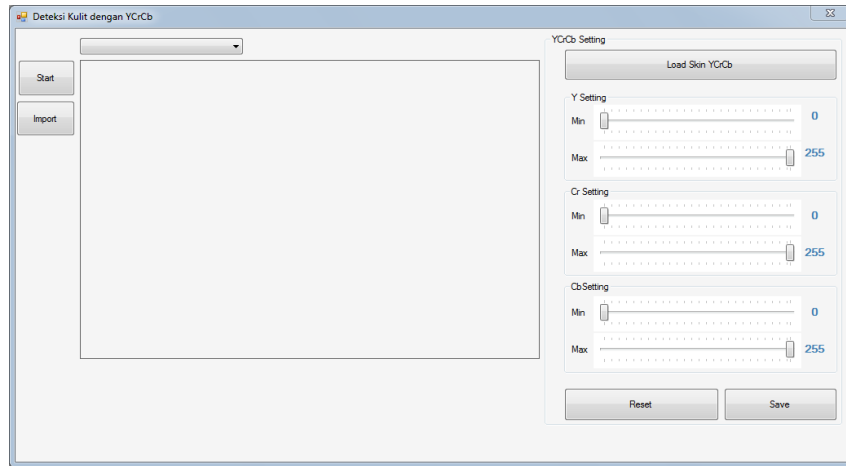
No	Pola Isyarat Jari Tangan	Navigasi Objek AR
1	Telunjuk	- : mengecil
2	Telunjuk dan tengah	+ : membesar
3	Kepalan jari tangan	D! : tidak terdeteksi
4	Jempol, Telunjuk, Tengah, Manis, Kelingking	D : terdeteksi

3.4.5. Perancangan *Interface*

Untuk membangun sistem interaksi *Hand Tracaking* pada objek *Aungmented Reality* maka dibutuhkan suatu *interface* yang bisa menampilkan pengambilan gambar tiap *frame-frame* yang dilakukan oleh kamera *stereo vision*. Pada *interface* kali ini terdiri dari 3 tampilan, tampilan pertama adalah menu utama di mana terdapat 3 pilihan menu program yaitu *skin Calibration*, *Main program* dan *exit*, pada tampilan kedua merupakan isi dari *skin calibration* yaitu menu deteksi kulit dengan *YCrCb* yang berfungsi untuk mengubah cita asli menjadi jadi biner, sedangkan untuk tampilan ketiga berfungsi menampilkan proses *hand Tracking*. Pada tampilan *interface* kali ini masing-masing memiliki resolusi 1264x768 *pixel*. Pada proses perancangan *interface* kali ini menggunakan aplikasi *Microsoft Visual Studio 2010*.

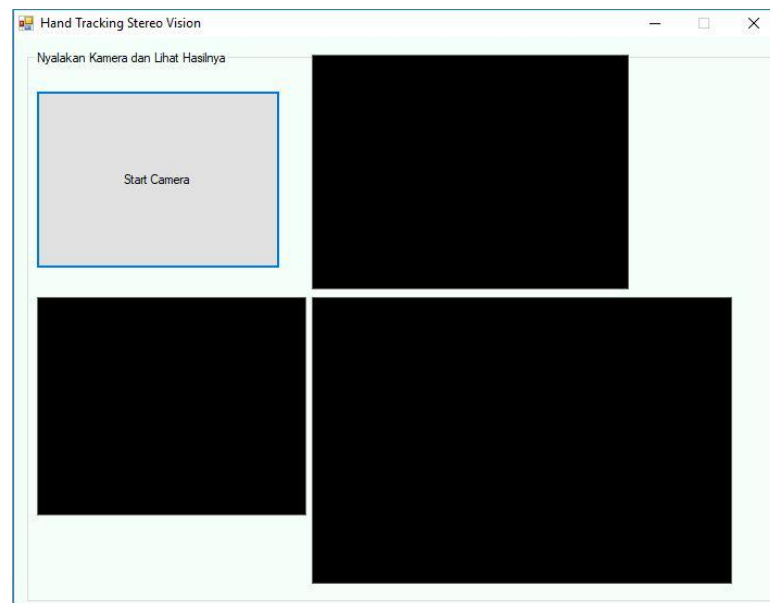


Gambar 3.7 *Interface* Sistem 1



Gambar 3.8 *Interface Sistem 2*

Tahapan-tahapan yang dilakukan pada sistem deteksi kulit dengan YCrCb adalah memilih kamera 1,2,3 kemudian klik *Start* dan citra asli akan tampil, untuk dapat mengubah Citra asli berupa tangan kemudian dirubah ke biner adalah dengan cara *YCrCb Setting* yaitu dengan mengatur Min dan Max pada masing-masing pilihan YCrCb Setting, setelah selesai maka citra gambar akan menjadi biner kemudian klik tombol save untuk menyimpan pengaturan YCrCb. Tombol load di gunakan keika kita ingin membuka kembali interface kemudian tidak harus melakukan YCrCb setting karna sudah di simpan pada pengaturan sebelumnya.



Gambar 3.9 *Interface Sistem 3*

Setelah melakukan YCrCb setting, tahapan selanjutnya ialah *hand Tracking* pada objek *augmented reality*, yaitu dengan mengklik *start* kamera kemudian hand tracking pada objek *augmented reality* dapat dilakukan.

BAB IV

PENGUJIAN DAN ANALISA

4.1. Pendahuluan

Pada bab ini akan dibahas pengujian dan analisis dari hasil pengujian sistem *hand tracking* pada objek *augmented reality* yang telah dirancang. Pengujian yang akan dilakukan meliputi pengujian *hardware* dan *software*. Pengujian dilakukan agar dapat mengetahui kelebihan dan kekurangan sistem yang telah dirancang. kemudian hasil pengujian tersebut akan dianalisis sehingga dapat melakukan pengembangan dalam membangun sistem *hand tracking*.

4.2. Pengujian Kamera Minoru3D (*Stereo Vision*)

Kamera minoru 3D digunakan untuk mendeteksi tangan melalui *image processing* sebagai proses dan pendeteksian tangan. Pada pengujian kamera Minoru 3D dilakukan dengan cara kamera dapat terhubung dengan interface yang telah dirancang sebelumnya.



Gambar 4.1. Kamera Minoru 3D

4.3. Pengujian Perangkat Lunak

perangkat lunak dari sistem *hand tracking* yaitu dengan merancang interface yang terhubung pada kamera minoru 3D, dalam pengujiannya terdiri dari proses-proses yang akan dilalui, yaitu sebagai berikut :

4.3.1. *Image Processing*

Image processing terjadi pada *interface* yang telah di rancang, proses dari image processing sendiri ialah mengolah citra tangan yang di dapatkan oleh kamera kemudian dilakukan segmentasi warna kulit menggunakan metode YCrCb sehingga tangan dapat dideteksi dan kamera dapat melakukan *hand tracking*. Berikut ini adalah proses dari pengolahan citra dari segmentasi warna kulit.

4.3.1.1. Pengambilan Gambar Citra Tangan

Pengambilan gambar terjadi ketika kita menghubungkan kamera dengan pc yang sudah terintegrasi dengan interface yang telah kita buat. Resolusi yang digunakan pada sistem ini memiliki ukuran 800 X 600 *pixel*. Berikut ini merupakan gambar sudah *dicapture* oleh kamera dan terhubung pada *interface*

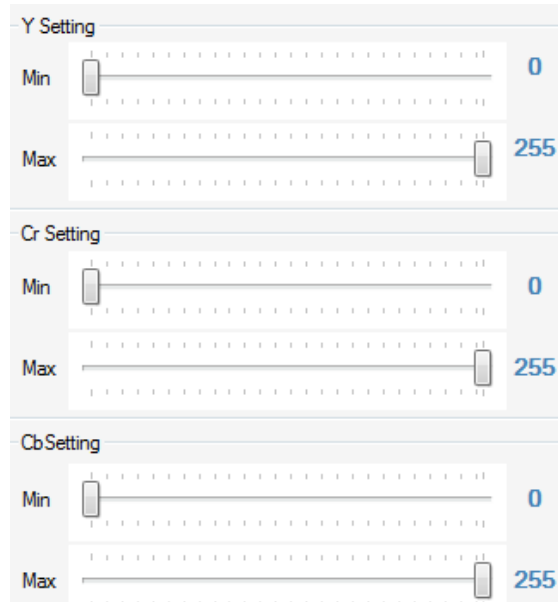


Gambar 4.2. Pengambilan Gambar Citra Tangan.

4.3.1.2. Ruang Warna YCrCb

Ruang warna YCrCb akan digunakan untuk memisahkan citra kedalam suatu komponen *luminance* dan komponen warna. Untuk dapat mendeteksi tangan maka kita harus menyesuaikan warna kulit dengan rentang warna YCrCb. Rentang warna YCrCb merupakan nilai minimum dan maksimum YCrCb objek tersebut. Nilai minimum dan nilai maksimum YCrCb di bagi menjadi 3 unsur

yaitu Y,Cr, dan Cb. Nilai minimum Y,Cr dan Cb yaitu 0 dan nilai maksimumnya 255. Dapat dilihat pada gambar 4.3 berikut ini.



Gambar 4.3. Nilai Ruang Warna YCrCb

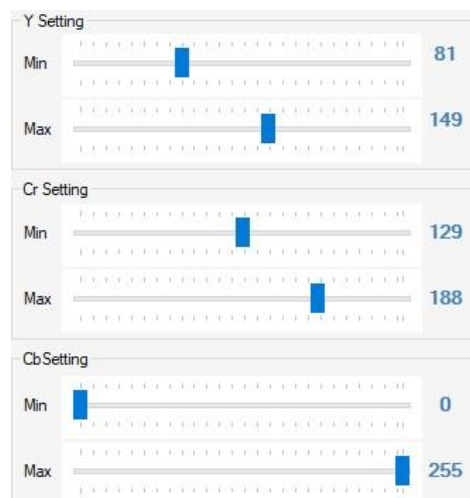
4.3.1.3. Segmentasi Warna Kulit

Segmentasi warna kulit adalah proses pengolahan citra dimana segmentasi warna kulit dapat dideteksi menggunakan *tone* warna dimana frame yang akan ditangkap oleh kamera , kemudian dikonversi dari nilai RGB ke ruang warna YCrCb. Untuk dapat melakukan segmentasi warna kulit tersebut ditentukan dengan rentang warna YCrCb, pengujian yang akan dilakukan adalah dengan menetapkan nilai rentang dari Y,Cr, dan Cb supaya didapatkan nilai yang sesuai dengan warna kulit. Berikut ini adalah hasil dari pengujian segmentasi warna kulit.



Gambar 4.4. Segmentasi Warna Kulit

Untuk mengsegmentasikan supaya dapat menampilkan objek tangan saja, maka diberikan rentang nilai pada ruang YCrCb. Segmentasi warna kulit disesuaikan dengan nilai YCrCb-nya. Untuk objek tangan seperti gambar 4.4 mempunyai nilai rentang sebagai berikut.



Gambar 4.5. Rentang Nilai YCrCb Segmentasi Warna Kulit.

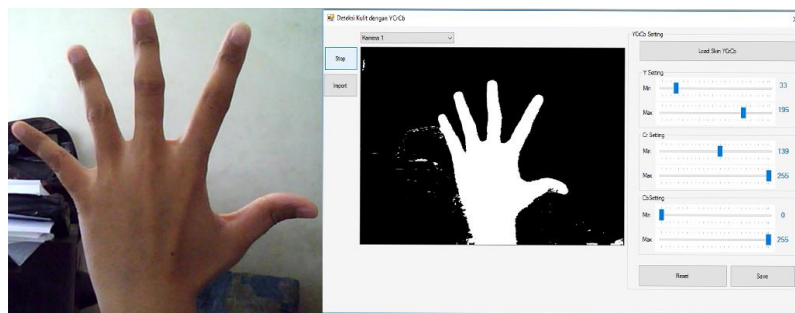
Dalam proses segmentasi warna kulit menggunakan metode YCrCb kita harus menetapkan rentang nilai YCrCb-nya yaitu nilai minimum YCrCb dan nilai maksimum YCrCb tersebut. Pada pengujian ini dilakukan menggunakan *interface* yang telah dirancang dan akan ditampilkan output proses dari PC dan perhitungan manualnya. Proses pengujian segmentasi warna kulit dilakukan pada

2 sample warna kulit dan 3 kondisi cahaya yang berbeda. Berikut ini hasil pengujian dari proses segmentasi warna kulit yang menggunakan ruang warna YCrCb.

4.3.2. Pengujian Warna Kulit

Pengujian yang dilakukan kali ini adalah mendeteksi tangan dengan kondisi warna kulit berbeda - beda meliputi kulit putih dan kulit gelap. Untuk dapat melakukan deteksi hal - hal yang dapat dilakukan adalah menentukan rentang antara YCrCb minimum dan maksimum.

1. Percobaan pertama



Gambar 4.6. Percobaan Pertama Kulit Putih.

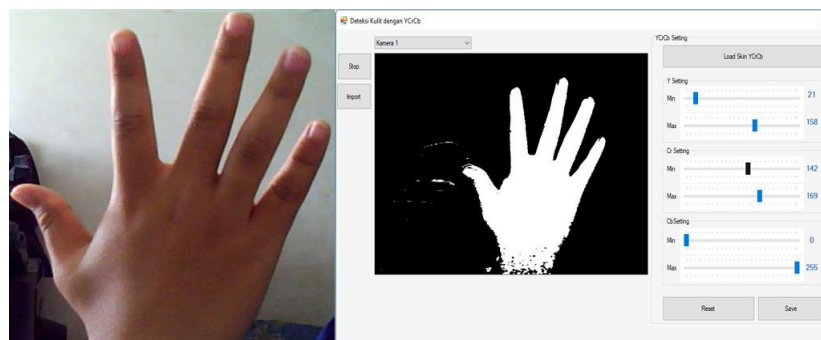
Pada percobaan pertama segmentasi warna kulit, objek tangan dengan kondisi warna kulit putih dapat disegmentasi dan terdeteksi dengan baik oleh sistem. Ruang warna YCrCb yang diberikan untuk dapat mengsegmentasikan objek yaitu, pada nilai Y minimal 33 dan Y maksimal 195, nilai Cr minimal 139 dan Cr maksimal 255, nilai Cb minimal 0 dan Cb maksimal 255.

Tabel 2

Hasil Percobaan Pertama

Pengujian	Rentang YCrCb	Y	Cr	Cb
Interface	Minimum	33	139	0
	Maksimum	195	255	255

2. Percobaan Kedua



Gambar 4.7. Percobaan Kedua Kulit Sedang.

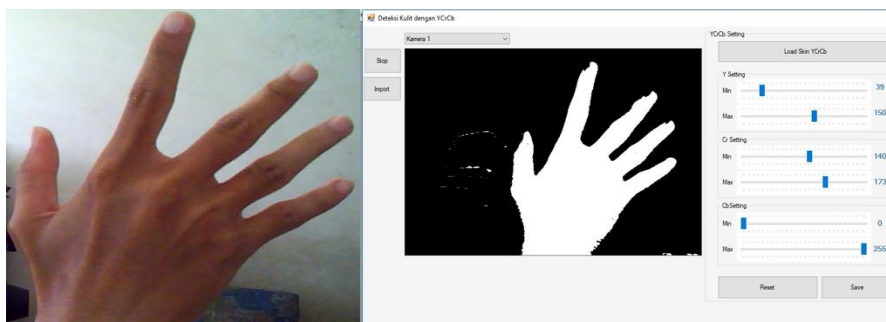
Pada percobaan kedua segmentasi warna kulit, objek tangan dengan kondisi warna kulit sedang juga dapat disegmentasi dan terdeteksi dengan baik oleh sistem. Ruang warna YCrCb yang diberikan untuk dapat mengsegmentasikan objek yaitu, pada nilai Y minimal 21 dan Y maksimal 158, nilai Cr minimal 142 dan Cr maksimal 169, nilai Cb minimal 0 dan Cb maksimal 255.

Tabel 3

Hasil Percobaan Kedua

Pengujian	Rentang YCrCb	Y	Cr	Cb
Interface	Minimum	21	142	0
	Maksimum	158	169	255

3. Percobaan Ketiga



Gambar 4.8. Percobaan Ketiga Kulit Gelap.

Pada percobaan ketiga segmentasi warna kulit, objek tangan dengan kondisi warna kulit gelap juga dapat disegmentasi dan terdeteksi dengan baik oleh sistem. Ruang warna YCrCb yang diberikan untuk dapat mengsegmentasikan objek yaitu, pada nilai Y minimal 39 dan Y maksimal 150, nilai Cr minimal 140 dan Cr maksimal 143, nilai Cb minimal 0 dan Cb maksimal 255.

Tabel 4

Hasil Percobaan Ketiga

Pengujian	Rentang YCrCb	Y	Cr	Cb
Interface	Minimum	39	140	0
	Maksimum	150	173	255

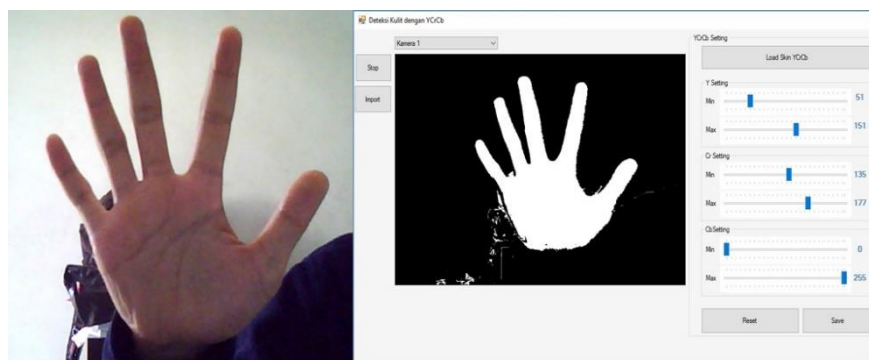
Pada pengujian segmentasi warna berdasarkan jenis warna kulit yang berbeda menggunakan ruang warna YCrCb yang dilakukan sebanyak 3 kali, maka didapatkan adanya perbedaan nilai minimum dan maksimum pada YCrCb dimana kulit putih nilai minimum dan maksimum Y cenderung lebih tinggi dengan nilai min = 33 dan max = 195 dibandingkan dengan nilai minimum dan maksimum Y pada kulit sedang dan gelap cenderung lebih rendah dengan nilai min = 21 hingga 39 dan max = 150 hingga 158, sedangkan nilai minimum Cr pada kulit putih lebih rendah atau dengan nilai min = 139 akan tetapi untuk nilai Cr maksimum memiliki nilai lebih tinggi yaitu max = 255 dibandingkan dengan nilai minimum dan maksimum pada kulit sedang dan gelap dengan nilai min = 140 hingga 142 dan nilai max 169 hingga 173, dan untuk nilai Cb minimum dan maksimum dari ketiga warna kulit memiliki nilai yang sama yaitu min = 0 dan max = 255.

4.3.2.1. Pengujian Berdasarkan Kondisi Cahaya

Pengujian berdasarkan kondisi cahaya masih menggunakan media yang sama dengan pengujian warna kulit yaitu mendeteksi tangan. Terdapat banyak hal yang dapat mempengaruhi keberhasilan dalam segmentasi objek diantaranya adalah kondisi cahaya, pada saat pengujian penulis melakukan percobaan pendeksian tangan pada kondisi cahaya yang berbeda-beda didalam ruangan,

pengujian dilakukan sebanyak 3 kali yaitu pada saat kondisi banyak cahaya, kondisi cahaya sedang, dan kondisi minim cahaya. Dari ketiga kondisi pada percobaan tersebut maka nilai dari intensitas cahaya yang masuk akan berbeda pula, karena perbedaan dari intensitas cahaya maka nilai YCrCb yang diberikan akan berbeda-beda ketika mendeteksi objek tangan. Berikut ini adalah hasil yang didapatkan saat pengujian segmentasi warna kulit berdasarkan kondisi cahaya didalam ruangan.

1. Percobaan pertama



Gambar 4.9. Percobaan Pertama Kondisi Cahaya Terang.

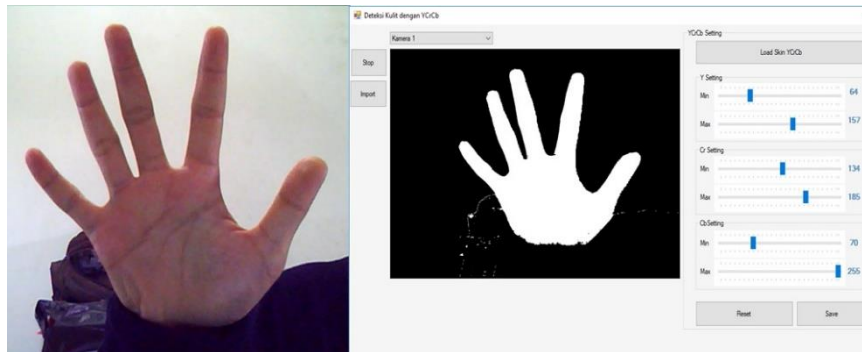
Pengujian ini dilakukan didalam ruangan pada saat siang hari, dengan kondisi cahaya terang bernilai = 150 lx menggunakan aplikasi *lux meter* sehingga didapatkan settingan ruang warna YCrCb dimana Y minimal 51 dan Y maksimal 151, Cr minimal 135 dan Cr maksimal 177, Cb minimal 0 dan Cb maksimal 255.

Tabel 5

Hasil Percobaan Pertama

Pengujian	Rentang YCrCb	Y	Cr	Cb
Interface	Minimum	51	135	0
	maksimum	151	177	255

2. Percobaan kedua



Gambar 4.10. Percobaan Kedua Kondisi Cahaya Sedang.

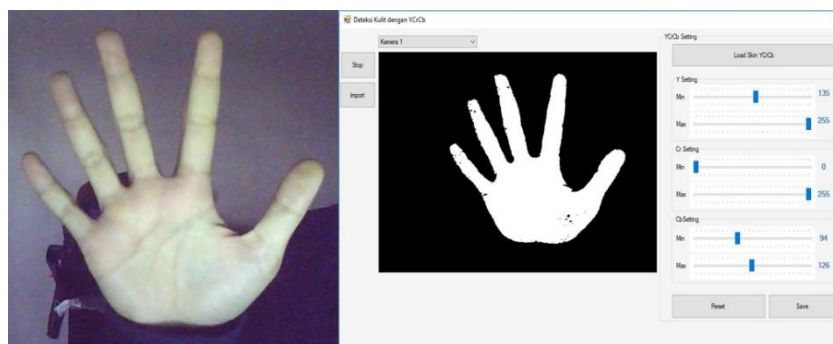
Pengujian ini dilakukan didalam ruangan pada saat siang hari, dengan kondisi cahaya sedang bernilai= 70 lx menggunakan aplikasi *lux meter* sehingga didapatkan settingan ruang warna YCrCb dimana Y minimal 64 dan Y maksimal 157, Cr minimal 34 dan Cr maksimal 185, Cb minimal 0 dan Cb maksimal 255.

Tabel 6

Hasil Percobaan Kedua

Pengujian	Rentang YCrCb	Y	Cr	Cb
Interface	Minimum	64	34	0
	Maksimum	157	185	255

3. Percobaan ketiga



Gambar 4.11. Percobaan Ketiga Kondisi Minim Cahaya.

Pengujian ini dilakukan didalam ruangan pada saat siang hari, dengan kondisi minim cahaya bernilai= 10 lx menggunakan aplikasi *lux meter* sehingga didapatkan settingan ruang warna YCrCb dimana Y minimal 135 dan Y maksimal 255, Cr minimal 0 dan Cr maksimal 255, Cb minimal 94 dan Cb maksimal 126.

Tabel 7

Hasil Percobaan Ketiga

Pengujian	Rentang YCrCb	Y	Cr	Cb
Interface	Minimum	135	0	94
	Maksimum	255	255	126

Pengujian segmentasi warna YCrCb berdasarkan kondisi cahaya didalam ruangan dilakukan sebanyak 3 kali percobaan yaitu pada kondisi cahaya terang, cahaya sedang dan minim cahaya. Dari ketiga percobaan tersebut rentang nilai YCrCb yang didapat memiliki nilai yang berbeda-beda.

Dari hasil pengujian segmentasi warna kulit berdasarkan kondisi cahaya didapatkan kesimpulan sebagai berikut walaupun dari ketiga percobaan tersebut objeknya sama, jika intensitas yang didapatkan berbeda-beda maka nilai minimum dan maksimum dari YCrCb pada objek tersebut akan berubah.

4.3.2.2. Pengujian Konversi Nilai RGB ke YCrCb

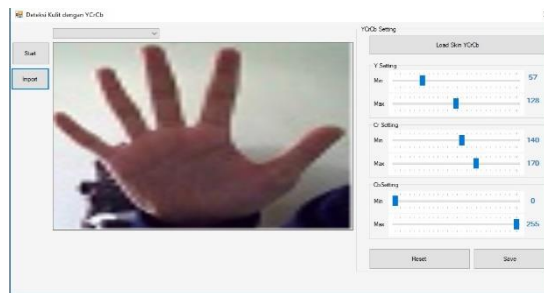
Pada pengujian konversi nilai RGB ke nilai YCrCb kali ini dilakukan pada kondisi cahaya sedang bernila 70 lx dengan settingan ruang warna YCrCb dimana Y minimal 57 dan Y maksimal 128, Cr minimal 140 dan Cr maksimal 170, Cb minilal 0 dan Cb maksimal 255.

Program akan mengambil gambar tangan secara otomatis dengan resolusi 800 X 600 *pixel* setelah penguji menentukan setting nilai min dan max YCrCb, proses selanjutnya adalah melakukan *cropping* gambar secara manual menjadi 617 x 621 *pixel* kemudian dilakukan *resize* menjadi 60 x 61 pixel, proses tersebut dilakukan supaya dapat mempercepat proses *output* data pada komputer.



Gambar 4.12. Citra Tangan

Setelah mendapatkan citra tangan kemudian gambar tersebut diinput kedalam program dengan setting YCrCb yang telah di sesuaikan sebelumnya seperti gambar 4.13 di bawah ini.



Gambar 4.13. Input Citra Tangan

Berikut merupakan tabel RGB yang didapat dari gambar Citra tangan pada gambar 4.13, pada pengujian ini hanya mengambil sebagian data citra untuk mempermudah dalam pengujian data.

Tabel 8

Nilai RGB Pada Citra Gambar Tangan

R	G	B
200	208	197
202	210	199
203	211	200
203	211	200
206	214	203
208	216	205
212	217	210
215	220	213
217	224	216
219	226	218

Setelah mendapatkan nilai RGB, sistem akan mengkonversi nilai RGB menjadi YCrCb. Data yang didapat dari konversi didalam sistem ditunjukkan pada tabel 9.

Tabel 9

Data Ruang Warna YCrCb Hasil Perhitungan Sistem.

Y	Cr	Cb
204	125	124
206	125	124
207	125	124
207	125	124
210	125	124
212	125	124
215	126	125
218	126	125
221	125	125

Dengan menggunakan persamaan 2.1 dapat mencari nilai YCrCb dari sampel nilai RGB, seperti yang ditunjukkan pada perhitungan dibawah ini yang merupakan hasil dari perhitungan secara manual.

$$Y = (0,29900 \times R) + (0,58700 \times G) + (0,11400 \times B)$$

$$Y = (0,29900 \times 200) + (0,58700 \times 208) + (0,11400 \times 197)$$

$$Y = 204,354$$

$$Cr = (0,50000 \times R) - (0,41869 \times G) - (0,08131 \times B) + 128$$

$$Cr = (0,50000 \times 200) - (0,41869 \times 208) - (0,08131 \times 197) + 128$$

$$Cr = 124,8944$$

$$Cb = (-0,16874 \times R) - (0,33126 \times G) + (0,5000 \times B) + 128$$

$$Cb = (-0,16874 \times 200) - (0,33126 \times 208) + (0,5000 \times 197) + 128$$

$$Cb = 123,8499$$

Berdasarkan hasil perhitungan diatas, hasil yang didapat memiliki selisih yang sangat sedikit, sehingga apabila dibulatkan, akan sama dengan nilai konversi yang dihasilkan dari perhitungan sistem. Hasil perhitungan secara manual ditunjukkan pada tabel 10.

Tabel 10

Data Ruang Warna YCrCb Hasil Perhitungan Manual

Y	Cr	Cb
204,354	124,894	123,85
206,354	124,894	123,85
207,354	124,894	123,85
207,354	124,894	123,85
210,354	124,894	123,85
212,354	124,894	123,85
214,707	126,069	125,344
217,707	126,069	125,344
220,995	125,151	125,181

4.3.4. Pengujian Nilai Segmentasi

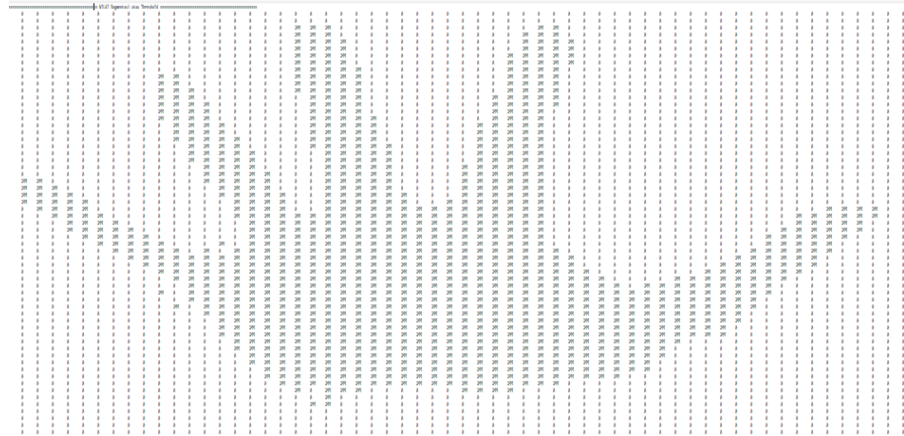
Pada proses Segmentasi kali ini output keluarannya berupa citra biner, yaitu citra yang mempunyai dua nilai tingkat keabuan yaitu hitam dan putih. Pada proses Segmentasi nilai variabel di tentukan secara manual yaitu melalui proses YCrCb setting dengan nilai ambang yang di cari dari masing - masing nilai ditentukan oleh *user*, pada gambar 4.14 merupakan objek tangan yang telah di segmentasi sebelumnya dan pada gambar 4.15 merupakan nilai dari segmentasi warna kulit.



Gambar 4.14. Tangan Yang Tersegmentasi

Pada proses Segmentasi jika nilai min $Y=57$ sampai dengan nilai max $Y=128$ maka objek tangan akan berwarna putih dengan nilai 1, sebaliknya objek tangan

akan berwarna hitam dengan nilai 0 jika dibawah nilai Min Y dan diatas nilai Max Y yang telah ditentukan , proses ini juga berlaku pada nilai Cr dan Cb.



Gambar 4.15. Nilai Segmentasi Tangan

Pada gambar di atas citra berwarna hitam di wakilkkan oleh nilai 0 sedangkan citra putih di wakilkkan dengan nilai 255.

4.3.5. Pola Isyarat Jari Tangan dan Objek *Augmented Reality*

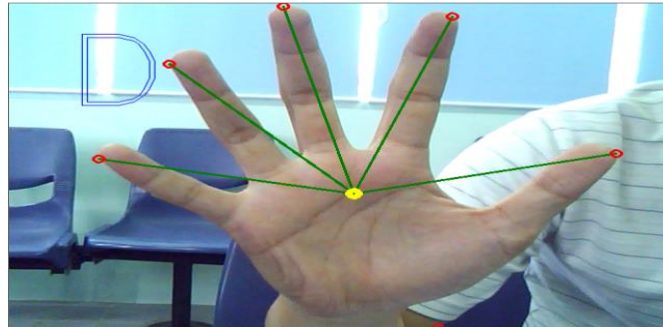
Citra dari pola isyarat jari tangan sebagai imputan navigasi untuk objek AR dapat berfungsi dengan baik jika memenuhi pola isyarat jari tangan yang sudah disesuaikan didalam program. Pada tabel 4.5 merupakan syarat-syarat yang digunakan sebagai inputan citra pada pola isyarat jari tangan pengguna dan juga merupakan perintah navigasi objek *augmented reality*.

TABEL 11

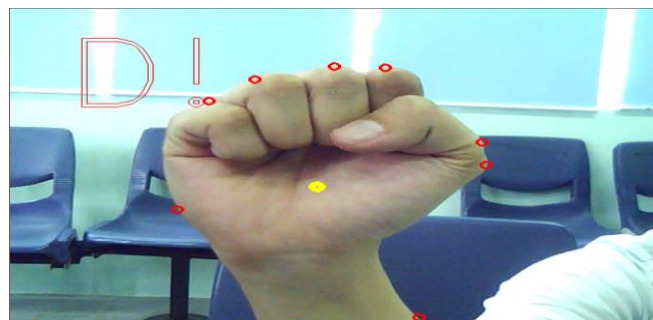
Syarat Inputan Citra Pola Isyarat Jari Tangan

No	Pola isyarat jari tangan yang terdeteksi	Navigasi objek AR
1	Jempol, Telunjuk, Tengah, Manis, Kelingking	Terdeteksi
2	Kepalan tangan	Tidak Terdeteksi
3	Jari telunjuk	Mengecil
4	Jari tengah dan telunjuk	Membesar

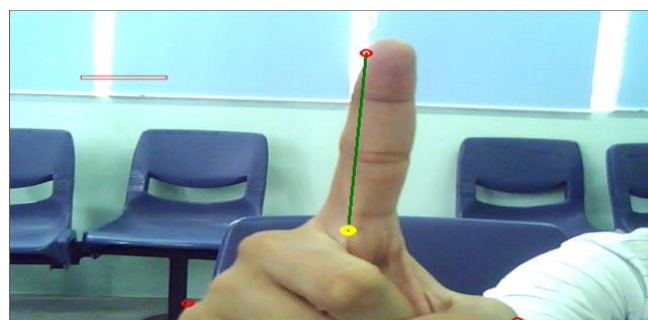
Dari berbagai percobaan yang dilakukan maka didapatkan beberapa contoh dari circa pola isyarat jari tangan yang di menggunakan kamera Minoru 3D yang diletakkan di sudut kanan atas laptop *user* seperti gambar 4.14. berikut ini.



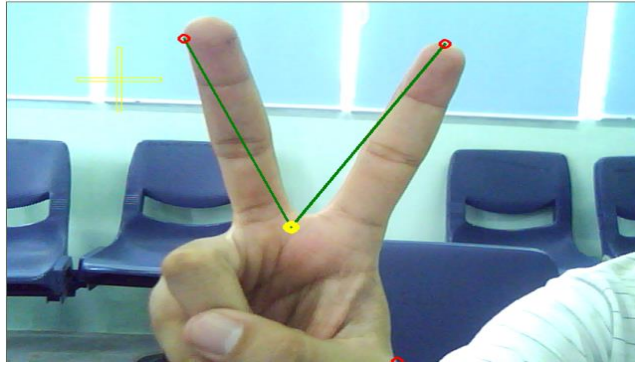
(1)



(2)



(3)

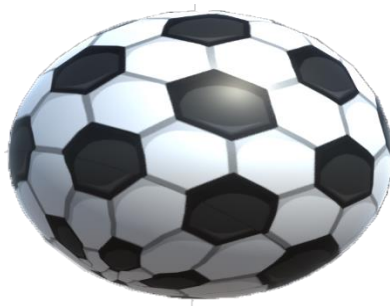


(4)

Gambar 4.15. Citra Navigasi Tangan

- (1) Citra isyarat jari tangan navigasi objek AR terdeteksi
- (2) Citra isyarat jari tangan navigasi objek AR tidak terdeteksi
- (3) Citra isyarat jari tangan navigasi objek AR *Mengecil*
- (4) Citra isyarat jari tangan navigasi objek AR *Membesar*

Objek Augmented Reality yang digunakan pada penelitian kali ini berupa bola sederhana dapat dilihat pada gambar 4.15. , dimana objek AR akan muncul apabila program telah mendeteksi citra isyarat jari tangan yang telah di sesuaikan sebelumnya



Gambar 4.16. Objek Augmented Reality

4.3.5.1. Pengujian Nilai Pada Pola Isyarat Jari Tangan

Pada proses pengujian kali ini akan dilakukan perhitungan jarak *euclidean* diantara titik-titik dari defect dan hull dengan *center of gravity*, yang didapatkan dari kamera *minoru* 3D yang diletakan sudut kanan atas laptop *user* yang kemudian di proses menggunakan program kalibrasi yang telah dibuat sebelumnya yaitu agar dapat mendeteksi warna kulit pada ruang warna YCrCb. Nilai pada pola isyarat jari tangan didapatkan dari tahapan proses yang telah dibahas

sebelumnya dimana kontur terbesar pada jari tangan akan diekstraksikan konturnya agar dapat menghitung momen dari gambar dan dapat menentukan nilai dari *center of gravity*. Untuk dapat mencari nilai dari euclidean distance maka digunakan persamaan 2.5 berikut ini.

$$d(\mathbf{q}, \mathbf{p}) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \dots + (q_n - p_n)^2} = \sqrt{\sum_{i=1}^n (q_i - p_i)^2} \quad \dots\dots 2.6$$

Data euclidian distance pada program

Titik Tengah X 319, Titik Tengah Y : 276

Titik 1 (138,78) =>Jarak dari COG : 268.262930722826

Titik 2 (68,233) =>Jarak dari COG : 254.656631564937

Titik 3 (569,227) =>Jarak dari COG : 254.75674672126

Titik 4 (375,11) =>Jarak dari COG : 270.852358306144

Titik 5 (221,13) =>Jarak dari COG : 280.665281073381

Perhitungan manual euclidean distance :

Diketahui :

➤ Titik 1

q1= 319 p1=276

q2=138 p2=78

Ditanya : jarak COG ?

Jawab :

$$\begin{aligned} d(\mathbf{q}, \mathbf{p}) &= \sqrt{(319 - 138)^2 + (276 - 78)^2} \\ &= \sqrt{32761 + 39204} \\ &= \sqrt{71965} \\ &= 268,2629 \end{aligned}$$

➤ Titik 2

q1= 319 p1=276

q2=68 p2=233

Ditanya : jarak COG ?

Jawab :

$$\begin{aligned}
d(q,p) &= \sqrt{(319 - 68)^2 + (276 - 233)^2} \\
&= \sqrt{63001 + 1849} \\
&= \sqrt{64850} \\
&= 254,6566
\end{aligned}$$

➤ Titik 3

$$q1= 319 \quad p1=276$$

$$q2=68 \quad p2=233$$

Ditanya : jarak COG ?

Jawab :

$$\begin{aligned}
d(q,p) &= \sqrt{(319 - 569)^2 + (276 - 277)^2} \\
&= \sqrt{62500 + 2401} \\
&= \sqrt{64901} \\
&= 254,7567
\end{aligned}$$

➤ Titik 4

$$q1= 319 \quad p1=276$$

$$q2=375 \quad p2=11$$

Ditanya : jarak COG ?

Jawab :

$$\begin{aligned}
d(q,p) &= \sqrt{(319 - 375)^2 + (276 - 11)^2} \\
&= \sqrt{3136 + 70225} \\
&= \sqrt{73361} \\
&= 270,8524
\end{aligned}$$

➤ Titik 5

$$q1= 319 \quad p1=276$$

$$q2= 221 \quad p2=13$$

Ditanya : jarak COG ?

Jawab :

$$d(q,p) = \sqrt{(319 - 221)^2 + (276 - 13)^2}$$

$$\begin{aligned}
&= \sqrt{9604 + 69169} \\
&= \sqrt{78773} \\
&= 280,6653
\end{aligned}$$

Dengan menggunakan persamaan 2.5 maka kita dapat mencari nilai *euclidean distance* , seperti yang ditunjukkan oleh tabel 11 , 12, dan 13 berikut ini.

TABEL 12

Hasil Pengujian *euclidean distance* pada navigasi terdeteksi (D)

No	Jarak dari Center Of Grafity (Pixel)	titik
1	268.262930722826	1
2	254.656631564937	2
3	254.75674672126	3
4	270.852358306144	4
5	280.665281073381	5

TABEL 13

Hasil Pengujian *euclidean distance* pada navigasi *Membesar* (+)

No	Jarak dari Center Of Grafity (Pixel)	Titik
1	268.700576850888	1
2	286.225435627234	2

TABEL 14

Hasil Pengujian *euclidean distance* pada navigasi *Mengecil* (-)

No	Jarak dari Center Of Grafity (Pixel)	Titik
1	262.617592708486	1

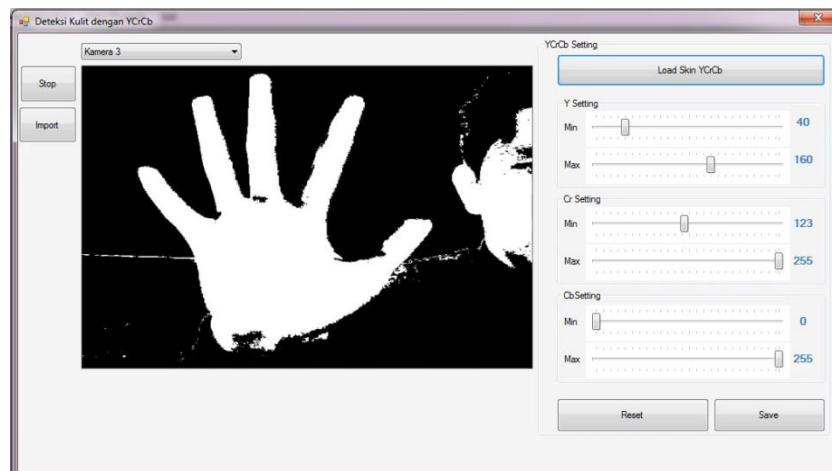
4.3.6. Pengujian Navigasi Untuk Objek *Augmented Reality*

Pada pengujian kali ini dilakukan sebanyak dua kali didalam ruangan pada siang hari dengan kondisi cahaya sedang bernilai 70 lx, dimana pada pengujian pertama *background* terdapat *obstacles* sedangkan pada pengujian kedua *background* hanya berwarna putih .

1. Pengujian pertama

Pada pengujian yang pertama dengan *background* terdapat *obstacles* didapatkan setting kalibrasi Y minimal 60 dan Y maksimal 217, Cr minimal 122

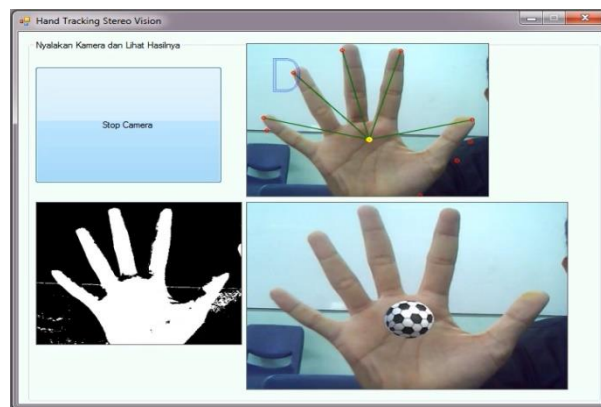
dan Cr maksimal 187, Cb minimal 97 dan Cb maksimal 134. Untuk setting kalibrasi YCrCb dapat dilihat gambar 4.17 dibawah ini.



Gambar 4.17. Setting Kalibrasi YCrCb Kondisi Siang Hari Cahaya Sedang.

Pada pengujian pertama navigasi pada objek *augmented reality* seperti deteksi, tidak terdeteksi, mengecil dan membesar dilakukan sebanyak 20 kali percobaan. Dimana masing-masing navigasi memiliki *persentasi* rata-rata *error* yang berbeda-beda dan dapat dilihat hasilnya dibawah ini.

1) Navigasi Deteksi (D)



Gambar 4.18. Navigasi Deteksi (D) pada objek Augmented Reality.

Tabel 15

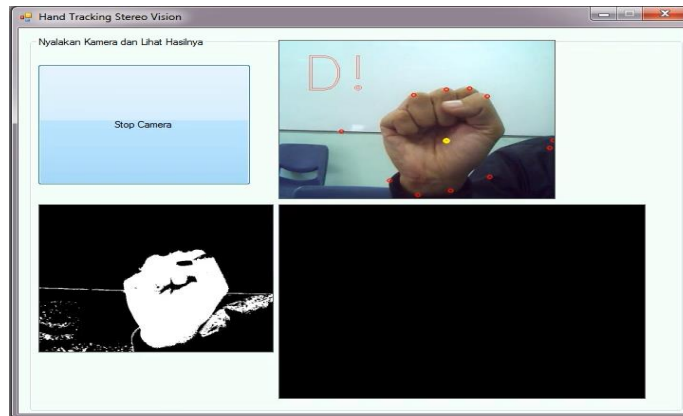
Data Pengujian Sistem Navigasi Deteksi (D).

Percobaan	Navigasi deteksi (D)	
	Benar	Salah
1	✓	-
2	-	✓
3	✓	-
4	✓	-
5	✓	-
6	✓	-
7	✓	-
8	✓	-
9	-	✓
10	✓	-
11	✓	-
12		✓
13	✓	-
14	✓	-
15	-	✓
16	✓	-
17	✓	-
18	✓	-
19	✓	-
20	✓	-

Berdasarkan data tabel 4.15 diatas, pada pengujian navigasi deteksi (D) didapatkan data yang benar sebanyak 17 dari 20 kali percobaan, sedangkan data yang salah 3 maka didapatkan *persentasi* rata-rata error sebagai berikut.

$$\text{Persentasi rata-rata error} = \frac{3}{20} \times 100\% = 15\%$$

2) Navigasi Tidak Deteksi (D!)



Gambar 4.19. Navigasi Tidak Deteksi (D!) pada objek Augmented Reality.

Tabel 16

Data Pengujian Sistem Navigasi Tidak Terdeteksi (D!)

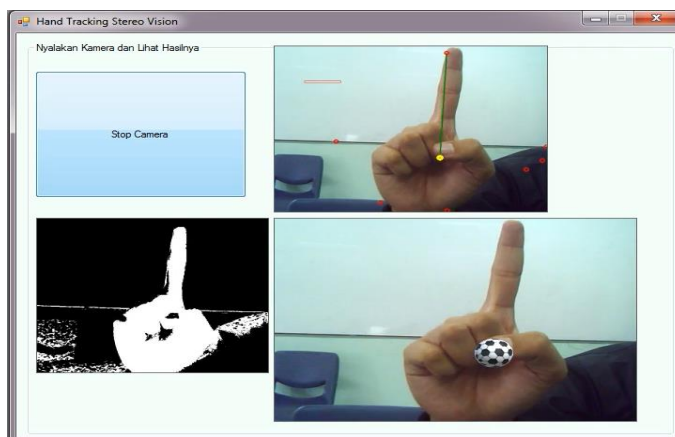
percobaan	Navigasi Tidak Tereteksi (D!)	
	Benar	Salah
1	✓	-
2	✓	-
3	✓	-
4	✓	-
5	✓	-
6	✓	-
7	✓	-
8	✓	-
9	✓	-
10	✓	-
11	✓	-
12	✓	-
13	✓	-
14	✓	-
15	✓	-
16	✓	-

17	✓	-
18	✓	-
19	✓	-
20	✓	-

Berdasarkan data tabel 4.16 diatas, pada pengujian navigasi tidak terdeteksi (D!) didapatkan data yang benar sebanyak 20 dari 20 kali percobaan, sedangkan data yang salah adalah 0 maka didapatkan *persentasi* rata-rata error sebagai berikut.

$$\text{Persentasi rata-rata error} = \frac{0}{20} \times 100\% = 0\%$$

3) Navigasi Mengecil



Gambar 4.20. Navigasi Mengecil (-) pada objek Augmented Reality.

Tabel 17

Data Pengujian Sistem Navigasi Mengecil (-)

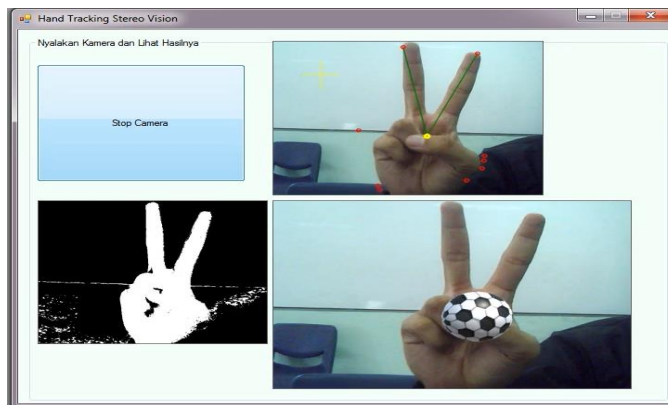
percobaan	Navigasi mengecil (-)	
	Benar	Salah
1	✓	-
2	✓	-
3	✓	-
4	-	✓
5	✓	-

6	✓	-
7	✓	-
8	✓	-
9	✓	-
10	✓	-
11	✓	-
12	✓	-
13	✓	-
14	✓	-
15	✓	-
16	✓	-
17	-	✓
18	-	✓
19	✓	-
20	✓	-

Berdasarkan data tabel 4.17 diatas, pada pengujian navigasi mengecil (-) didapatkan data yang benar sebanyak 17 dari 20 kali percobaan, sedangkan data yang salah adalah 3 maka didapatkan *persentasi* rata-rata error sebagai berikut.

$$Persentasi \text{ rata-rata error} = \frac{3}{20} \times 100\% = 15\%$$

4) Navigasi Membesar (+)



Gambar 4.21. Navigasi Membesar (+) pada objek Augmented Reality.

Tabel 18

Data Pengujian Sistem Navigasi Membesar (+)

percobaan	Navigasi membesar (+)	
	Benar	Salah
1	✓	-
2	✓	-
3	-	✓
4	✓	-
5	✓	-
6	✓	-
7	✓	-
8	✓	-
9	✓	-
10	✓	-
11	✓	-
12	✓	-
13	✓	-
14	✓	-
15	✓	-
16	-	✓
17	✓	-
18	✓	-
19	✓	-
20	✓	-

Berdasarkan data tabel 4.18 diatas, pada pengujian navigasi membesar (+) didapatkan data yang benar sebanyak 18 dari 20 kali percobaan, sedangkan data yang salah adalah 2 maka didapatkan *persentasi* rata-rata error sebagai berikut.

$$\text{Persentasi rata-rata error} = \frac{2}{20} \times 100\% = 10\%$$

2. Pengujian Kedua

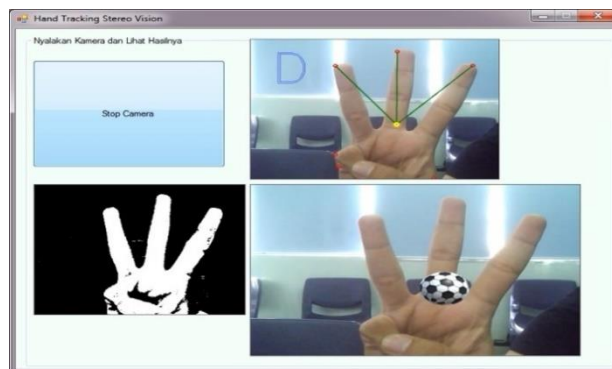
Pada pengujian yang kedua dimana *background* terdapat *obstacles* dan navigasi menggunakan jari yang lebih sedikit didapatkan setting kalibrasi Y minimal 104 dan Y maksimal 255, Cr minimal 130 dan Cr maksimal 255, Cb minimal 0 dan Cb maksimal 255. Untuk setting kalibrasi YCrCb dapat dilihat gambar 4.22 dibawah ini.



Gambar 4.22. Setting Kalibrasi YCrCb Kondisi Siang Hari Cahaya Sedang

Pada pengujian pertama navigasi pada objek *augmented reality* seperti deteksi, tidak terdeteksi, mengecil dan membesar dilakukan sebanyak 20 kali percobaan. Dimana masing-masing navigasi memiliki *persentasi* rata-rata *error* yang berbeda-beda dan dapat dilihat hasilnya dibawah ini.

1) Navigasi Deteksi



Gambar 4.23. Navigasi Deteksi (D) pada objek Augmented Reality.

Tabel 19

Data Pengujian Sistem Navigasi Deteksi (D).

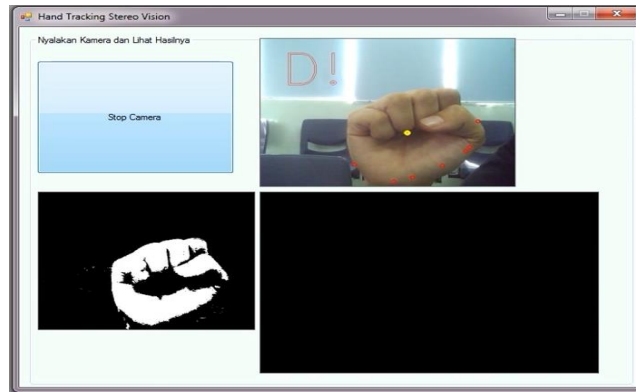
Percobaan	Navigasi deteksi (D)
-----------	----------------------

	Benar	Salah
1	✓	-
2	✓	-
3	✓	-
4	✓	-
5	✓	-
6	✓	-
7	✓	-
8	✓	-
9	✓	-
10	✓	-
11	✓	-
12	✓	-
13	✓	-
14	✓	-
15	✓	-
16	✓	-
17	✓	-
18	✓	-
19	✓	-
20	✓	-

Berdasarkan data tabel 4.19 diatas, pada pengujian navigasi deteksi (D) didapatkan data yang benar sebanyak 20 dari 20 kali percobaan, maka didapatkan *persentasi* rata-rata error sebagai berikut.

$$\text{Persentasi rata-rata error} = \frac{0}{20} \times 100\% = 0\%$$

2) Navigasi Tidak Deteksi (D!)



Gambar 4.24. Navigasi Tidak Deteksi (D!) pada objek Augmented Reality.

Tabel 20

Data Pengujian Sistem Navigasi Tidak Terdeteksi (D!)

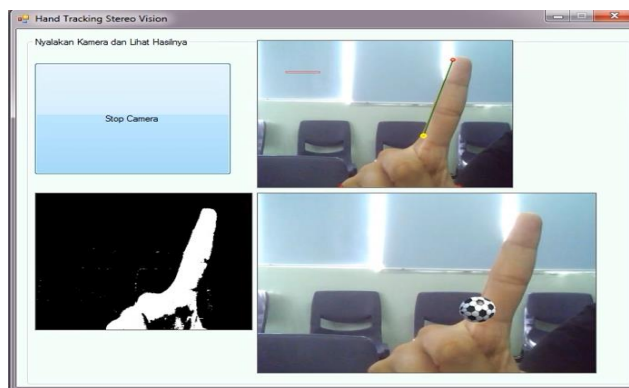
percobaan	Navigasi Tidak Tereteksi (D!)	
	Benar	Salah
1	✓	-
2	✓	-
3	✓	-
4	✓	-
5	✓	-
6	✓	-
7	✓	-
8	✓	-
9	✓	-
10	✓	-
11	✓	-
12	✓	-
13	✓	-
14	✓	-
15	✓	-
16	✓	-
17	✓	-
18	✓	-

19	✓	-
20	✓	-

Berdasarkan data tabel 4.20 diatas, pada pengujian navigasi tidak terdeteksi (D!) didapatkan data yang benar sebanyak 20 dari 20 kali percobaan, sedangkan data yang salah adalah 0 maka didapatkan *persentasi* rata-rata error sebagai berikut.

$$\text{Persentasi rata-rata error} = \frac{0}{20} \times 100\% = 0\%$$

3) Navigasi Mengecil



Gambar 4.25. Navigasi Mengecil (-) pada objek Augmented Reality.

Tabel 21

Data Pengujian Sistem Navigasi Mengecil (-)

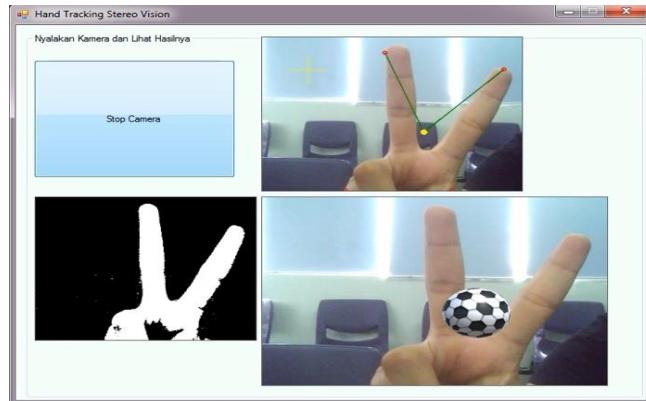
percobaan	Navigasi mengecil (-)	
	Benar	Salah
1	✓	-
2	✓	-
3	✓	-
4	✓	-
5	✓	-

6	✓	-
7	✓	-
8	✓	-
9	✓	-
10	✓	-
11	✓	-
12	✓	-
13	✓	-
14	✓	-
15	✓	-
16	✓	-
17	✓	-
18	✓	-
19	✓	-
20	✓	-

Berdasarkan data tabel 4.21 diatas, pada pengujian navigasi mengecil (-) didapatkan data yang benar sebanyak 20 dari 20 kali percobaan, sedangkan data yang salah adalah 0 maka didapatkan *persentasi* rata-rata error sebagai berikut.

$$Persentasi \text{ rata-rata error} = \frac{0}{20} \times 100\% = 0\%$$

4) Navigasi Membesar (+)



Gambar 4.26. Navigasi Membesar (+) pada objek Augmented Reality.

Tabel 22

Data Pengujian Sistem Navigasi Membesar (+)

percobaan	Navigasi membesar (+)	
	Benar	Salah
1	✓	-
2	✓	-
3	✓	-
4	✓	-
5	✓	-
6	✓	-
7	✓	-
8	✓	-
9	✓	-
10	✓	-
11	✓	-
12	✓	-
13	✓	-
14	✓	-
15	✓	-
16	✓	-
17	✓	-
18	✓	-

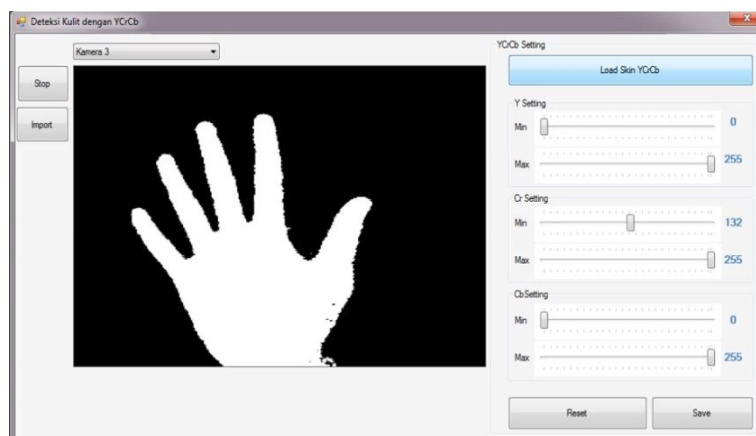
19	✓	-
20	✓	-

Berdasarkan data tabel 22 diatas, pada pengujian navigasi membesar (+) didapatkan data yang benar sebanyak 20 dari 20 kali percobaan, sedangkan data yang salah adalah 0 maka didapatkan *persentasi* rata-rata error sebagai berikut.

$$\text{Persentasi rata-rata error} = \frac{0}{20} \times 100\% = 0\%$$

3. Pengujian Ketiga

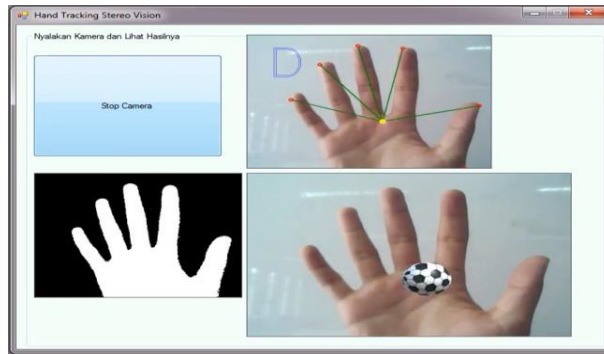
Pada pengujian yang kedua dimana background hanya berwarna putih didapatkan setting kalibrasi Y minimal 0 dan Y maksimal 255, Cr minimal 132 dan Cr maksimal 255, Cb minilal 0 dan Cb maksimal 255. Untuk setting kalibari YCrCb dapat dilihat gambar 4.27 dibawah ini.



Gambar 4.27. Setting Kalibari YCrCb Kondisi Siang Hari Cahaya Sedang.

Pada pengujian yang navigasi pada objek *augmented reality* seperti deteksi, tidak terdeteksi, mengecil dan membesar dilakukan sebanyak 20 kali percobaan. Dimana masing-masing navigasi memiliki *persentasi* rata-rata *error* yang berbeda-beda dan dapat dilihat hasilnya dibawah ini.

1) Navigasi Deteksi



Gambar 4.28. Navigasi Deteksi (D) pada objek Augmented Reality.

Tabel 23

Data Pengujian Sistem Navigasi Deteksi (D).

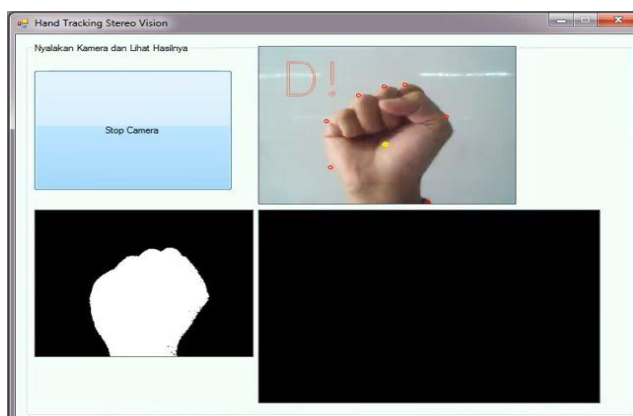
Percobaan	Navigasi deteksi (D)	
	Benar	Salah
1	✓	-
2	✓	-
3	✓	-
4	✓	-
5	✓	-
6	✓	-
7	✓	-
8	✓	-
9	✓	-
10	✓	-
11	✓	-
12	✓	-
13	✓	-
14	✓	-
15	✓	-
16	✓	-
17	✓	-
18	✓	-
19	✓	-

20	✓	-
----	---	---

Berdasarkan data tabel 4.23 diatas, pada pengujian navigasi deteksi (D) didapatkan data yang benar sebanyak 20 dari 20 kali percobaan, maka didapatkan *persentasi* rata-rata error sebagai berikut.

$$\text{Persentasi rata-rata error} = \frac{0}{20} \times 100\% = 0\%$$

2) Navigasi Tidak Deteksi (D!)



Gambar 4.29. Navigasi Tidak Deteksi (D!) pada objek Augmented Reality.

Tabel 24

Data Pengujian Sistem Navigasi Tidak Terdeteksi (D!)

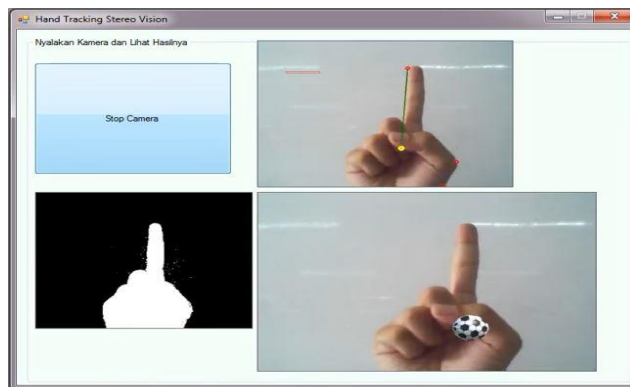
percobaan	Navigasi Tidak Tereteksi (D!)	
	Benar	Salah
1	✓	-
2	✓	-
3	✓	-
4	✓	-
5	✓	-
6	✓	-
7	✓	-
8	✓	-
9	✓	-

10	✓	-
11	✓	-
12	✓	-
13	✓	-
14	✓	-
15	✓	-
16	✓	-
17	✓	-
18	✓	-
19	✓	-
20	✓	-

Berdasarkan data tabel 4.24 diatas, pada pengujian navigasi tidak terdeteksi (D!) didapatkan data yang benar sebanyak 20 dari 20 kali percobaan, sedangkan data yang salah adalah 0 maka didapatkan *persentasi* rata-rata error sebagai berikut.

$$\text{Persentasi rata-rata error} = \frac{0}{20} \times 100\% = 0\%$$

3) Navigasi Mengecil



Gambar 4.30. Navigasi Mengecil (-) pada objek Augmented Reality.

Tabel 25

Data Pengujian Sistem Navigasi Mengecil (-)

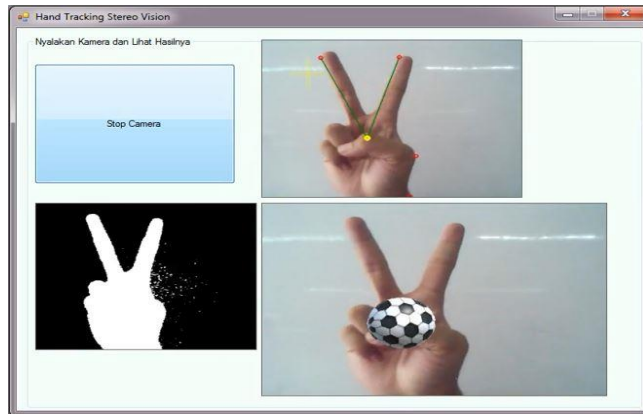
percobaan	Navigasi mengecil (-)
-----------	-----------------------

	Benar	Salah
1	✓	-
2	✓	-
3	✓	-
4	✓	-
5	✓	-
6	✓	-
7	✓	-
8	✓	-
9	✓	-
10	✓	-
11	✓	-
12	✓	-
13	✓	-
14	✓	-
15	✓	-
16	✓	-
17	✓	-
18	✓	-
19	✓	-
20	✓	-

Berdasarkan data tabel 4.25 diatas, pada pengujian navigasi mengecil (-) didapatkan data yang benar sebanyak 20 dari 20 kali percobaan, sedangkan data yang salah adalah 0 maka didapatkan *persentasi* rata-rata error sebagai berikut.

$$Persentasi \text{ rata-rata error} = \frac{0}{20} \times 100\% = 0\%$$

4) Navigasi Membesar (+)



Gambar 4.31. Navigasi Membesar (+) pada objek Augmented Reality.

Tabel 26

Data Pengujian Sistem Navigasi Membesar (+)

percobaan	Navigasi membesar (+)	
	Benar	Salah
1	✓	-
2	✓	-
3	✓	-
4	✓	-
5	✓	-
6	✓	-
7	✓	-
8	✓	-
9	✓	-
10	✓	-
11	✓	-
12	✓	-
13	✓	-
14	✓	-
15	✓	-
16	✓	-
17	✓	-
18	✓	-

19	✓	-
20	✓	-

Berdasarkan data tabel 26 diatas, pada pengujian navigasi membesar (+) didapatkan data yang benar sebanyak 20 dari 20 kali percobaan, sedangkan data yang salah adalah 0 maka didapatkan *persentasi* rata-rata error sebagai berikut.

$$\text{Persentasi rata-rata error} = \frac{0}{20} \times 100\% = 0\%$$

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Setelah melakukan pengujian dan menganalisa hasil pengujian maka didapatkan kesimpulan sebagai berikut :

1. Segmentasi warna kulit berhasil dilakukan dengan menggunakan metode YCrCb yang dilakukan dengan kondisi cahaya yang ideal yaitu cahaya sedang bernilai 70 lx
2. Jarak dari *Webcam* dengan tangan sangat berpengaruh dalam pendeteksian, jika jarak antara *Webcam* dengan tangan jauh maka sistem akan kesulitan dalam melakukan pendeteksian. Jarak ideal pada penelitian ini adalah 30 cm hingga 45 cm.
3. Pengujian navigasi tangan pada objek *Augmented Reality* dengan *background* yang terdapat obstacles pada navigasi terdeteksi, tidak terdeteksi, mengecil dan membesar dan didapatkan rata-rata presentasi *error* masing-masing yaitu 15 % , 0 % , 15% dan 10%.
4. Pengujian Pengujian navigasi tangan pada objek *Augmented Reality* dengan *background* yang ditentukan berwarna putih memiliki akurasi 100 %
5. Semakin sedikit jumlah jari yang dideteksi maka semakin kecil error yang didapatkan
6. warna *backgroud* berpengaruh dalam deteksi tangan, sebaiknya menghindari warna *background* yang mempunyai kesamaan dengan warna kulit.

5.2. Saran

Adapun beberapa saran yang didapatkan dari hasil penelitian ini agar dapat dijadikan acuan pada penelitian selanjutnya adalah sebagai berikut :

1. Ketika melakukan pendeteksian tangan disarankan agar dapat melakukan kalibrasi YCrCb secara otomatis.
2. Selain untuk melakukan navigasi objek Augmented Reality , diharapkan pengenalan pola jari tangan juga dapat dikembangkan agar dapat melakukan navigasi lain seperti mengendalikan robot dan lain sebagainya

Daftar pustaka

- [1] G. M. Re and M. Bordegoni, *Virtual, Augmented and Mixed Reality. Applications of Virtual and Augmented Reality*, vol. 8526, no. PART 2. Cham: Springer International Publishing, 2014.
- [2] Y. Shiqiang, Q. Dan, and L. Peilei, "Research on Hand Recognition Method Based on Markov Random Field," *Procedia Eng.*, vol. 174, pp. 482–488, 2017.
- [3] C. Ttofis, C. Kyrkou, and T. Theocharides, "A Low-Cost Real-Time Embedded Stereo Vision System for Accurate Disparity Estimation Based on Guided Image Filtering," *IEEE Trans. Comput.*, vol. 65, no. 9, pp. 2678–2693, Sep. 2016.
- [4] A. P. Renold and S. Chandrakala, "Convex-Hull-Based Boundary Detection in Unattended Wireless Sensor Networks," *IEEE Sensors Lett.*, vol. 1, no. 4, pp. 1–4, Aug. 2017.
- [5] A. J. P. Gomes, "A Total Order Heuristic-Based Convex Hull Algorithm for Points in the Plane," *CAD Comput. Aided Des.*, vol. 70, pp. 153–160, 2016.
- [6] W. Chen, H. Gao, and X. Wen, "An image feature extraction method based on geometric invariant moments," *Bol. Tec. Bull.*, vol. 55, no. 12, pp. 19–26, 2017.
- [7] I. Dokmanic, R. Parhizkar, J. Ranieri, and M. Vetterli, "Euclidean Distance Matrices: Essential theory, algorithms, and applications," *IEEE Signal Process. Mag.*, vol. 32, no. 6, pp. 12–30, Nov. 2015.
- [8] N. Kawai, T. Sato, Y. Nakashima, and N. Yokoya, "Augmented Reality Marker Hiding with Texture Deformation," *IEEE Trans. Vis. Comput. Graph.*, vol. 23, no. 10, pp. 2288–2300, Oct. 2017.
- [9] J. Peddie, *Augmented Reality*, no. 103. Cham: Springer International Publishing, 2017.

Nilai RGB

243, 254, 255	240, 255, 255	237, 255, 255	238, 255, 255	245, 255, 255	246, 254, 255	245, 253, 255	244, 252, 255	243, 252, 255
244, 253, 255	243, 254, 255	238, 255, 255	238, 255, 255	243, 254, 255	244, 254, 255	242, 252, 254	241, 251, 253	241, 249, 252
240, 248, 251	240, 248, 251	238, 247, 252	237, 248, 252	235, 249, 252	235, 249, 250	236, 247, 249	236, 246, 247	239, 247, 249
239, 247, 250	238, 246, 249	237, 246, 251	237, 246, 251	236, 247, 251	234, 248, 249	236, 248, 248	238, 247, 246	233, 237, 238
232, 244, 244	230, 244, 244	228, 244, 244	229, 243, 244	231, 242, 244	232, 242, 244	232, 242, 243	238, 240, 239	255, 244, 247
230, 242, 242	230, 242, 242	228, 242, 242	227, 241, 242	229, 239, 241	230, 238, 240	230, 239, 238	240, 236, 237	255, 235, 238
228, 237, 236	228, 237, 236	231, 237, 237	231, 236, 239	231, 236, 239	231, 236, 239	231, 237, 237	239, 235, 234	255, 234, 234
174, 185, 181	176, 185, 182	180, 184, 183	181, 185, 186	182, 186, 187	182, 186, 188	184, 188, 187	190, 186, 185	207, 189, 185
178, 189, 189	174, 189, 182	181, 190, 185	184, 189, 185	184, 188, 187	184, 188, 187	182, 186, 185	189, 190, 184	192, 185, 175
170, 191, 182	177, 194, 184	177, 187, 179	184, 189, 183	185, 187, 184	183, 188, 184	184, 189, 185	189, 192, 185	189, 189, 177
178, 198, 187	177, 193, 183	189, 197, 186	189, 192, 183	188, 189, 183	190, 193, 186	190, 195, 191	185, 187, 182	188, 190, 179
183, 190, 183	191, 192, 186	200, 191, 184	198, 188, 179	195, 191, 182	183, 184, 176	186, 188, 183	187, 189, 184	189, 194, 188
198, 184, 183	208, 187, 186	184, 144, 145	173, 135, 134	205, 190, 183	185, 185, 175	188, 190, 185	189, 194, 188	191, 193, 186
205, 187, 187	209, 179, 181	140, 85, 90	128, 73, 76	149, 124, 117	211, 202, 193	191, 190, 185	187, 189, 186	191, 196, 190
198, 195, 190	222, 201, 198	139, 84, 87	136, 76, 78	98, 61, 55	211, 192, 185	201, 193, 190	185, 187, 184	189, 195, 191
189, 191, 186	225, 210, 205	159, 109, 108	140, 79, 78	117, 72, 66	158, 129, 123	205, 195, 193	185, 187, 184	180, 186, 182
191, 193, 190	211, 200, 196	203, 163, 153	132, 78, 66	126, 76, 67	105, 70, 64	214, 204, 202	187, 191, 190	183, 188, 184
197, 199, 198	200, 191, 186	236, 204, 191	131, 83, 69	121, 69, 56	133, 85, 75	162, 147, 144	198, 198, 196	193, 192, 190
187, 191, 190	188, 183, 179	224, 200, 188	167, 128, 113	136, 82, 70	139, 87, 76	104, 76, 72	207, 196, 194	197, 191, 191
192, 198, 196	186, 185, 180	202, 185, 175	225, 193, 180	151, 109, 95	136, 87, 73	113, 76, 68	158, 134, 130	206, 196, 196
197, 201, 200	187, 188, 183	195, 186, 177	234, 212, 199	205, 173, 160	140, 94, 81	124, 75, 68	102, 64, 61	206, 188, 188
195, 195, 195	190, 191, 186	183, 183, 175	204, 194, 182	223, 201, 188	153, 117, 105	112, 62, 53	108, 63, 58	143, 115, 114
202, 196, 196	196, 195, 191	178, 186, 175	186, 186, 174	188, 191, 182	191, 184, 178	137, 93, 82	135, 90, 84	125, 84, 82
202, 194, 192	193, 192, 187	182, 192, 181	181, 189, 174	193, 183, 171	202, 189, 180	212, 181, 176	138, 95, 89	139, 90, 86
202, 197, 193	192, 192, 184	187, 195, 184	181, 192, 178	189, 189, 179	202, 189, 180	212, 181, 176	138, 95, 89	146, 93, 87
198, 195, 188	193, 194, 186	183, 191, 178	182, 190, 177	188, 191, 182	191, 184, 178	223, 202, 197	169, 132, 126	132, 79, 71
194, 197, 186	193, 196, 185	185, 193, 180	186, 192, 182	186, 187, 181	191, 186, 182	201, 190, 186	222, 195, 188	139, 86, 78
193, 201, 188	186, 194, 181	191, 197, 185	188, 191, 182	186, 185, 181	192, 188, 185	181, 176, 172	239, 219, 212	174, 125, 118
190, 202, 188	186, 194, 181	193, 194, 186	190, 187, 180	194, 191, 186	188, 185, 180	194, 191, 184	211, 194, 187	222, 182, 176
188, 200, 186	191, 199, 186	193, 193, 185	196, 193, 187	201, 198, 191	189, 189, 181	195, 195, 187	196, 186, 177	255, 222, 217
189, 199, 188	196, 204, 193	198, 197, 192	203, 198, 192	201, 197, 188	196, 196, 186	185, 188, 177	203, 196, 186	249, 219, 217
195, 203, 192	196, 202, 192	204, 204, 196	198, 195, 188	196, 193, 184	196, 196, 186	196, 199, 188	198, 194, 183	229, 205, 203
198, 201, 190	198, 201, 190	198, 199, 191	197, 198, 190	197, 198, 190	196, 197, 189	197, 198, 190	202, 199, 190	211, 198, 190
198, 200, 187	198, 200, 187	198, 201, 192	198, 201, 192	198, 201, 194	198, 201, 194	200, 200, 192	201, 198, 189	206, 199, 189

Nilai YcrCb

251,122,130	251,120,130	250,119,131	250,119,131	252,123,130	252,124,130	251,124,130	250,124,131	250,123,131
251,123,130	251,122,130	250,119,131	250,119,131	251,122,130	251,123,130	249,123,131	248,123,131	247,124,131
246,124,131	246,124,131	245,123,132	245,122,132	245,122,132	245,121,131	244,122,131	243,123,130	245,124,130
245,124,131	244,124,131	244,123,132	244,123,132	244,122,132	244,121,131	244,122,130	244,124,129	238,129,128
240,122,130	240,121,130	239,120,131	239,121,131	239,122,131	239,123,131	239,123,130	239,127,128	248,133,127
238,122,130	238,122,130	238,121,130	237,121,131	236,123,131	236,124,130	236,124,129	237,130,128	241,138,126
234,124,129	234,124,129	235,125,129	235,125,130	235,125,130	235,125,130	235,125,129	236,130,127	240,139,125
181,123,128	182,124,128	183,126,128	184,126,129	185,126,129	186,125,129	187,126,128	187,130,126	194,137,123
191,119,127	184,121,127	187,124,127	187,126,127	187,126,128	187,126,128	185,126,128	189,128,125	186,132,122
184,118,127	188,120,126	183,124,126	187,126,126	186,127,127	186,126,127	187,126,127	190,127,125	188,129,122
191,119,126	187,121,126	193,125,124	190,127,124	188,128,125	191,127,125	193,126,127	186,127,126	188,128,123
187,125,126	191,128,125	193,133,123	190,134,122	191,131,123	183,128,124	187,127,126	188,127,126	189,128,124
188,135,125	193,139,124	156,148,122	146,147,121	194,136,122	184,129,123	189,127,126	192,126,126	191,127,125
192,137,125	188,143,124	102,155,121	90,155,120	131,141,120	204,133,122	190,129,125	188,127,127	194,126,126
195,130,125	207,139,123	101,155,120	94,158,119	71,147,119	197,138,121	195,132,125	197,128,127	193,125,127
190,127,126	214,136,123	124,153,119	97,159,117	85,151,117	137,143,120	198,133,125	186,127,127	184,125,127
192,127,127	203,134,124	174,149,116	93,156,113	90,154,115	80,146,119	207,133,125	190,126,128	186,126,127
198,127,128	193,133,124	212,145,116	96,153,113	83,155,113	67,149,117	151,136,124	198,128,127	192,129,127
190,126,128	184,131,125	206,141,118	138,149,114	97,156,113	98,153,115	84,142,121	199,134,125	193,131,127
196,125,128	185,129,125	189,137,120	201,145,116	100,154,113	101,155,114	86,147,118	141,140,122	200,134,126
200,126,128	187,128,126	188,133,122	217,140,118	181,145,116	106,152,114	89,153,116	75,147,120	193,137,125
195,128,128	190,128,126	182,129,124	196,134,120	185,129,122	126,147,116	76,154,115	76,151,118	123,142,123
198,131,127	195,129,126	182,125,124	185,129,122	206,140,118	105,151,115	105,151,115	103,151,117	96,149,120
196,132,126	192,129,125	188,124,124	185,125,122	185,134,120	208,142,118	124,147,117	107,151,117	104,153,118
198,131,125	191,129,124	191,125,124	187,124,123	188,129,123	192,135,121	190,144,120	107,150,118	108,155,116
195,130,124	193,128,124	187,125,123	186,125,123	189,127,124	185,132,124	208,139,122	142,147,119	94,155,115
195,127,123	194,127,123	189,125,123	189,126,124	185,128,125	187,131,125	193,134,124	202,142,120	101,155,115
197,125,123	190,125,123	194,126,123	189,127,124	186,129,126	189,130,126	177,131,125	224,139,121	139,153,116
197,123,123	190,125,123	193,128,124	187,130,124	191,130,125	185,130,125	191,130,124	198,137,122	194,149,118
195,123,123	195,125,123	192,129,124	194,131,124	198,130,124	188,129,124	194,129,124	188,134,122	231,145,120
195,124,124	200,125,124	197,129,125	199,131,124	197,131,123	195,129,123	186,127,123	197,132,122	228,143,122
199,125,124	199,126,124	203,129,124	195,130,124	193,130,123	195,129,123	197,127,123	194,131,122	212,140,123
199,127,123	199,127,123	198,128,124	197,128,124	197,128,124	196,128,124	197,128,124	199,130,123	201,135,122
198,128,122	198,128,122	199,127,124	199,127,124	199,127,125	199,127,125	199,129,124	198,130,123	200,132,122

Background memiliki obstacles 1

navigasi Deteksi (D)

```
===== NILAI RGB Pixels
=====
Titik Tengah X 348,Titik Tengah Y : 307
Titik 1 (152,108) =>Jarak dari COG : 279.315234099395
Titik 2 (1,271) =>Jarak dari COG : 348.862437072265
Titik 3 (589,231) =>Jarak dari COG : 252.69942619642
Titik 4 (449,37) =>Jarak dari COG : 288.272440583556
Titik 5 (293,17) =>Jarak dari COG : 295.169442862909
===== NILAI RGB Pixels
=====
Titik Tengah X 363,Titik Tengah Y : 309
Titik 1 (162,96) =>Jarak dari COG : 292.865156684779
Titik 2 (66,273) =>Jarak dari COG : 299.173862494704
Titik 3 (461,35) =>Jarak dari COG : 290.998281781869
Titik 4 (281,33) =>Jarak dari COG : 287.923600977759
===== NILAI RGB Pixels
=====
Titik Tengah X 359,Titik Tengah Y : 308
Titik 1 (186,91) =>Jarak dari COG : 277.521170363632
Titik 2 (1,270) =>Jarak dari COG : 360.011110939649
Titik 3 (607,241) =>Jarak dari COG : 256.891027480525
Titik 4 (467,37) =>Jarak dari COG : 291.727612680048
Titik 5 (307,17) =>Jarak dari COG : 295.609539764873
===== NILAI RGB Pixels
=====
Titik Tengah X 363,Titik Tengah Y : 309
Titik 1 (169,107) =>Jarak dari COG : 280.071419462965
Titik 2 (1,270) =>Jarak dari COG : 364.094767883308
Titik 3 (609,239) =>Jarak dari COG : 255.765517613301
Titik 4 (471,41) =>Jarak dari COG : 288.942900933731
Titik 5 (307,18) =>Jarak dari COG : 296.339332522701
===== NILAI RGB Pixels
=====
Titik Tengah X 368,Titik Tengah Y : 311
Titik 1 (172,97) =>Jarak dari COG : 290.193039199771
Titik 2 (1,271) =>Jarak dari COG : 369.173400992
Titik 3 (611,243) =>Jarak dari COG : 252.335094665803
Titik 4 (467,37) =>Jarak dari COG : 291.33657511545
Titik 5 (308,20) =>Jarak dari COG : 297.121187396658
===== NILAI RGB Pixels
=====
Titik Tengah X 370,Titik Tengah Y : 307
Titik 1 (176,96) =>Jarak dari COG : 286.630424065555
Titik 2 (1,270) =>Jarak dari COG : 370.850374140299
Titik 3 (613,241) =>Jarak dari COG : 251.80349481292
Titik 4 (473,37) =>Jarak dari COG : 288.979238008546
Titik 5 (316,20) =>Jarak dari COG : 292.035956690268
===== NILAI RGB Pixels
=====
Titik Tengah X 365,Titik Tengah Y : 308
Titik 1 (174,110) =>Jarak dari COG : 275.109069279804
Titik 2 (1,270) =>Jarak dari COG : 365.978141423774
Titik 3 (612,241) =>Jarak dari COG : 255.925770488241
Titik 4 (472,36) =>Jarak dari COG : 292.289240308295
Titik 5 (318,20) =>Jarak dari COG : 291.809869606907
===== NILAI RGB Pixels
=====
Titik Tengah X 370,Titik Tengah Y : 313
```

```

Titik 1 (178,111) =>Jarak dari COG : 278.689791703966
Titik 2 (24,272) =>Jarak dari COG : 348.420722690256
Titik 3 (617,243) =>Jarak dari COG : 256.727481972616
Titik 4 (479,39) =>Jarak dari COG : 294.884723239438
Titik 5 (322,20) =>Jarak dari COG : 296.90570893804
===== NILAI RGB Pixels
=====
Titik Tengah X 382,Titik Tengah Y : 309
Titik 1 (182,111) =>Jarak dari COG : 281.432052190222
Titik 2 (1,270) =>Jarak dari COG : 382.990861509775
Titik 3 (485,39) =>Jarak dari COG : 288.979238008546
Titik 4 (305,34) =>Jarak dari COG : 285.576609686438
===== NILAI RGB Pixels
=====
D
Titik Tengah X 372,Titik Tengah Y : 311
Titik 1 (184,97) =>Jarak dari COG : 284.850838159202
Titik 2 (1,270) =>Jarak dari COG : 373.258623477074
Titik 3 (619,245) =>Jarak dari COG : 255.665797477879
Titik 4 (481,39) =>Jarak dari COG : 293.027302482209
Titik 5 (328,21) =>Jarak dari COG : 293.318939040765
===== NILAI RGB Pixels
=====
D
Titik Tengah X 372,Titik Tengah Y : 311
Titik 1 (183,115) =>Jarak dari COG : 272.281104742874
Titik 2 (62,274) =>Jarak dari COG : 312.200256245891
Titik 3 (618,247) =>Jarak dari COG : 254.188906130854
Titik 4 (487,45) =>Jarak dari COG : 289.794754955986
Titik 5 (322,24) =>Jarak dari COG : 291.322844967572
===== NILAI RGB Pixels
=====
D
Titik Tengah X 324,Titik Tengah Y : 301
Titik 1 (123,91) =>Jarak dari COG : 290.690557122174
Titik 2 (44,233) =>Jarak dari COG : 288.138855415232
Titik 3 (597,239) =>Jarak dari COG : 279.951781562468
Titik 4 (408,22) =>Jarak dari COG : 291.370897654519
Titik 5 (252,20) =>Jarak dari COG : 290.077575831018
===== NILAI RGB Pixels
=====
D
Titik Tengah X 330,Titik Tengah Y : 305
Titik 1 (46,233) =>Jarak dari COG : 292.98464123568
Titik 2 (601,253) =>Jarak dari COG : 275.943834864996
Titik 3 (419,21) =>Jarak dari COG : 297.618883809479
Titik 4 (223,31) =>Jarak dari COG : 294.151321601655
Titik 5 (122,93) =>Jarak dari COG : 296.998316493545
===== NILAI RGB Pixels
=====
D
Titik Tengah X 324,Titik Tengah Y : 303
Titik 1 (43,241) =>Jarak dari COG : 287.758579368192
Titik 2 (592,237) =>Jarak dari COG : 276.007246281687
Titik 3 (408,18) =>Jarak dari COG : 297.121187396658
Titik 4 (229,25) =>Jarak dari COG : 293.783934210161
Titik 5 (122,92) =>Jarak dari COG : 292.104433379571
===== NILAI RGB Pixels
=====
D
Titik Tengah X 333,Titik Tengah Y : 326
Titik 1 (43,237) =>Jarak dari COG : 303.349633261687
Titik 2 (596,238) =>Jarak dari COG : 277.331931086199
Titik 3 (413,23) =>Jarak dari COG : 313.38315206788
Titik 4 (119,105) =>Jarak dari COG : 307.631272792608
===== NILAI RGB Pixels
=====
D
Titik Tengah X 325,Titik Tengah Y : 305

```

```

Titik 1 (45,232) =>Jarak dari COG : 289.359637821172
Titik 2 (591,235) =>Jarak dari COG : 275.056357861439
Titik 3 (409,21) =>Jarak dari COG : 296.162117766604
Titik 4 (230,28) =>Jarak dari COG : 292.837839085047
Titik 5 (123,91) =>Jarak dari COG : 294.278779391243
===== NILAI RGB Pixels
=====
Titik Tengah X 368,Titik Tengah Y : 304
Titik 1 (176,98) =>Jarak dari COG : 281.602556806574
Titik 2 (1,270) =>Jarak dari COG : 368.571566998867
Titik 3 (633,257) =>Jarak dari COG : 269.135653528105
Titik 4 (460,26) =>Jarak dari COG : 292.82759432813
Titik 5 (314,11) =>Jarak dari COG : 297.934556572412
===== NILAI RGB Pixels
=====
Titik Tengah X 335,Titik Tengah Y : 304
Titik 1 (122,92) =>Jarak dari COG : 300.521213893462
Titik 2 (61,226) =>Jarak dari COG : 284.88594208911
Titik 3 (598,224) =>Jarak dari COG : 274.898162962214
Titik 4 (431,21) =>Jarak dari COG : 298.839421763595
Titik 5 (253,18) =>Jarak dari COG : 297.523108346226
===== NILAI RGB Pixels
=====
Titik Tengah X 331,Titik Tengah Y : 304
Titik 1 (60,224) =>Jarak dari COG : 282.561497731025
Titik 2 (601,227) =>Jarak dari COG : 280.765026312039
Titik 3 (427,19) =>Jarak dari COG : 300.734101824186
Titik 4 (252,12) =>Jarak dari COG : 302.497933877242
Titik 5 (121,96) =>Jarak dari COG : 295.574017802648
===== NILAI RGB Pixels
=====
Titik Tengah X 336,Titik Tengah Y : 309
Titik 1 (121,102) =>Jarak dari COG : 298.452676315693
Titik 2 (57,237) =>Jarak dari COG : 288.140590684478
Titik 3 (594,236) =>Jarak dari COG : 268.128700440665
Titik 4 (429,29) =>Jarak dari COG : 295.040675161917
Titik 5 (236,38) =>Jarak dari COG : 288.861558536265

```

Navigasi tidak Deteksi (D!)

```

===== NILAI RGB Pixels
=====
Titik Tengah X 465,Titik Tengah Y : 267
===== NILAI RGB Pixels
=====
Titik Tengah X 424,Titik Tengah Y : 276
===== NILAI RGB Pixels
=====
Titik Tengah X 380,Titik Tengah Y : 283
===== NILAI RGB Pixels
=====
Titik Tengah X 357,Titik Tengah Y : 283
===== NILAI RGB Pixels
=====
Titik Tengah X 350,Titik Tengah Y : 285
===== NILAI RGB Pixels
=====
Titik Tengah X 369,Titik Tengah Y : 284
===== NILAI RGB Pixels
=====
Titik Tengah X 375,Titik Tengah Y : 280

```

```

===== NILAI RGB Pixels
=====
Titik Tengah X 363,Titik Tengah Y : 287
===== NILAI RGB Pixels
=====
Titik Tengah X 409,Titik Tengah Y : 345
===== NILAI RGB Pixels
=====
Titik Tengah X 389,Titik Tengah Y : 342
===== NILAI RGB Pixels
=====
Titik Tengah X 390,Titik Tengah Y : 340
===== NILAI RGB Pixels
=====
Titik Tengah X 309,Titik Tengah Y : 337
===== NILAI RGB Pixels
=====
Titik Tengah X 395,Titik Tengah Y : 304
===== NILAI RGB Pixels
=====
Titik Tengah X 310,Titik Tengah Y : 304
===== NILAI RGB Pixels
=====
Titik Tengah X 314,Titik Tengah Y : 313
===== NILAI RGB Pixels
=====
Titik Tengah X 312,Titik Tengah Y : 314
===== NILAI RGB Pixels
=====
Titik Tengah X 304,Titik Tengah Y : 315
===== NILAI RGB Pixels
=====
Titik Tengah X 297,Titik Tengah Y : 316
===== NILAI RGB Pixels
=====
Titik Tengah X 298,Titik Tengah Y : 317
===== NILAI RGB Pixels
=====
Titik Tengah X 301,Titik Tengah Y : 316

```

Data navigasi Mengecil (-)

```

===== NILAI RGB Pixels
===== -
Titik Tengah X 399,Titik Tengah Y : 318
Titik 1 (419,2) =>Jarak dari COG : 316.632278834613
===== NILAI RGB Pixels
===== -
Titik Tengah X 388,Titik Tengah Y : 319
Titik 1 (417,4) =>Jarak dari COG : 316.332103966701
===== NILAI RGB Pixels
===== -
Titik Tengah X 388,Titik Tengah Y : 323
Titik 1 (404,19) =>Jarak dari COG : 304.420761447047
===== NILAI RGB Pixels
=====
Titik Tengah X 382,Titik Tengah Y : 333
Titik 1 (637,292) =>Jarak dari COG : 258.27504718807

```

```

Titik 2 (377,36) =>Jarak dari COG : 297.042084560421
===== NILAI RGB Pixels
===== -
Titik Tengah X 359,Titik Tengah Y : 325
Titik 1 (384,30) =>Jarak dari COG : 296.057426861749
===== NILAI RGB Pixels
===== -
Titik Tengah X 377,Titik Tengah Y : 345
Titik 1 (432,43) =>Jarak dari COG : 306.96742498187
===== NILAI RGB Pixels
===== -
Titik Tengah X 397,Titik Tengah Y : 357
Titik 1 (448,35) =>Jarak dari COG : 326.013803388752
===== NILAI RGB Pixels
===== -
Titik Tengah X 407,Titik Tengah Y : 358
Titik 1 (435,46) =>Jarak dari COG : 313.253890638249
===== NILAI RGB Pixels
===== -
Titik Tengah X 402,Titik Tengah Y : 351
Titik 1 (441,44) =>Jarak dari COG : 309.467284215957
===== NILAI RGB Pixels
===== -
Titik Tengah X 410,Titik Tengah Y : 356
Titik 1 (441,44) =>Jarak dari COG : 313.536281792076
===== NILAI RGB Pixels
===== -
Titik Tengah X 396,Titik Tengah Y : 357
Titik 1 (441,44) =>Jarak dari COG : 316.218279041551
===== NILAI RGB Pixels
===== -
Titik Tengah X 414,Titik Tengah Y : 355
Titik 1 (453,45) =>Jarak dari COG : 312.443594909545
===== NILAI RGB Pixels
===== -
Titik Tengah X 421,Titik Tengah Y : 353
Titik 1 (471,50) =>Jarak dari COG : 307.097704322256
===== NILAI RGB Pixels
===== -
Titik Tengah X 395,Titik Tengah Y : 346
Titik 1 (142,276) =>Jarak dari COG : 262.505238042977
===== NILAI RGB Pixels
===== -
Titik Tengah X 418,Titik Tengah Y : 369
Titik 1 (1,270) =>Jarak dari COG : 428.590713851805
===== NILAI RGB Pixels
===== -
Titik Tengah X 422,Titik Tengah Y : 372
Titik 1 (1,270) =>Jarak dari COG : 433.18010111269
===== NILAI RGB Pixels
===== -
Titik Tengah X 350,Titik Tengah Y : 375
Titik 1 (1,270) =>Jarak dari COG : 364.453014804378
Titik 2 (638,291) =>Jarak dari COG : 300
===== NILAI RGB Pixels
=====

```

Titik Tengah X 381, Titik Tengah Y : 371
 Titik 1 (1,270) =>Jarak dari COG : 393.193336667854
 Titik 2 (638,291) =>Jarak dari COG : 269.163519073444
 ===== NILAI RGB Pixels
 ===== -
 Titik Tengah X 360, Titik Tengah Y : 367
 Titik 1 (638,292) =>Jarak dari COG : 287.939229699602
 ===== NILAI RGB Pixels
 ===== -
 Titik Tengah X 348, Titik Tengah Y : 315
 Titik 1 (615,291) =>Jarak dari COG : 268.076481624181

Data navigasi Membesar (+)

===== NILAI RGB Pixels
 ===== +
 Titik Tengah X 341, Titik Tengah Y : 250
 Titik 1 (404,7) =>Jarak dari COG : 251.033862257664
 Titik 2 (139,92) =>Jarak dari COG : 256.452724688197
 ===== NILAI RGB Pixels
 ===== +
 Titik Tengah X 336, Titik Tengah Y : 244
 Titik 1 (404,1) =>Jarak dari COG : 252.335094665803
 Titik 2 (143,79) =>Jarak dari COG : 253.917309374528
 ===== NILAI RGB Pixels
 ===== +
 Titik Tengah X 321, Titik Tengah Y : 280
 Titik 1 (386,9) =>Jarak dari COG : 278.686203461887
 ===== NILAI RGB Pixels
 ===== +
 Titik Tengah X 290, Titik Tengah Y : 283
 Titik 1 (140,72) =>Jarak dari COG : 258.884143971777
 Titik 2 (380,29) =>Jarak dari COG : 269.473560855235
 ===== NILAI RGB Pixels
 ===== +
 Titik Tengah X 291, Titik Tengah Y : 308
 Titik 1 (127,111) =>Jarak dari COG : 256.329865602899
 Titik 2 (376,56) =>Jarak dari COG : 265.949243277735
 ===== NILAI RGB Pixels
 ===== +
 Titik Tengah X 335, Titik Tengah Y : 307
 Titik 1 (259,21) =>Jarak dari COG : 295.925666342073
 Titik 2 (458,42) =>Jarak dari COG : 292.154068943084
 ===== NILAI RGB Pixels
 ===== +
 Titik Tengah X 338, Titik Tengah Y : 307
 Titik 1 (261,35) =>Jarak dari COG : 282.688874913747
 Titik 2 (472,44) =>Jarak dari COG : 295.169442862909
 ===== NILAI RGB Pixels
 ===== +
 Titik Tengah X 335, Titik Tengah Y : 306
 Titik 1 (257,24) =>Jarak dari COG : 292.588448165679
 Titik 2 (459,42) =>Jarak dari COG : 291.671047586146
 ===== NILAI RGB Pixels
 ===== +
 Titik Tengah X 350, Titik Tengah Y : 306
 Titik 1 (289,13) =>Jarak dari COG : 299.282475263755
 Titik 2 (477,45) =>Jarak dari COG : 290.258505474

```

===== NILAI RGB Pixels
===== +
Titik Tengah X 354, Titik Tengah Y : 308
Titik 1 (281,28) =>Jarak dari COG : 289.359637821172
Titik 2 (488,46) =>Jarak dari COG : 294.278779391243
===== NILAI RGB Pixels
===== +
Titik Tengah X 358, Titik Tengah Y : 311
Titik 1 (283,32) =>Jarak dari COG : 288.904828620084
Titik 2 (499,51) =>Jarak dari COG : 295.771871549679
===== NILAI RGB Pixels
===== +
Titik Tengah X 357, Titik Tengah Y : 307
Titik 1 (284,33) =>Jarak dari COG : 283.557754258282
Titik 2 (497,47) =>Jarak dari COG : 295.296461204668
===== NILAI RGB Pixels
===== +
Titik Tengah X 361, Titik Tengah Y : 308
Titik 1 (290,33) =>Jarak dari COG : 284.017605088135
Titik 2 (507,53) =>Jarak dari COG : 293.838390956662
===== NILAI RGB Pixels
===== +
Titik Tengah X 361, Titik Tengah Y : 307
Titik 1 (291,21) =>Jarak dari COG : 294.441844852256
Titik 2 (493,47) =>Jarak dari COG : 291.588751497722
===== NILAI RGB Pixels
===== +
Titik Tengah X 372, Titik Tengah Y : 336
Titik 1 (464,54) =>Jarak dari COG : 296.627712798383
Titik 2 (32,272) =>Jarak dari COG : 345.971097058699
===== NILAI RGB Pixels
=====
Titik Tengah X 346, Titik Tengah Y : 259
Titik 1 (192,62) =>Jarak dari COG : 250.049995001
Titik 2 (449,1) =>Jarak dari COG : 277.800287976813
Titik 3 (274,1) =>Jarak dari COG : 267.858171426597
===== NILAI RGB Pixels
===== +
Titik Tengah X 305, Titik Tengah Y : 280
Titik 1 (391,29) =>Jarak dari COG : 265.324329830493
Titik 2 (224,26) =>Jarak dari COG : 266.60270066149
===== NILAI RGB Pixels
===== +
Titik Tengah X 296, Titik Tengah Y : 283
Titik 1 (377,39) =>Jarak dari COG : 257.093368253637
Titik 2 (210,34) =>Jarak dari COG : 263.43310346272
===== NILAI RGB Pixels
===== +
Titik Tengah X 310, Titik Tengah Y : 285
Titik 1 (393,41) =>Jarak dari COG : 257.730479377197
Titik 2 (226,37) =>Jarak dari COG : 261.83964558485
===== NILAI RGB Pixels
===== +
Titik Tengah X 309, Titik Tengah Y : 294
Titik 1 (384,49) =>Jarak dari COG : 256.222559506379
Titik 2 (214,60) =>Jarak dari COG : 252.549005145536

```

Background memiliki obstacles 1

Deteksi

```
===== NILAI RGB Pixels
=====
Titik Tengah X 290,Titik Tengah Y : 326
Titik 1 (72,191) =>Jarak dari COG : 256.415678147808
Titik 2 (473,147) =>Jarak dari COG : 255.988280981767
Titik 3 (272,76) =>Jarak dari COG : 250.647162361755
===== NILAI RGB Pixels
=====
Titik Tengah X 314,Titik Tengah Y : 294
Titik 1 (296,29) =>Jarak dari COG : 265.610617257669
Titik 2 (104,125) =>Jarak dari COG : 269.557044055614
Titik 3 (517,95) =>Jarak dari COG : 284.270997465447
===== NILAI RGB Pixels
=====
Titik Tengah X 348,Titik Tengah Y : 281
Titik 1 (356,3) =>Jarak dari COG : 278.115084092898
Titik 2 (158,69) =>Jarak dari COG : 284.682279041039
Titik 3 (575,49) =>Jarak dari COG : 324.581268714016
===== NILAI RGB Pixels
=====
Titik Tengah X 345,Titik Tengah Y : 282
Titik 1 (354,2) =>Jarak dari COG : 280.144605516508
Titik 2 (156,73) =>Jarak dari COG : 281.783604916965
Titik 3 (574,59) =>Jarak dari COG : 319.640422975568
===== NILAI RGB Pixels
=====
Titik Tengah X 354,Titik Tengah Y : 289
Titik 1 (356,31) =>Jarak dari COG : 258.00775182153
Titik 2 (158,124) =>Jarak dari COG : 256.204996048086
Titik 3 (563,99) =>Jarak dari COG : 282.455306199052
===== NILAI RGB Pixels
=====
Titik Tengah X 346,Titik Tengah Y : 296
Titik 1 (150,133) =>Jarak dari COG : 254.921556562014
Titik 2 (551,107) =>Jarak dari COG : 278.829697127117
Titik 3 (344,41) =>Jarak dari COG : 255.007843016641
===== NILAI RGB Pixels
=====
Titik Tengah X 341,Titik Tengah Y : 294
Titik 1 (344,36) =>Jarak dari COG : 258.017441270934
Titik 2 (150,123) =>Jarak dari COG : 256.363023854845
Titik 3 (541,99) =>Jarak dari COG : 279.329554469268
```



```

===== NILAI RGB Pixels
=====
Titik Tengah X 340,Titik Tengah Y : 293
Titik 1 (356,24) =>Jarak dari COG : 269.475416318446
Titik 2 (148,123) =>Jarak dari COG : 256.444925861285
Titik 3 (541,101) =>Jarak dari COG : 277.965825237564
===== NILAI RGB Pixels
=====
Titik Tengah X 355,Titik Tengah Y : 317
Titik 1 (326,61) =>Jarak dari COG : 257.637342013925
Titik 2 (124,216) =>Jarak dari COG : 252.115053100762
Titik 3 (571,154) =>Jarak dari COG : 270.601182554696
===== NILAI RGB Pixels
=====
Titik Tengah X 328,Titik Tengah Y : 312
Titik 1 (88,225) =>Jarak dari COG : 255.282196794058
Titik 2 (541,146) =>Jarak dari COG : 270.046292327816
Titik 3 (302,53) =>Jarak dari COG : 260.301747977227

===== NILAI RGB Pixels
=====
Titik Tengah X 315,Titik Tengah Y : 311
Titik 1 (74,208) =>Jarak dari COG : 262.087771557545
Titik 2 (521,136) =>Jarak dari COG : 270.29798371427
Titik 3 (282,48) =>Jarak dari COG : 265.062256837898
===== NILAI RGB Pixels
=====
Titik Tengah X 311,Titik Tengah Y : 324
Titik 1 (70,245) =>Jarak dari COG : 253.617822717568
Titik 2 (533,168) =>Jarak dari COG : 271.33005731028
Titik 3 (276,72) =>Jarak dari COG : 254.418945835407
===== NILAI RGB Pixels
=====
Titik Tengah X 320,Titik Tengah Y : 323
Titik 1 (82,239) =>Jarak dari COG : 252.388589282479
Titik 2 (527,162) =>Jarak dari COG : 262.240347772802
Titik 3 (296,66) =>Jarak dari COG : 258.118189982806
===== NILAI RGB Pixels
=====
Titik Tengah X 318,Titik Tengah Y : 321
Titik 1 (82,237) =>Jarak dari COG : 250.503492989619
Titik 2 (525,161) =>Jarak dari COG : 261.627597932634
Titik 3 (294,66) =>Jarak dari COG : 256.126921661898
===== NILAI RGB Pixels
=====
Titik Tengah X 322,Titik Tengah Y : 318
Titik 1 (88,229) =>Jarak dari COG : 250.353749722268
Titik 2 (547,166) =>Jarak dari COG : 271.5308453933
Titik 3 (304,61) =>Jarak dari COG : 257.629579047127
===== NILAI RGB Pixels
=====
Titik Tengah X 320,Titik Tengah Y : 322
Titik 1 (86,227) =>Jarak dari COG : 252.549005145536
Titik 2 (545,166) =>Jarak dari COG : 273.790065561189
Titik 3 (303,64) =>Jarak dari COG : 258.5594709153
===== NILAI RGB Pixels
=====
Titik Tengah X 323,Titik Tengah Y : 322
Titik 1 (88,231) =>Jarak dari COG : 252.003968222725
Titik 2 (523,156) =>Jarak dari COG : 259.915370842126
Titik 3 (302,64) =>Jarak dari COG : 258.853240273326

```

```
===== NILAI RGB Pixels
=====
Titik Tengah X 323,Titik Tengah Y : 322
Titik 1 (87,237) =>Jarak dari COG : 250.840586827571
Titik 2 (547,168) =>Jarak dari COG : 271.830829745266
Titik 3 (302,66) =>Jarak dari COG : 256.859883983467
===== NILAI RGB Pixels
=====
Titik Tengah X 325,Titik Tengah Y : 322
Titik 1 (92,231) =>Jarak dari COG : 250.139960821937
Titik 2 (537,166) =>Jarak dari COG : 263.210942021794
Titik 3 (308,66) =>Jarak dari COG : 256.56383221335
===== NILAI RGB Pixels
=====
Titik Tengah X 328,Titik Tengah Y : 321
Titik 1 (90,229) =>Jarak dari COG : 255.162693197889
Titik 2 (563,176) =>Jarak dari COG : 276.134025429682
Titik 3 (312,62) =>Jarak dari COG : 259.493737882054
```

Tidak Detedksi

```
===== NILAI RGB Pixels
=====
Titik Tengah X 251,Titik Tengah Y : 45
===== NILAI RGB Pixels
=====
Titik Tengah X 465,Titik Tengah Y : 74
===== NILAI RGB Pixels
=====
Titik Tengah X 441,Titik Tengah Y : 164
===== NILAI RGB Pixels
=====
Titik Tengah X 437,Titik Tengah Y : 202
===== NILAI RGB Pixels
=====
Titik Tengah X 407,Titik Tengah Y : 186
===== NILAI RGB Pixels
=====
Titik Tengah X 402,Titik Tengah Y : 214
===== NILAI RGB Pixels
=====
Titik Tengah X 431,Titik Tengah Y : 278
===== NILAI RGB Pixels
=====
Titik Tengah X 422,Titik Tengah Y : 295
===== NILAI RGB Pixels
=====
Titik Tengah X 423,Titik Tengah Y : 294
===== NILAI RGB Pixels
=====
Titik Tengah X 430,Titik Tengah Y : 279
===== NILAI RGB Pixels
=====
Titik Tengah X 396,Titik Tengah Y : 235
===== NILAI RGB Pixels
=====
Titik Tengah X 404,Titik Tengah Y : 217
===== NILAI RGB Pixels
=====
Titik Tengah X 421,Titik Tengah Y : 220
```

```

===== NILAI RGB Pixels
=====
Titik Tengah X 424,Titik Tengah Y : 226
===== NILAI RGB Pixels
=====
Titik Tengah X 425,Titik Tengah Y : 228
===== NILAI RGB Pixels
=====
Titik Tengah X 424,Titik Tengah Y : 227
===== NILAI RGB Pixels
=====
Titik Tengah X 428,Titik Tengah Y : 221
===== NILAI RGB Pixels
=====
Titik Tengah X 435,Titik Tengah Y : 222
===== NILAI RGB Pixels
=====
Titik Tengah X 455,Titik Tengah Y : 223
===== NILAI RGB Pixels
=====
Titik Tengah X 455,Titik Tengah Y : 225

```

Mengecil

```

===== NILAI RGB Pixels
=====
Titik Tengah X 464,Titik Tengah Y : 350
Titik 1 (490,92) =>Jarak dari COG : 259.306768133807
===== NILAI RGB Pixels
=====
Titik Tengah X 473,Titik Tengah Y : 355
Titik 1 (506,95) =>Jarak dari COG : 262.085863792765
===== NILAI RGB Pixels
=====
Titik Tengah X 460,Titik Tengah Y : 353
Titik 1 (478,100) =>Jarak dari COG : 253.639507963566
===== NILAI RGB Pixels
=====
Titik Tengah X 465,Titik Tengah Y : 354
Titik 1 (488,103) =>Jarak dari COG : 252.051582022411
===== NILAI RGB Pixels
=====
Titik Tengah X 465,Titik Tengah Y : 353
Titik 1 (498,91) =>Jarak dari COG : 264.070066459642
===== NILAI RGB Pixels
=====
Titik Tengah X 464,Titik Tengah Y : 353
Titik 1 (490,104) =>Jarak dari COG : 250.353749722268
===== NILAI RGB Pixels
=====
Titik Tengah X 465,Titik Tengah Y : 354
Titik 1 (494,104) =>Jarak dari COG : 251.67637950352
===== NILAI RGB Pixels
=====
Titik Tengah X 464,Titik Tengah Y : 356
Titik 1 (500,90) =>Jarak dari COG : 268.425036090153
===== NILAI RGB Pixels
=====
Titik Tengah X 464,Titik Tengah Y : 352
Titik 1 (498,91) =>Jarak dari COG : 263.205243108871

```

```

===== NILAI RGB Pixels
=====
Titik Tengah X 465,Titik Tengah Y : 354
Titik 1 (492,105) =>Jarak dari COG : 250.459577576902
===== NILAI RGB Pixels
=====
Titik Tengah X 466,Titik Tengah Y : 354
Titik 1 (504,91) =>Jarak dari COG : 265.731067058408
===== NILAI RGB Pixels
=====
Titik Tengah X 470,Titik Tengah Y : 352
Titik 1 (502,102) =>Jarak dari COG : 252.039679415762
===== NILAI RGB Pixels
=====
Titik Tengah X 466,Titik Tengah Y : 354
Titik 1 (508,89) =>Jarak dari COG : 268.307659227239
===== NILAI RGB Pixels
=====
Titik Tengah X 386,Titik Tengah Y : 305
Titik 1 (408,9) =>Jarak dari COG : 296.816441593117
===== NILAI RGB Pixels
=====
Titik Tengah X 399,Titik Tengah Y : 308
Titik 1 (430,7) =>Jarak dari COG : 302.592134729242
===== NILAI RGB Pixels
=====
Titik Tengah X 403,Titik Tengah Y : 311
Titik 1 (440,1) =>Jarak dari COG : 312.200256245891
===== NILAI RGB Pixels
=====
Titik Tengah X 439,Titik Tengah Y : 333
Titik 1 (514,30) =>Jarak dari COG : 312.14419744727
===== NILAI RGB Pixels
=====
Titik Tengah X 454,Titik Tengah Y : 347
Titik 1 (518,69) =>Jarak dari COG : 285.271800218669
===== NILAI RGB Pixels
=====
Titik Tengah X 460,Titik Tengah Y : 365
Titik 1 (504,115) =>Jarak dari COG : 253.842470835753
===== NILAI RGB Pixels
=====
Titik Tengah X 458,Titik Tengah Y : 359
Titik 1 (510,84) =>Jarak dari COG : 279.873185568035

```

Membesar

```

===== NILAI RGB Pixels
=====
Titik Tengah X 314,Titik Tengah Y : 305
Titik 1 (491,119) =>Jarak dari COG : 256.758641529355
Titik 2 (332,54) =>Jarak dari COG : 251.644590643232
===== NILAI RGB Pixels
=====
Titik Tengah X 311,Titik Tengah Y : 295
Titik 1 (489,99) =>Jarak dari COG : 264.764045897475
Titik 2 (318,43) =>Jarak dari COG : 252.097203475168
===== NILAI RGB Pixels
=====
Titik Tengah X 310,Titik Tengah Y : 291
Titik 1 (495,87) =>Jarak dari COG : 275.392447245744
Titik 2 (332,21) =>Jarak dari COG : 270.8948135347

```

```

===== NILAI RGB Pixels
=====
Titik Tengah X 305,Titik Tengah Y : 288
Titik 1 (485,81) =>Jarak dari COG : 274.315511774307
Titik 2 (312,26) =>Jarak dari COG : 262.093494768565
===== NILAI RGB Pixels
=====
Titik Tengah X 314,Titik Tengah Y : 289
Titik 1 (503,97) =>Jarak dari COG : 269.416035157524
Titik 2 (320,33) =>Jarak dari COG : 256.070302846699
===== NILAI RGB Pixels
=====
Titik Tengah X 315,Titik Tengah Y : 289
Titik 1 (509,97) =>Jarak dari COG : 272.946881279124
Titik 2 (336,27) =>Jarak dari COG : 262.840255668724
===== NILAI RGB Pixels
=====
Titik Tengah X 316,Titik Tengah Y : 289
Titik 1 (511,101) =>Jarak dari COG : 270.867126096911
Titik 2 (354,20) =>Jarak dari COG : 271.67075661543
===== NILAI RGB Pixels
=====
Titik Tengah X 327,Titik Tengah Y : 289
Titik 1 (511,82) =>Jarak dari COG : 276.956675312223
Titik 2 (328,25) =>Jarak dari COG : 264.0018939326
===== NILAI RGB Pixels
=====
Titik Tengah X 329,Titik Tengah Y : 295
Titik 1 (509,85) =>Jarak dari COG : 276.586333718787
Titik 2 (324,33) =>Jarak dari COG : 262.047705580492
===== NILAI RGB Pixels
=====
Titik Tengah X 334,Titik Tengah Y : 296
Titik 1 (521,89) =>Jarak dari COG : 278.958778316797
Titik 2 (350,25) =>Jarak dari COG : 271.47191383272
===== NILAI RGB Pixels
=====
Titik Tengah X 337,Titik Tengah Y : 298
Titik 1 (523,93) =>Jarak dari COG : 276.804985504235
Titik 2 (342,37) =>Jarak dari COG : 261.047888327027
===== NILAI RGB Pixels
=====
Titik Tengah X 337,Titik Tengah Y : 295
Titik 1 (527,92) =>Jarak dari COG : 278.044960393099
Titik 2 (352,25) =>Jarak dari COG : 270.416345659799
===== NILAI RGB Pixels
=====
Titik Tengah X 337,Titik Tengah Y : 290
Titik 1 (529,90) =>Jarak dari COG : 277.243575218615
Titik 2 (342,33) =>Jarak dari COG : 257.0486335307
===== NILAI RGB Pixels
=====
Titik Tengah X 400,Titik Tengah Y : 296
Titik 1 (302,38) =>Jarak dari COG : 275.98550686585
Titik 2 (597,99) =>Jarak dari COG : 278.6000717875
===== NILAI RGB Pixels
=====
Titik Tengah X 403,Titik Tengah Y : 287
Titik 1 (302,39) =>Jarak dari COG : 267.777893038242
Titik 2 (589,95) =>Jarak dari COG : 267.320032919346
===== NILAI RGB Pixels
=====
Titik Tengah X 407,Titik Tengah Y : 289

```

```

Titik 1 (304,40) =>Jarak dari COG : 269.462427807663
Titik 2 (597,96) =>Jarak dari COG : 270.830205110139
===== NILAI RGB Pixels
=====
Titik Tengah X 404,Titik Tengah Y : 289
Titik 1 (298,44) =>Jarak dari COG : 266.947560393423
Titik 2 (597,98) =>Jarak dari COG : 271.532686798477
===== NILAI RGB Pixels
=====
Titik Tengah X 407,Titik Tengah Y : 287
Titik 1 (300,42) =>Jarak dari COG : 267.346217478385
Titik 2 (597,96) =>Jarak dari COG : 269.408611592132
===== NILAI RGB Pixels
=====
Titik Tengah X 401,Titik Tengah Y : 296
Titik 1 (306,45) =>Jarak dari COG : 268.376601066486
Titik 2 (603,102) =>Jarak dari COG : 280.071419462965
===== NILAI RGB Pixels
=====
Titik Tengah X 398,Titik Tengah Y : 296
Titik 1 (302,48) =>Jarak dari COG : 265.932322217515
Titik 2 (595,100) =>Jarak dari COG : 277.893864631805

```

Background berwarna putih

Navigasi deteksi (D)

```

===== NILAI RGB Pixels
===== D
Titik Tengah X 336,Titik Tengah Y : 308
Titik 1 (84,227) =>Jarak dari COG : 264.697941057349
Titik 2 (591,239) =>Jarak dari COG : 264.170399553016
Titik 3 (413,39) =>Jarak dari COG : 279.803502479865
Titik 4 (248,27) =>Jarak dari COG : 294.457127609437
Titik 5 (154,96) =>Jarak dari COG : 279.406513882551
===== NILAI RGB Pixels
===== D
Titik Tengah X 328,Titik Tengah Y : 306
Titik 1 (80,233) =>Jarak dari COG : 258.52079220055
Titik 2 (589,246) =>Jarak dari COG : 267.807766877662
Titik 3 (386,35) =>Jarak dari COG : 277.137150162153
Titik 4 (252,25) =>Jarak dari COG : 291.09620402884
Titik 5 (152,95) =>Jarak dari COG : 274.767174167512
===== NILAI RGB Pixels
===== D
Titik Tengah X 330,Titik Tengah Y : 311
Titik 1 (80,232) =>Jarak dari COG : 262.185049154219
Titik 2 (587,243) =>Jarak dari COG : 265.843939182371
Titik 3 (413,43) =>Jarak dari COG : 280.558371823049
Titik 4 (234,40) =>Jarak dari COG : 287.501304344867
Titik 5 (150,98) =>Jarak dari COG : 278.870937890631

```

```

===== NILAI RGB Pixels
=====
Titik Tengah X 331,Titik Tengah Y : 310
Titik 1 (150,98) =>Jarak dari COG : 278.756165851089
Titik 2 (80,229) =>Jarak dari COG : 263.746090018411
Titik 3 (587,245) =>Jarak dari COG : 264.123077371138
Titik 4 (384,36) =>Jarak dari COG : 279.078841906727
Titik 5 (232,40) =>Jarak dari COG : 287.577815556068
===== NILAI RGB Pixels
=====
Titik Tengah X 330,Titik Tengah Y : 309
Titik 1 (80,234) =>Jarak dari COG : 261.007662722764
Titik 2 (585,243) =>Jarak dari COG : 263.40273347101
Titik 3 (382,37) =>Jarak dari COG : 276.925982890736
Titik 4 (230,42) =>Jarak dari COG : 285.112258592997
Titik 5 (148,100) =>Jarak dari COG : 277.137150162153
===== NILAI RGB Pixels
=====
Titik Tengah X 326,Titik Tengah Y : 309
Titik 1 (78,234) =>Jarak dari COG : 259.092647522078
Titik 2 (585,243) =>Jarak dari COG : 267.277009860556
Titik 3 (384,37) =>Jarak dari COG : 278.115084092898
Titik 4 (244,29) =>Jarak dari COG : 291.760175486649
Titik 5 (150,98) =>Jarak dari COG : 274.767174167512
===== NILAI RGB Pixels
=====
Titik Tengah X 329,Titik Tengah Y : 310
Titik 1 (148,100) =>Jarak dari COG : 277.238164760915
Titik 2 (78,229) =>Jarak dari COG : 263.746090018411
Titik 3 (587,243) =>Jarak dari COG : 266.557686064386
Titik 4 (407,39) =>Jarak dari COG : 282.001773044071
Titik 5 (248,27) =>Jarak dari COG : 294.363720590701
===== NILAI RGB Pixels
=====
Titik Tengah X 327,Titik Tengah Y : 309
Titik 1 (148,99) =>Jarak dari COG : 275.936586918082
Titik 2 (76,234) =>Jarak dari COG : 261.965646602756
Titik 3 (585,241) =>Jarak dari COG : 266.810794384335
Titik 4 (384,37) =>Jarak dari COG : 277.908258243615
Titik 5 (244,29) =>Jarak dari COG : 292.042805081721
===== NILAI RGB Pixels
=====
Titik Tengah X 332,Titik Tengah Y : 312
Titik 1 (146,104) =>Jarak dari COG : 279.034048101661
Titik 2 (80,231) =>Jarak dari COG : 264.697941057349
Titik 3 (589,245) =>Jarak dari COG : 265.589909446876
Titik 4 (409,41) =>Jarak dari COG : 281.726818034776
Titik 5 (232,42) =>Jarak dari COG : 287.923600977759

===== NILAI RGB Pixels
=====
Titik Tengah X 317,Titik Tengah Y : 318
Titik 1 (138,134) =>Jarak dari COG : 256.704109822963
Titik 2 (64,287) =>Jarak dari COG : 254.892134048895
Titik 3 (573,245) =>Jarak dari COG : 266.204808371299
Titik 4 (385,53) =>Jarak dari COG : 273.585452829642
Titik 5 (228,74) =>Jarak dari COG : 259.724854413281
===== NILAI RGB Pixels
=====
Titik Tengah X 316,Titik Tengah Y : 319
Titik 1 (140,129) =>Jarak dari COG : 258.990347310474
Titik 2 (62,286) =>Jarak dari COG : 256.134730171447
Titik 3 (573,246) =>Jarak dari COG : 267.166614680802

```

```

Titik 4 (389,53) =>Jarak dari COG : 275.835095664058
Titik 5 (228,69) =>Jarak dari COG : 265.035846632111
===== NILAI RGB Pixels
=====
Titik Tengah X 349,Titik Tengah Y : 295
Titik 1 (144,100) =>Jarak dari COG : 282.931087015902
Titik 2 (76,248) =>Jarak dari COG : 277.01624501101
Titik 3 (609,189) =>Jarak dari COG : 280.777491975407
Titik 4 (423,13) =>Jarak dari COG : 291.547594742265
Titik 5 (268,18) =>Jarak dari COG : 288.600069300061
===== NILAI RGB Pixels
=====
Titik Tengah X 345,Titik Tengah Y : 299
Titik 1 (142,115) =>Jarak dari COG : 273.979926271981
Titik 2 (78,262) =>Jarak dari COG : 269.551479313322
Titik 3 (603,197) =>Jarak dari COG : 277.431072520725
Titik 4 (411,21) =>Jarak dari COG : 285.72714256787
Titik 5 (258,31) =>Jarak dari COG : 281.767634763115
===== NILAI RGB Pixels
=====
Titik Tengah X 319,Titik Tengah Y : 323
Titik 1 (130,126) =>Jarak dari COG : 273.001831495688
Titik 2 (64,263) =>Jarak dari COG : 261.963737948595
Titik 3 (617,293) =>Jarak dari COG : 299.506260368627
Titik 4 (374,63) =>Jarak dari COG : 265.753645318366
Titik 5 (212,67) =>Jarak dari COG : 277.461709069918
===== NILAI RGB Pixels
=====
Titik Tengah X 320,Titik Tengah Y : 327
Titik 1 (124,135) =>Jarak dari COG : 274.37201023428
Titik 2 (60,265) =>Jarak dari COG : 267.290104568052
Titik 3 (621,291) =>Jarak dari COG : 303.145179740665
Titik 4 (369,66) =>Jarak dari COG : 265.55978611228
Titik 5 (219,69) =>Jarak dari COG : 277.064974329127
===== NILAI RGB Pixels
=====
Titik Tengah X 314,Titik Tengah Y : 325
Titik 1 (114,140) =>Jarak dari COG : 272.442654516506
Titik 2 (54,277) =>Jarak dari COG : 264.393645914572
Titik 3 (619,291) =>Jarak dari COG : 306.88923083093
Titik 4 (350,66) =>Jarak dari COG : 261.489961566405
Titik 5 (200,73) =>Jarak dari COG : 276.586333718787
===== NILAI RGB Pixels
=====
Titik Tengah X 313,Titik Tengah Y : 332
Titik 1 (38,291) =>Jarak dari COG : 278.0395655298
Titik 2 (595,281) =>Jarak dari COG : 286.574597618142
Titik 3 (354,71) =>Jarak dari COG : 264.200681301165
Titik 4 (190,87) =>Jarak dari COG : 274.142298815779
Titik 5 (120,143) =>Jarak dari COG : 270.129598526337
===== NILAI RGB Pixels
=====
Titik Tengah X 398,Titik Tengah Y : 285
Titik 1 (204,62) =>Jarak dari COG : 295.575709421461
Titik 2 (124,195) =>Jarak dari COG : 288.402496521788
Titik 3 (631,181) =>Jarak dari COG : 255.156814527851
Titik 4 (465,1) =>Jarak dari COG : 291.796161729383
Titik 5 (302,2) =>Jarak dari COG : 298.839421763595
===== NILAI RGB Pixels
=====
Titik Tengah X 342,Titik Tengah Y : 308
Titik 1 (164,103) =>Jarak dari COG : 271.49401466699
Titik 2 (86,272) =>Jarak dari COG : 258.518858112904

```


Titik 3 (599,241) =>Jarak dari COG : 265.589909446876
Titik 4 (421,31) =>Jarak dari COG : 288.045135352083
Titik 5 (232,53) =>Jarak dari COG : 277.713881539976
===== NILAI RGB Pixels
===== D
Titik Tengah X 341,Titik Tengah Y : 309
Titik 1 (156,121) =>Jarak dari COG : 263.759360023488
Titik 2 (90,280) =>Jarak dari COG : 252.669744924081
Titik 3 (595,241) =>Jarak dari COG : 262.944861140126
Titik 4 (421,35) =>Jarak dari COG : 285.440011210762
Titik 5 (249,38) =>Jarak dari COG : 286.190495998732

Navigasi tidak deteksi (D!)

===== NILAI RGB Pixels
===== D!
Titik Tengah X 551,Titik Tengah Y : 302
===== NILAI RGB Pixels
===== D!
Titik Tengah X 550,Titik Tengah Y : 302
===== NILAI RGB Pixels
===== D!
Titik Tengah X 564,Titik Tengah Y : 327
===== NILAI RGB Pixels
===== D!
Titik Tengah X 567,Titik Tengah Y : 324
===== NILAI RGB Pixels
===== D!
Titik Tengah X 575,Titik Tengah Y : 319
===== NILAI RGB Pixels
===== D!
Titik Tengah X 574,Titik Tengah Y : 319
===== NILAI RGB Pixels
===== D!
Titik Tengah X 523,Titik Tengah Y : 311
===== NILAI RGB Pixels
===== D!
Titik Tengah X 469,Titik Tengah Y : 309
===== NILAI RGB Pixels
===== D!
Titik Tengah X 451,Titik Tengah Y : 307
===== NILAI RGB Pixels
===== D!
Titik Tengah X 460,Titik Tengah Y : 309
===== NILAI RGB Pixels
===== D!
Titik Tengah X 422,Titik Tengah Y : 319
===== NILAI RGB Pixels
===== D!
Titik Tengah X 389,Titik Tengah Y : 325
===== NILAI RGB Pixels
===== D!
Titik Tengah X 387,Titik Tengah Y : 333
===== NILAI RGB Pixels
===== D!
Titik Tengah X 383,Titik Tengah Y : 334
===== NILAI RGB Pixels
===== D!
Titik Tengah X 361,Titik Tengah Y : 331
===== NILAI RGB Pixels
===== D!

```

Titik Tengah X 363,Titik Tengah Y : 334
===== NILAI RGB Pixels
===== D!
Titik Tengah X 348,Titik Tengah Y : 324
===== NILAI RGB Pixels
===== D!
Titik Tengah X 347,Titik Tengah Y : 305
===== NILAI RGB Pixels
===== D!
Titik Tengah X 364,Titik Tengah Y : 306
===== NILAI RGB Pixels
===== D!
Titik Tengah X 366,Titik Tengah Y : 307

```

Navigasi Mengecil (-)

```

===== NILAI RGB Pixels
===== -
Titik Tengah X 339,Titik Tengah Y : 294
Titik 1 (340,20) =>Jarak dari COG : 274.001824811442
===== NILAI RGB Pixels
===== -
Titik Tengah X 339,Titik Tengah Y : 295
Titik 1 (340,20) =>Jarak dari COG : 275.001818175808
===== NILAI RGB Pixels
===== -
Titik Tengah X 336,Titik Tengah Y : 296
Titik 1 (336,23) =>Jarak dari COG : 273
===== NILAI RGB Pixels
===== -
Titik Tengah X 335,Titik Tengah Y : 297
Titik 1 (332,26) =>Jarak dari COG : 271.016604657353
===== NILAI RGB Pixels
===== -
Titik Tengah X 333,Titik Tengah Y : 299
Titik 1 (332,27) =>Jarak dari COG : 272.001838229083
===== NILAI RGB Pixels
===== -
Titik Tengah X 333,Titik Tengah Y : 299
Titik 1 (330,32) =>Jarak dari COG : 267.01685340068
===== NILAI RGB Pixels
===== -
Titik Tengah X 325,Titik Tengah Y : 302
Titik 1 (320,34) =>Jarak dari COG : 268.046637733063
===== NILAI RGB Pixels
===== -
Titik Tengah X 326,Titik Tengah Y : 301
Titik 1 (322,39) =>Jarak dari COG : 262.030532572065
===== NILAI RGB Pixels
===== -
Titik Tengah X 326,Titik Tengah Y : 301
Titik 1 (324,33) =>Jarak dari COG : 268.007462582668
===== NILAI RGB Pixels
===== -
Titik Tengah X 321,Titik Tengah Y : 305
Titik 1 (318,40) =>Jarak dari COG : 265.016980588037
===== NILAI RGB Pixels
===== -
Titik Tengah X 318,Titik Tengah Y : 306
Titik 1 (314,40) =>Jarak dari COG : 266.030073487942

```

```

===== NILAI RGB Pixels
=====
Titik Tengah X 322,Titik Tengah Y : 306
Titik 1 (320,39) =>Jarak dari COG : 267.007490531633
===== NILAI RGB Pixels
=====
Titik Tengah X 323,Titik Tengah Y : 305
Titik 1 (320,40) =>Jarak dari COG : 265.016980588037
===== NILAI RGB Pixels
=====
Titik Tengah X 321,Titik Tengah Y : 306
Titik 1 (316,49) =>Jarak dari COG : 257.0486335307
===== NILAI RGB Pixels
=====
Titik Tengah X 321,Titik Tengah Y : 306
Titik 1 (318,46) =>Jarak dari COG : 260.017307116276
===== NILAI RGB Pixels
=====
Titik Tengah X 322,Titik Tengah Y : 308
Titik 1 (318,45) =>Jarak dari COG : 263.030416492086
===== NILAI RGB Pixels
=====
Titik Tengah X 320,Titik Tengah Y : 308
Titik 1 (318,48) =>Jarak dari COG : 260.007692193904
===== NILAI RGB Pixels
=====
Titik Tengah X 316,Titik Tengah Y : 308
Titik 1 (312,57) =>Jarak dari COG : 251.031870486598
===== NILAI RGB Pixels
=====
Titik Tengah X 318,Titik Tengah Y : 308
Titik 1 (314,51) =>Jarak dari COG : 257.031126519727
===== NILAI RGB Pixels
=====
Titik Tengah X 318,Titik Tengah Y : 309
Titik 1 (314,56) =>Jarak dari COG : 253.0316185776
===== NILAI RGB Pixels
=====
Titik Tengah X 321,Titik Tengah Y : 308
Titik 1 (318,51) =>Jarak dari COG : 257.017509131187

```

Navigasi Membesar (+)

```

===== NILAI RGB Pixels
=====
Titik Tengah X 279,Titik Tengah Y : 300
Titik 1 (180,45) =>Jarak dari COG : 273.543415201317
Titik 2 (340,50) =>Jarak dari COG : 257.334412778392
===== NILAI RGB Pixels
=====
Titik Tengah X 274,Titik Tengah Y : 297
Titik 1 (168,62) =>Jarak dari COG : 257.800310317889
Titik 2 (375,47) =>Jarak dari COG : 269.631229645232
===== NILAI RGB Pixels
=====
Titik Tengah X 277,Titik Tengah Y : 295
Titik 1 (166,43) =>Jarak dari COG : 275.363396260287
Titik 2 (371,35) =>Jarak dari COG : 276.470613266582
===== NILAI RGB Pixels
=====

```

```

Titik Tengah X 278, Titik Tengah Y : 295
Titik 1 (168,48) =>Jarak dari COG : 270.386760030886
Titik 2 (358,37) =>Jarak dari COG : 270.118492517636
===== NILAI RGB Pixels
===== +
Titik Tengah X 278, Titik Tengah Y : 297
Titik 1 (168,36) =>Jarak dari COG : 283.233119532303
Titik 2 (369,39) =>Jarak dari COG : 273.578142401764
===== NILAI RGB Pixels
===== +
Titik Tengah X 280, Titik Tengah Y : 295
Titik 1 (170,47) =>Jarak dari COG : 271.300571322657
Titik 2 (373,42) =>Jarak dari COG : 269.551479313322
===== NILAI RGB Pixels
===== +
Titik Tengah X 279, Titik Tengah Y : 297
Titik 1 (170,35) =>Jarak dari COG : 283.769272473254
Titik 2 (368,42) =>Jarak dari COG : 270.085171751431
===== NILAI RGB Pixels
===== +
Titik Tengah X 276, Titik Tengah Y : 300
Titik 1 (166,47) =>Jarak dari COG : 275.878596487658
Titik 2 (367,45) =>Jarak dari COG : 270.750807939699
===== NILAI RGB Pixels
===== +
Titik Tengah X 273, Titik Tengah Y : 300
Titik 1 (162,49) =>Jarak dari COG : 274.448537981167
Titik 2 (367,45) =>Jarak dari COG : 271.773803005367
===== NILAI RGB Pixels
===== +
Titik Tengah X 271, Titik Tengah Y : 298
Titik 1 (162,44) =>Jarak dari COG : 276.400072358891
Titik 2 (369,51) =>Jarak dari COG : 265.731067058408
===== NILAI RGB Pixels
===== +
Titik Tengah X 270, Titik Tengah Y : 298
Titik 1 (160,37) =>Jarak dari COG : 283.233119532303
Titik 2 (367,46) =>Jarak dari COG : 270.024073000909
===== NILAI RGB Pixels
===== +
Titik Tengah X 264, Titik Tengah Y : 299
Titik 1 (150,47) =>Jarak dari COG : 276.586333718787
Titik 2 (363,45) =>Jarak dari COG : 272.611445100898
===== NILAI RGB Pixels
===== +
Titik Tengah X 263, Titik Tengah Y : 300
Titik 1 (150,54) =>Jarak dari COG : 270.712024114187
Titik 2 (359,46) =>Jarak dari COG : 271.536369571371
===== NILAI RGB Pixels
===== +
Titik Tengah X 266, Titik Tengah Y : 301
Titik 1 (150,43) =>Jarak dari COG : 282.878065604246
Titik 2 (361,47) =>Jarak dari COG : 271.184439081596
===== NILAI RGB Pixels
===== +
Titik Tengah X 259, Titik Tengah Y : 299
Titik 1 (144,53) =>Jarak dari COG : 271.552941431316
Titik 2 (338,51) =>Jarak dari COG : 260.278696784812
===== NILAI RGB Pixels
===== +
Titik Tengah X 264, Titik Tengah Y : 301

```

```
Titik 1 (152,41) =>Jarak dari COG : 283.097156467528
Titik 2 (361,49) =>Jarak dari COG : 270.024073000909
===== NILAI RGB Pixels
===== +
Titik Tengah X 262,Titik Tengah Y : 300
Titik 1 (148,53) =>Jarak dari COG : 272.038600202251
Titik 2 (354,49) =>Jarak dari COG : 267.329384841996
===== NILAI RGB Pixels
===== +
Titik Tengah X 263,Titik Tengah Y : 303
Titik 1 (148,48) =>Jarak dari COG : 279.732014613987
Titik 2 (359,50) =>Jarak dari COG : 270.601182554696
===== NILAI RGB Pixels
===== +
Titik Tengah X 261,Titik Tengah Y : 305
Titik 1 (146,46) =>Jarak dari COG : 283.383132878441
Titik 2 (357,51) =>Jarak dari COG : 271.536369571371
===== NILAI RGB Pixels
===== +
Titik Tengah X 264,Titik Tengah Y : 303
Titik 1 (152,42) =>Jarak dari COG : 284.015844628429
Titik 2 (367,51) =>Jarak dari COG : 272.237029075767
```

AugmentedRealityProjector.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

using System.Drawing;
using Emgu.CV.Structure;
using Emgu.CV;

namespace handtrackingstereovision
{
    class AugmentedRealityProjector
    {
        private CameraOperator camOp;
        private HandProcessor handP;
        private System.Windows.Forms.Timer timer;
        private System.Windows.Forms.PictureBox pbportviewer;
        private int ii;
        private int ic;

        private int lastIncrementSize;

        private Bitmap[] bitmaps_ar;

        public AugmentedRealityProjector(CameraOperator camOp,
HandProcessor handP)
        {
            this.camOp = camOp;
            this.handP = handP;
            timer = new System.Windows.Forms.Timer();
            this.timer.Tick += new System.EventHandler(this.timerTick);
            ii = 1;
            ic = 0;
            bitmaps_ar = new Bitmap[9];

            for(int i=0;i<bitmaps_ar.Length;i++)
            {
                Bitmap bitmap = new Bitmap("D:\\AnimAR\\" + ii.ToString() +
".png");
                bitmaps_ar[i] = bitmap;
                ii += 3;
            }
        }

        public void timerTick(object sender, EventArgs e)
        {
            MCvFont font = new
MCvFont(Emgu.CV.CvEnum.FONT.CV_FONT_HERSHEY_DUPLEX, 5d, 5d);
            Image<Bgr, Byte> currentFrame = camOp.getFrameImage();

            Bitmap bitmap = bitmaps_ar[ic]; // new Bitmap("D:\\AnimAR\\" +
ii.ToString() + ".png");

            ii++;
            if (ii % 3==0)
```

```

    {
        ic++;
        if (ic==9)
        {
            ic = 0;
        }
    }

    Image<Bgra, Byte> bola = new Image<Bgra, byte>(bitmap);

    int val = handP.GetIncreasedSize() * 30;
    lastIncrementSize += val;

    if (handP.GetIncreasedSize() == 0)
    {
        lastIncrementSize = 0;
    }

    handP.SetNormal();

    Image<Bgra, Byte> bolabaru = new Image<Bgra, byte>(100+val,
100+val);
    CvInvoke.cvResize(bola, bolabaru,
Emgu.CV.CvEnum.INTER.CV_INTER_LINEAR);

    Point cog = handP.GetCenterofGravity();
    int half = bolabaru.Width / 2;
    if (currentFrame != null)
    {
        // currentFrame.ToBitmap().MakeTransparent();
        Image<Bgra, Byte> currentFrameA = new Image<Bgra,
byte>(currentFrame.Width, currentFrame.Height);
        // System.Diagnostics.Trace.WriteLine("Leebar:" +
currentFrameA.Width);
        //for (int i = 0; i < currentFrameA.Height; i++)
        //{
        //    for (int j = 0; j < currentFrameA.Width ; j++)
        //    {
        //        //currentFrameA.Data[i, j, 3] = 255;

        //    }
        //}

        int maxwidth = currentFrame.Width;
        int maxheight = currentFrame.Height;
        //currentFrame.Draw("X", ref font,
handP.GetCenterofGravity(), new Bgr(Color.Blue));
        int newX = cog.X - half;
        int newY = cog.Y - half;

        if (!(newX < 0 || newY < 0))
        {
            if (handP.IsHandDetected())
            {

                currentFrameA.ROI = new Rectangle(newX, newY,
bolabaru.Width, bolabaru.Height); //Set the ROI to match img to be copied

```



```

private System.Windows.Forms.PictureBox pbportviewer;

public CameraOperator(int idxdevice, int fps)
{
    timer = new System.Windows.Forms.Timer();
    idxcamera = idxdevice;
    frameRate = fps;
    timer.Interval = 1000 / fps;
    this.timer.Tick += new System.EventHandler(this.timerTick);
    pbportviewer = null;
}

public int getFrameRatePerSecond()
{
    return frameRate;
}

public void setPortViewer(System.Windows.Forms.PictureBox
pbportviewer)
{
    this.pbportviewer = pbportviewer;
}

public void timerTick(object sender, EventArgs e)
{
    OriImage = camera.QueryFrame();
    // OriImage.Save("D:\\test.jpg");
    if (pbportviewer!=null) pbportviewer.Image =
OriImage.ToBitmap();
}

public void startCamera()
{
    camera = new Emgu.CV.Capture(idxcamera);
    timer.Start();
    bStarting = true;
}

public void stopCamera()
{
    timer.Stop();
    timer.Dispose();
    camera.Dispose();
    bStarting = false;
}

public Image<Bgr, Byte> getFrameImage()
{
    if (OriImage != null)
    {
        return OriImage.Copy();
    }
    else
    {
        return null;
    }
}

```

```

    }
}

public Boolean isStarting()
{
    return this.bStarting;
}
}
}

```

HandProcessor.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Windows.Forms;

using System.Drawing;
using Emgu.CV.Structure;
using Emgu.CV;

namespace handtrackingstereovision
{
    class HandProcessor
    {
        SkinCalibrator hsvcalib;
        private System.Windows.Forms.Timer timer;
        private System.Windows.Forms.Label lblResult;
        private Font msgFont;
        private System.Windows.Forms.PictureBox pbportviewer;

        Image<Bgr, Byte> currentFrame;
        Image<Bgr, Byte> currentFrameCopy;

        private Seq<Point> hull;
        private Seq<Point> filteredHull;
        private Seq<MCvConvexityDefect> defects;
        private MCvConvexityDefect[] defectArray;
        private Rectangle handRect;
        private MCvBox2D box;
        private Ellipse ellip;

        private Point cogPt; // center of gravity (COG) of
        contour

        private List<Point> fingerTips;

        int curTime;
        bool isstop;
        private ListBox lbdata;

        private bool isDetected = false;
    }
}

```

```

private int isIncreasedSize = 0;

public HandProcessor(SkinCalibrator hsvcalib,
System.Windows.Forms.Label lblResult)
{
    this.hsvcalib = hsvcalib;
    timer = new System.Windows.Forms.Timer();
    this.timer.Tick += new System.EventHandler(this.timerTick);
    this.lblResult = lblResult;
    fingerTips = new List<Point>();

    isstop = false;

    msgFont = new Font("verdana", 20,
System.Drawing.FontStyle.Bold, GraphicsUnit.Point);

    curTime = 0;
    box = new MCvBox2D();
    ellip = new Ellipse();
}

public void setLogger(ListBox lb)
{
    lbdata = lb;
}

public void setPortViewer(System.Windows.Forms.PictureBox
pbportviewer)
{
    this.pbportviewer = pbportviewer;
}

private void ExtractContourAndHull(Image<Gray, byte> skin)
{
    using (MemStorage storage = new MemStorage())
    {
        Contour<Point> contours =
skin.FindContours(Emgu.CV.CvEnum.CHAIN_APPROX_METHOD.CV_CHAIN_APPROX_SIMPLE
, Emgu.CV.CvEnum.RETR_TYPE.CV_RETR_LIST, storage);
        Contour<Point> biggestContour = null;

        Double Result1 = 0;
        Double Result2 = 0;
        while (contours != null)
        {
            Result1 = contours.Area;
            if (Result1 > Result2)
            {
                Result2 = Result1;
                biggestContour = contours;
            }
            contours = contours.HNext;
        }
    }
}

```

```

        if (biggestContour != null)
        {
            Contour<Point> currentContour =
biggestContour.ApproxPoly(biggestContour.Perimeter * 0.0025, storage);
            //currentFrame.Draw(currentContour, new
Bgr(Color.LimeGreen), 2);
            biggestContour = currentContour;
            extractContourinfo(biggestContour);
            hull =
biggestContour.GetConvexHull(Emgu.CV.CvEnum.ORIENTATION.CV_CLOCKWISE);
            box = biggestContour.GetMinAreaRect();
            PointF[] points = box.GetVertices();

            Point[] ps = new Point[points.Length];
            for (int i = 0; i < points.Length; i++)
                ps[i] = new Point((int)points[i].X,
(int)points[i].Y);

            //currentFrame.DrawPolyline(hull.ToArray(), true, new
Bgr(200, 125, 75), 2);
            //currentFrame.Draw(new CircleF(new
PointF(box.center.X, box.center.Y), 3), new Bgr(200, 125, 75), 2);

            PointF center;
            float radius;

            filteredHull = new Seq<Point>(storage);
            for (int i = 0; i < hull.Total; i++)
            {
                if (Math.Sqrt(Math.Pow(hull[i].X - hull[i + 1].X,
2) + Math.Pow(hull[i].Y - hull[i + 1].Y, 2)) > box.size.Width / 10)
                {
                    filteredHull.Push(hull[i]);
                }
            }

            defects = biggestContour.GetConvexityDefacts(storage,
Emgu.CV.CvEnum.ORIENTATION.CV_CLOCKWISE);
            storage.Dispose();
            defectArray = defects.ToArray();
        }
    }

private void extractContourinfo(Contour<Point> bigCountour)
{
    MCvMoments moments = new MCvMoments();

    CvInvoke.cvMoments(bigCountour, ref moments, 1);

    // pusat gravitasi
    double m00 = CvInvoke.cvGetSpatialMoment(ref moments, 0, 0);

```

```

double m10 = CvInvoke.cvGetSpatialMoment(ref moments, 1, 0);
double m01 = CvInvoke.cvGetSpatialMoment(ref moments, 0, 1);

if (m00 != 0)
{
    // menghitung pusat / COG (Centre Of Gravity)
    int xCenter = (int)Math.Round(m10 / m00) ;
    int yCenter = (int)Math.Round(m01 / m00) ;
    cogPt.X = xCenter;
    cogPt.Y = yCenter;
}

System.Diagnostics.Trace.WriteLine("Titik Tengah X " +
cogPt.X.ToString() + ",Titik Tengah Y : " + cogPt.Y.ToString());

double m11 = CvInvoke.cvGetCentralMoment(ref moments, 1, 1);
double m20 = CvInvoke.cvGetCentralMoment(ref moments, 2, 0);
double m02 = CvInvoke.cvGetCentralMoment(ref moments, 0, 2);
//Cv.GetNormalizedCentralMoment
//  contourAxisAngle = calculateTilt(m11, m20, m02);

/*
if (fingerTips.Count > 0)
{
    int yTotal = 0;
    foreach(Point pt in fingerTips)
        yTotal += pt.Y ;
    int avgYFinger = yTotal/fingerTips.Count ;
    if (avgYFinger > cogPt.Y ) // jari di bawah COG
        contourAxisAngle += 180;
}*/
// contourAxisAngle = 180 - contourAxisAngle;
}

public Point GetCenterofGravity()
{
    return cogPt;
}

private void showEuclideanDistance()
{
    for (int i = 0; i < fingerTips.Count; i++)
    {
        Point pt = fingerTips[i];
        //Menghitung Jarak Antara COG dan Titik-titik di ujung Jari
        double distance = Math.Sqrt(Math.Pow(cogPt.X - pt.X, 2) +
Math.Pow(cogPt.Y - pt.Y, 2));
        if (i < 5)
        {
            System.Diagnostics.Trace.WriteLine("Titik " +
(i+1).ToString() + " (" + pt.X.ToString() + "," + pt.Y.ToString() + ")
=>Jarak dari COG : " + distance.ToString());
        }
        if (!isstop)
        {

```

```

        ///
        lbdata.Items.Add(namedFingers[i].ToString().ToUpper() + "=>Jarak dari COG :
" + distance.ToString());
    }
}

private void DrawDefectsAndCOG()
{
    int fingerNum = 0;
    #region hull drawing
    //for (int i = 0; i < filteredHull.Total; i++)
    //{
    //    PointF hullPoint = new PointF((float)filteredHull[i].X,
    //                                  (float)filteredHull[i].Y);
    //    CircleF hullCircle = new CircleF(hullPoint, 4);
    //    currentFrame.Draw(hullCircle, new Bgr(Color.Aquamarine),
2);
    //}
    #endregion

    if (defects == null) return;
    fingerTips.Clear();

    #region defects drawing
        PointF depthPoint = new PointF((float)cogPt.X,
                                       (float)cogPt.Y);
        CircleF depthCircle = new CircleF(depthPoint, 5f);

        // System.Diagnostics.Trace.WriteLine("Jumlah Defect " +
defects.Total.ToString());

        for (int i = 0; i < defects.Total; i++)
        {
            PointF startPoint = new
PointF((float)defectArray[i].StartPoint.X,
(float)defectArray[i].StartPoint.Y);
            Point startIPoint = new Point(defectArray[i].StartPoint.X,
defectArray[i].StartPoint.Y);

            /* PointF depthPoint = new
PointF((float)defectArray[i].DepthPoint.X,
(float)defectArray[i].DepthPoint.Y);*/
            Point depthIPoint = new Point(cogPt.X, cogPt.Y);

            PointF endPoint = new
PointF((float)defectArray[i].EndPoint.X,
(float)defectArray[i].EndPoint.Y);
            //LineSegment2D startDepthLine = new
LineSegment2D(defectArray[i].StartPoint, defectArray[i].DepthPoint);
            LineSegment2D startDepthLine = new
LineSegment2D(defectArray[i].StartPoint, depthIPoint);

            LineSegment2D depthEndLine = new
LineSegment2D(defectArray[i].DepthPoint, defectArray[i].EndPoint);

```

```

        CircleF startCircle = new CircleF(startPoint, 5f);

        CircleF endCircle = new CircleF(endPoint, 5f);

        if ((startCircle.Center.Y < box.center.Y ||
depthCircle.Center.Y < box.center.Y) && (startCircle.Center.Y <
depthCircle.Center.Y) && (Math.Sqrt(Math.Pow(startCircle.Center.X -
depthCircle.Center.X, 2) + Math.Pow(startCircle.Center.Y -
depthCircle.Center.Y, 2)) > box.size.Height / 6.5))
        {

            double distance = Math.Sqrt(Math.Pow(cogPt.X -
startIPoint.X, 2) + Math.Pow(cogPt.Y - startIPoint.Y, 2));
            if (distance > 250)
            {
                fingerNum++;
                fingerTips.Add(startIPoint);
                currentFrame.Draw(startDepthLine, new
Bgr(Color.Green), 2);
            }

            //currentFrame.Draw(depthEndLine, new
Bgr(Color.Magenta), 2);
        }

        currentFrame.Draw(startCircle, new Bgr(Color.Red), 2);

        //currentFrame.Draw(endCircle, new Bgr(Color.DarkBlue), 4);
    }
    currentFrame.Draw(depthCircle, new Bgr(Color.Yellow), 5);
#endregion

    MCvFont font = new
MCvFont(Emgu.CV.CvEnum.FONT.CV_FONT_HERSHEY_DUPLEX, 5d, 5d);

    if (fingerNum == 3)
    {
        currentFrame.Draw("D", ref font, new Point(50, 150), new
Bgr(Color.Blue));
        isDetected = true;
    }
    else if (fingerNum == 0 )
    {
        currentFrame.Draw("D!", ref font, new Point(50, 150), new
Bgr(Color.Red ));
        isDetected = false;
    }
    else if (fingerNum == 2)
    {
        //double distance = Math.Sqrt(Math.Pow(fingerTips[0].X -
fingerTips[1].X, 2) + Math.Pow(fingerTips[0].Y - fingerTips[1].Y, 2));
        //System.Diagnostics.Trace.WriteLine("Jarak Antara 2 Jari "
+ distance);

        //double dist_threshold = 34;//ambil ketika 2 jari rapat
        // if (distance > dist_threshold)
        //{
            isIncreasedSize = 1;
        }
    }
}

```

```

        currentFrame.Draw("+", ref font, new Point(50, 150), new
Bgr(Color.Yellow));
        isDetected = true;
        //}
    }else if (fingerNum == 1)
    {
        currentFrame.Draw("-", ref font, new Point(50, 150), new
Bgr(Color.Red ));
        isIncreasedSize = -1;
        isDetected = true;
    }
    else if (fingerNum == 4)
    {
        currentFrame.Draw("", ref font, new Point(50, 150), new
Bgr(Color.Red));
        isDetected = false;
    }
    else if (fingerNum == 5)
    {
        currentFrame.Draw("", ref font, new Point(50, 150), new
Bgr(Color.Red));
        isDetected = false;
    }
}

//private IplImage resizePicture(IplImage sourceImage,int scale)
//{
//    IplImage gbrbaru = Cv.CreateImage(new
CvSize(sourceImage.Width/scale, sourceImage.Height/scale ), BitDepth.U8,
3);
//    Cv.Resize(sourceImage, gbrbaru);
//    return gbrbaru;
//}

public void startHandDetection(int fps)
{
    timer.Interval = 1000 / fps;
    timer.Start();
}

public void stopHandDetection()
{
    //timer.Stop();
    isstop = true;
    // timer.Dispose();
}

public void SetNormal()

```



```

    {
        isIncreasedSize = 0;
    }

    public int GetIncreasedSize()
    {
        return isIncreasedSize;
    }

    public bool IsHandDetected()
    {
        return isDetected;
    }

    public void timerTick(object sender, EventArgs e)
    {

        Image<Gray,Byte> skin = hsvcalib.getCalibratedImage();
        Image<Bgr, Byte> oriImg = hsvcalib.getOriImage();

        if (skin!=null)
        {
            currentFrame = oriImg.Copy();
            ExtractContourAndHull(skin);

            DrawDefectsAndCOG();
            showEuclideanDistance();
            pbportviewer.Image = currentFrame.ToBitmap();

        }

    }

}

```

IOSetting.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace handtrackingstereovision
{
    class IOSetting
    {
        public YCrCb readYCrCbSetting(String strfilepath)
        {
            YCrCb ycrcbsetting = new YCrCb();
            int counter = 0;
            string line;
            string strline="";
            // Read the file and display it line by line.
            System.IO.StreamReader file = new
            System.IO.StreamReader(strfilepath);
            while ((line = file.ReadLine()) != null)

```

```

        {
            //Console.WriteLine(line);
            strline = line;
            System.Diagnostics.Trace.WriteLine(line);
            counter++;
        }
        String[] arrset = strline.Split(';');
        String[] val = arrset[0].Split(':');
        // System.Diagnostics.Trace.WriteLine(val[0]);
        // System.Diagnostics.Trace.WriteLine(val[1]);
        ycrcbsetting.MinY = Int16.Parse(val[0]);
        ycrcbsetting.MaxY = Int16.Parse(val[1]);
        val = arrset[1].Split(':');
        ycrcbsetting.MinCr = Int16.Parse(val[0]);
        ycrcbsetting.MaxCr = Int16.Parse(val[1]);
        val = arrset[2].Split(':');
        ycrcbsetting.MinCb = Int16.Parse(val[0]);
        ycrcbsetting.MaxCb = Int16.Parse(val[1]);
        file.Close();
        return ycrcbsetting;
    }

    public void saveSetting(String strfilepath, YCrCb ycrcbsetting)
    {
        String filestr = ycrcbsetting.MinY.ToString() + ":" +
        ycrcbsetting.MaxY.ToString();
        filestr += ";" + ycrcbsetting.MinCr.ToString() + ":" +
        ycrcbsetting.MaxCr.ToString();
        filestr += ";" + ycrcbsetting.MinCb.ToString() + ":" +
        ycrcbsetting.MaxCb.ToString();
        System.IO.File.WriteAllText(strfilepath, filestr );
    }

}

}
}

```

Program.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Windows.Forms;

namespace handtrackingstereovision
{
    static class Program
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main()
        {

```

```
        Application.EnableVisualStyles();
        Application.SetCompatibleTextRenderingDefault(false);
        Application.Run(new FormMain());
    }
}
```

SkinCalibrator.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Windows.Forms;

namespace handtrackingstereovision
{
    static class Program
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new FormMain());
        }
    }
}
```

YCrCb.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace handtrackingstereovision
{
    public class YCrCb
    {
        public int MinY { set; get; }
        public int MaxY { set; get; }
        public int MinCr { set; get; }
        public int MaxCr { set; get; }
        public int MinCb { set; get; }
        public int MaxCb { set; get; }
    }
}
```



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS SRIWIJAYA
FAKULTAS ILMU KOMPUTER

JURUSAN SISTEM KOMPUTER

Jalan Palembang – Prabumulih Km. 32 Indralaya Kabupaten Ogan Ilir Kode Pos 30662

Telepon (0711)7072729, 379249, 581700 Faksimili (0711) 379248, 581710

Pos-el : info@ilkom.unsri.ac.id

FORM PERBAIKAN UJIAN TUGAS AKHIR I

Nama Mahasiswa : Robbyansyah Asmir
NIM : 09111001020
Jurusan : Sistem Komputer
Hari / Tanggal : Sabtu / 18 Oktober 2017
Waktu : 10.00 s.d 12.00
Judul Tugas Akhir : Interaksi Hand Tracking pada Objek Augmented Reality
Menggunakan Stereo Vision
Pembimbing I : Sutarno, M.T.
Pembimbing II : Sri Desy Siswanti, M.T.
Perbaikan/Saran :

* dip for pustaka
* Validasi Manual

Jangka Waktu Perbaikan :

Telah diperbaiki sesuai dengan saran dan koreksi tim penguji ujian komprehensif.

No.	Nama Penguji	Status Penguji	Tanda Tangan
1.	Sutarno, M.T.	Ketua	

Palembang, 18 Oktober 2017
Ketua Jurusan Sistem Komputer,

Rossi Passarella, M.Eng.
NIP 19780611 201012 1 004



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS SRIWIJAYA
FAKULTAS ILMU KOMPUTER

JURUSAN SISTEM KOMPUTER

Jalan Palembang – Prabumulih Km. 32 Indralaya Kabupaten Ogan Ilir Kode Pos 30662

Telepon (0711)7072729, 379249, 581700 Faksimili (0711) 379248, 581710

Pos-el : info@ilkom.unsri.ac.id


FORM PERBAIKAN UJIAN TUGAS AKHIR I

Nama Mahasiswa : Robbyansyah Asmir
NIM : 09111001020
Jurusan : Sistem Komputer
Hari / Tanggal : Sabtu / 18 Oktober 2017
Waktu : 10.00 s.d 12.00
Judul Tugas Akhir : Interaksi Hand Tracking pada Objek Augmented Reality
Menggunakan Stereo Vision
Pembimbing I : Sutarno, M.T.
Pembimbing II : Sri Desy Siswanti, M.T.
Perbaikan/Saran :

- * cara kerja Augmented. → copy paste !! hal 66
- * hal 26. → data citra bergambar.
- * tabel 4.1 → hal 36
tabel 4.2 → hal 37
tabel 4.3 → 37 } penulisan tabel diperbaiki
- * kesimpulan diperbaiki

Jangka Waktu Perbaikan :

Telah diperbaiki sesuai dengan saran dan koreksi tim penguji ujian komprehensif.

No.	Nama Penguji	Status Penguji	Tanda Tangan
1.	Rossi Passarella, M.Eng.	Anggota	

13/17.

Palembang, 18 Oktober 2017
Ketua Jurusan Sistem Komputer,

Rossi Passarella, M.Eng.
NIP 19780611 201012 1 004



**KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS SRIWIJAYA
FAKULTAS ILMU KOMPUTER**

JURUSAN SISTEM KOMPUTER

Jalan Palembang – Prabumulih Km. 32 Indralaya Kabupaten Ogan Ilir Kode Pos 30662

Telepon (0711)7072729, 379249, 581700 Faksimili (0711) 379248, 581710

Pos-el : info@ilkom.unsri.ac.id

FORM PERBAIKAN UJIAN TUGAS AKHIR I

Nama Mahasiswa : Robbyansyah Asmir
NIM : 09111001020
Jurusan : Sistem Komputer
Hari / Tanggal : Sabtu / 18 Oktober 2017
Waktu : 10.00 s.d 12.00
Judul Tugas Akhir : Interaksi Hand Tracking pada Objek Augmented Reality
Menggunakan Stereo Vision
Pembimbing I : Sutarno, M.T.
Pembimbing II : Sri Desy Siswanti, M.T.
Perbaikan/Saran :

Jangka Waktu Perbaikan :

Telah diperbaiki sesuai dengan saran dan koreksi tim penguji ujian komprehensif.

No.	Nama Penguji	Status Penguji	Tanda Tangan
1.	Sri Desy Siswanti, M.T.	Sekretaris	

Palembang, 18 Oktober 2017
Ketua Jurusan Sistem Komputer,

Rossi Passarella, M.Eng.
NIP 19780611 201012 1 004



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS SRIWIJAYA
FAKULTAS ILMU KOMPUTER

JURUSAN SISTEM KOMPUTER

Jalan Palembang – Prabumulih Km. 32 Indralaya Kabupaten Ogan Ilir Kode Pos 30662
Telepon (0711)7072729, 379249, 581700 Faksimili (0711) 379248, 581710
Pos-el : info@ikom.unsri.ac.id

FORM PERBAIKAN UJIAN TUGAS AKHIR I

Nama Mahasiswa : Robbyansyah Asmir
NIM : 09111001020
Jurusan : Sistem Komputer
Hari / Tanggal : Sabtu / 18 Oktober 2017
Waktu : 10.00 s.d 12.00
Judul Tugas Akhir : Interaksi Hand Tracking pada Objek Augmented Reality
Menggunakan Stereo Vision
Pembimbing I : Sutarno, M.T.
Pembimbing II : Sri Desy Siswanti, M.T.
Perbaikan/Saran :

*Perbaikan 2 Cam ← key pad → kpl di drisell
key ut av → kpl drisell.*

Jangka Waktu Perbaikan :

Telah diperbaiki sesuai dengan saran dan koreksi tim penguji ujian komprehensif.

No.	Nama Penguji	Status Penguji	Tanda Tangan
1.	Erwin, M.Si.	Anggota	

Palembang, 18 Oktober 2017
Ketua Jurusan Sistem Komputer,

Rossi Passarella, M.Eng.
NIP 19780611 201012 1 004



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS SRIWIJAYA
FAKULTAS ILMU KOMPUTER
JURUSAN SISTEM KOMPUTER

Jalan Palembang – Prabumulih Km. 32 Indralaya Kabupaten Ogan Ilir Kode Pos 30662
Telepon (0711)7072729, 379249, 581700 Faksimili (0711) 379248, 581710
Pos-el : info@ilkom.unsri.ac.id

FORM PERBAIKAN UJIAN TUGAS AKHIR II

Nama Mahasiswa : Robbiansyah Asmir
NIM : 09111001020
Jurusan : Sistem Komputer
Hari / Tanggal : Sabtu / 24 Maret 2018
Waktu : 09.01 s.d 11.01
Judul Tugas Akhir : Interaksi Hand Tracking Pada Objek Augmented Reality
Menggunakan Stereo Vision
Pembimbing : Sutarno, M.T.
Sri Desy Siswanti, M.T.
Perbaikan/Saran :

ada % error.
coding dipelajari

Jangka Waktu Perbaikan :

Telah diperbaiki sesuai dengan saran dan koreksi tim penguji ujian komprehensif.

No.	Nama Penguji	Status Penguji	Tanda Tangan
1.	Rossi Passarella, M.Eng.	Anggota	

Palembang, 24 Maret 2018
Ketua Jurusan Sistem Komputer

Rossi Passarella, M.Eng.
NIP 19780611 201012 1 004



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS SRIWIJAYA
FAKULTAS ILMU KOMPUTER
JURUSAN SISTEM KOMPUTER

Jalan Palembang – Prabumulih Km. 32 Indralaya Kabupaten Ogan Ilir Kode Pos 30662
Telepon (0711)7072729, 379249, 581700 Faksimili (0711) 379248, 581710
Pos-el : info@ilkom.unsri.ac.id

FORM PERBAIKAN UJIAN TUGAS AKHIR II

Nama Mahasiswa : Robbiansyah Asmir
NIM : 09111001020
Jurusan : Sistem Komputer
Hari / Tanggal : Sabtu / 24 Maret 2018
Waktu : 09.01 s.d 11.01
Judul Tugas Akhir : Interaksi Hand Tracking Pada Objek Augmented Reality
Menggunakan Stereo Vision
Pembimbing : Sutarno, M.T.
Sri Desy Siswanti, M.T.
Perbaikan/Saran :

- perbaiki code program.
- saw dy pyri 2

Jangka Waktu Perbaikan :

Telah diperbaiki sesuai dengan saran dan koreksi tim penguji ujian komprehensif.

No.	Nama Penguji	Status Penguji	Tanda Tangan
1.	Erwin, M.Si.	Anggota	

Palembang, 24 Maret 2018
Ketua Jurusan Sistem Komputer

Rossi Passarella, M.Eng.
NIP 19780611 201012 1 004



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS SRIWIJAYA
FAKULTAS ILMU KOMPUTER
JURUSAN SISTEM KOMPUTER

Jalan Palembang – Prabumulih Km. 32 Indralaya Kabupaten Ogan Ilir Kode Pos 30662
Telepon (0711)7072729, 379249, 581700 Faksimili (0711) 379248, 581710
Pos-el : info@ilkom.unsri.ac.id

FORM PERBAIKAN UJIAN TUGAS AKHIR II

Nama Mahasiswa : Robbiansyah Asmir
NIM : 09111001020
Jurusan : Sistem Komputer
Hari / Tanggal : Sabtu / 24 Maret 2018
Waktu : 09.01 s.d 11.01
Judul Tugas Akhir : Interaksi Hand Tracking Pada Objek Augmented Reality
Menggunakan Stereo Vision
Pembimbing : Sutarno, M.T.
Sri Desy Siswanti, M.T.
Perbaikan/Saran :

Tulis.

Jangka Waktu Perbaikan :

Telah diperbaiki sesuai dengan saran dan koreksi tim penguji ujian komprehensif.

No.	Nama Penguji	Status Penguji	Tanda Tangan
1.	Sri Desy Siswanti, M.T.	Sekretaris	<i>[Signature]</i>

Palembang, 24 Maret 2018
Ketua Jurusan Sistem Komputer

Rossi Passarella, M.Eng.
NIP 19780611 201012 1 004



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS SRIWIJAYA
FAKULTAS ILMU KOMPUTER
JURUSAN SISTEM KOMPUTER

Jalan Palembang – Prabumulih Km. 32 Indralaya Kabupaten Ogan Ilir Kode Pos 30662
Telepon (0711)7072729, 379249, 581700 Faksimili (0711) 379248, 581710
Pos-el : info@ilkom.unsri.ac.id

FORM PERBAIKAN UJIAN TUGAS AKHIR II

Nama Mahasiswa : Robbiansyah Asmir
NIM : 09111001020
Jurusan : Sistem Komputer
Hari / Tanggal : Sabtu / 24 Maret 2018
Waktu : 09.01 s.d 11.01
Judul Tugas Akhir : Interaksi Hand Tracking Pada Objek Augmented Reality
Menggunakan Stereo Vision
Pembimbing : Sutarno, M.T.
Sri Desy Siswanti, M.T.
Perbaikan/Saran :

**Teori s program*

Jangka Waktu Perbaikan :

Telah diperbaiki sesuai dengan saran dan koreksi tim penguji ujian komprehensif.

No.	Nama Penguji	Status Penguji	Tanda Tangan
1.	Sutarno, M.T.	Ketua	<i>[Signature]</i>

Palembang, 24 Maret 2018
Ketua Jurusan Sistem Komputer

[Signature]
Rossi Passarella, M.Eng.
NIP 19780611 201012 1 004



UNIVERSITAS SRIWIJAYA
FAKULTAS ILMU KOMPUTER
JURUSAN SISTEM KOMPUTER

Jalan Palembang - Prabumulih Km.32 Inderalaya, Ogan Ilir, Kode Pos 30662
Telepon (0711)7072729, 379249, Faksimile (0711)379248, email:ilkom@unsri.ac.id

KARTU KENDALI PLAGIAT

Nama : Robbyansyah Asmir
NIM : 09111001020
Judul Proposal Skripsi/Skripsi : *Hand Tracking Pada Objek Augmented Reality Menggunakan Stereo Vision*
No. SK Pembimbing : 0021/UN9.1.9/DL/2017
Pembimbing I : Sutarno, S.T, M.T
Pembimbing II : Sri Desy Siswanti, S.T, M.T

No	Proposal Skripsi	Skripsi	Tanggal Cek Plagiat	Nilai Ithenticate	Paraf		
					Admin Jurusan	Pembimbing I	Pembimbing II
1	✓		31/10/17	14%	<i>RS</i>	<i>SD</i>	<i>SS</i>

* Mohon kartu ini dibawa saat sidang proposal dan sidang skripsi

Mengetahui
Ketua Jurusan,

Rossi Passarella, M.Eng.
NIP 19780611 201012 1 004



UNIVERSITAS SRIWIJAYA
FAKULTAS ILMU KOMPUTER
JURUSAN SISTEM KOMPUTER

Jalan Palembang - Prabumulih Km.32 Inderalaya, Ogan Ilir, Kode Pos 30662
Telepon (0711)7072729, 379249, Faksimile (0711)379248, email:ilkom@unsri.ac.id

KARTU KENDALI PLAGIAT

Nama : Robbyansyah Asmir
NIM : 0911001020
Judul Proposal Skripsi/Skripsi : Interaksi Hand Tracking pada Objek Augmented Reality
Menggunakan Stereo Vision
No. SK Pembimbing : 0066
Pembimbing I : Sutarno, S.T.,M.T
Pembimbing II : Sri Desi Siswanti, S.T.,M.T

No	Proposal Skripsi	Skripsi	Tanggal Cek Plagiat	Nilai Ithenticate	Paraf		
					Admin Jurusan	Pembimbing I	Pembimbing II
1.	-	✓	8/3 2018	123	f.	f.	llia

* Mohon kartu ini dibawa saat sidang proposal dan sidang skripsi

Mengetahui
Ketua Jurusan,

Rossi Passarella, M.Eng.
NIP 19780611 201012 1 004